VERÄNDERUNGEN IM DISKURS ÜBER INFORMATIONSSICHERHEIT

EINE UNTERSUCHUNG VON JOURNAL- UND INTERNETBASIERTEN BEITRÄGEN MITTELS TEXT-MINING

KAI DIETRICH, DANIEL KÄS

Diplomarbeit

Informatik und Gesellschaft
Fakultät IV
Technische Universität Berlin

15. September 2009

Kai Dietrich, Daniel Käs: *Veränderungen im Diskurs über Informationssicherheit,* Diplomarbeit, © 15. September 2009

GUTACHTER UND BETREUER:
Prof. Dr. Bernd Lutterbeck
Prof. Dr. Hans-Ulrich Heiß
Dr. Ing. Frank Pallas

Die Grafiken in dieser Arbeit wurden unter Verwendung der Oxygen-Icons (`http://www.oxygen-icons.org/`, CC-BY-SA, Oxygen Team) und der Skadge SVG Widgets (`http://openclipart.org/SOMbanners/files/skadge/10004`, CC-PD) erstellt. Das Python Logo wurde entsprechend der Lizensierung der Python Software Foundation verwendet.

DANKSAGUNGEN

Das Leben war hart in den letzten sechs Monaten: Wenig Schlaf, schlechte Ernährung, Kaffee gab es aus Tassen, die sich von innen bereits dauerhaft braun färbten. Mit einem Lächeln auf den Lippen, wie es sonst nur zu einem John Wayne gehört und die Hutkrempe etwas tiefer auf die Stirn gezogen, wurde diese Arbeit von zwei Cowboys nun endlich abgegeben. Jedoch, auch Cowboys sind nicht immer nur allein mit ihren Pferden und Kühen unterwegs – der Sattel, auf dem sie reiten, wurde vom Sattler in der Stadt gefertigt und der Whiskey, den sie in der Bar trinken, kommt auch nicht aus dem Brunnen. Und so wäre auch diese Arbeit nicht ohne die Hilfe einiger wichtiger Personen zustande gekommen.

Als Erstes möchten wir uns bei unserem Betreuer Frank Pallas bedanken. Wer einmal ein wenig in dem gelesen hat, was er selbst geschrieben hat, wird erkennen, was für ein brillanter Geist in ihm steckt und zuweilen auch auf andere scheint. Dieses Vorbild und seine Art, neue Schwierigkeiten in Leichtigkeiten aufzulösen, hat uns immer wieder getrieben nicht auf halbem Wege aufzuhören. Ebenso möchten wir uns bei Prof. Bernd Lutterbeck bedanken, der es versteht seinen Studenten und Mitarbeitern die Freiheit zu lassen, die so wichtig ist, damit sie ihre eigenen Ideen (wie unsere) umsetzen können und dabei zudem jede erdenkliche Unterstützung leistet. So hat er auch uns die Freiheit gegeben, diese Arbeit zu realisieren und ermöglichte uns den Kontakt mit der IEEE. Bei dieser Gelegenheit möchten wir uns auch bei Frau Bettina Golz, Leiterin der Abteilung Medienbearbeitung der Universitätsbibliothek TU Berlin, für ihre Unterstützung bei der Kontaktaufnahme mit der IEEE bedanken. Ebenso möchten wir uns natürlich bei der IEEE selbst für die Bereitstellung der riesigen Datenmengen bedanken, auch wenn vorerst noch nicht die gewünschten Ergebnisse daraus hervorgegangen sind.

Neben diesen, für uns beide wichtigen Personen, sind nun sicher noch ein paar persönliche Anmerkungen der beiden einzelnen Autoren angebracht.

DANKSAGUNGEN, DANIEL KÄS

Um es mit den Worten eines berühmten Cowboys zu sagen: *„Alles, was ich bei meiner Ankunft wollte, war eine Flasche Whiskey und ein heißes Bad.“*[1] Ich bin jetzt am Ende der Diplomarbeit und damit meines Studiums angekommen. Allerdings habe ich mich nicht allein durch die Prärie schlagen müssen – mir standen einige helfende Hände zur Seite, denen ich gerne meinen Dank aussprechen möchte. Zu allererst möchte ich meiner Familie danken, die mir während dieser nicht immer leichten Zeit zur Seite stand und mich in vielerlei Hinsicht unterstützt hat. Dies gilt insbesondere meinen Eltern Gabriela und Ulrich Käs die mir nicht nur geistig, sondern auch materiell unter die Arme gegriffen haben, so dass ich mich komplett auf mein Studium und die Diplomarbeit konzentrieren konnte. Desweiteren möchte ich den Mitgliedern der

1 Nein, es ist nicht John Wayne, sondern Clint Eastwood in dem Western-Streifen „Ein Fremder ohne Namen“.

Freitagsrunde für den Beweis danken, dass ein Studium nicht nur harte Arbeit ist, sondern auch Spaß machen darf. Die vielen inspirierenden Gespräche und gemeinsamen Tätigkeiten, welche mich während meiner Studienzeit begleitet haben, werden mir noch lange in Erinnerung bleiben. Zuletzt möchte ich noch Marco Blumendorf und Dirk Roscher aus meinem Forschungsteam im DAI-Labor für das unkomplizierte Verschieben der Arbeitszeiten danken, was mir besonders in den letzten Wochen genügend Luft für die Diplomarbeit geschaffen hat. Und auch die vielen Freunde, die mir mit Rat und Tat in den letzten Jahren während des Studiums beiseite standen, sollen erwähnt werden, auch wenn es mir jetzt unmöglich ist hier alle einzeln aufzuzählen.

DANKSAGUNGEN, KAI DIETRICH

Manchmal ist es nicht direkte Hilfe, für die es sich lohnt Danke zu sagen, manchmal ist es auch einfach schon das geduldige Ertragen unkommunikativen oder scheinbar unsozialen Verhaltens. Das Bild des Grashalm-kauenden, grübelnden und ungesprächigen Cowboys trifft vielleicht ganz gut das merkwürdige Verhalten arbeitender Großstädter in der Diplomarbeitszeit. Ich jedenfalls war bestimmt mal so. Und dafür, dies ertragen und mich nebenbei sogar auch immer mal wieder aufgemuntert zu haben, möchte ich meiner Partnerin Anja Hähnel danken, die während dieser Zeit liebevoll an meiner Seite stand. Ebenso „ertragen“ musste mich in dieser Zeit meine Mutter Susanne Dietrich, der ich nicht nur dafür, sondern auch für die unendliche Unterstützung während der gesamten Studienzeit danken möchte – eine Studienzeit, die von so vielen Studenten als anstrengend empfunden wird. Mir wurde sie durch die Tätigkeit und Gemeinschaftlichkeit in der Fachschaft der Fakultät IV sehr viel kurzweiliger gemacht. Dafür möchte ich allen Mitgliedern der Freitagsrunde und insbesondere Robert Buchholz, ohne den ich dort nie gelandet wäre, danken. Die Liste der Menschen, die mich auf meinem Weg die letzten Jahre begleitet haben und denen ich zu Dank verpflichtet bin, ließe sich noch beliebig erweitern. Papier jedoch, ist endlich und so möchte ich mich bei alljenen bedanken, die mich in den letzten Jahren und Monaten unterstützt und inspiriert haben.

ZUSAMMENFASSUNG

In dieser Arbeit wird der Themenwandel im Diskurs über Informationssicherheit untersucht. Aus vielen möglichen Methoden des Data-Minings bzw. den speziellen Verfahren des Text-Minings wird ein Verfahren zur Einteilung von Dokumenten in Unterthemen gewählt. Diese Methode basiert auf TFIDF-Feature-Vektoren und dichtebasiertem Clustering mit dem DBSCAN-Algorithmus. Untersucht werden großen Mengen wissenschaftlicher Arbeiten und Beiträge auf Mailinglisten und Newsgroups, die von den verwendeten Algorithmen in Gruppen ähnlicher Dokumente eingeteilt werden. Für diese Gruppen kann dann eine zeitliche Verteilung der Publikationshäufigkeit aufgetragen, sowie eine automatisch generierte Zusammenfassung der Inhalte der Gruppen präsentiert werden. Aus diesen Publikationsgraphen und den Inhalten der zugehörigen Gruppen werden anschließend drei Thesen über die Entwicklung des Diskurses über Informationssicherheit aufgestellt, unterstützt und mit den Modellen aus von Solms (2000) und Pallas (2009) in Zusammenhang gebracht. In einer Bewertung der Methode muss jedoch festgestellt werden, dass es noch nicht möglich ist, ausreichend eindeutige Gruppen mit abgrenzbaren Inhalten aus den Dokumenten zu extrahieren. Diese Schwäche des Verfahrens musste in die Betrachtung der Ergebnisse mit einfließen.

INHALTSVERZEICHNIS

1 ZIEL DER ARBEIT 1

2 DATEN 3
- 2.1 Datenquellen . . . 3
- 2.2 Charakteristik der Daten . . . 5
  - 2.2.1 Journale . . . 5
  - 2.2.2 Newsgroups . . . 8
  - 2.2.3 Mailinglisten . . . 9

3 METHODEN 13
- 3.1 The Big Picture . . . 13
  - 3.1.1 Clustering und Klassifikation . . . 15
  - 3.1.2 Vorverarbeitung . . . 16
  - 3.1.3 Visualisierung der Ergebnisse . . . 18
- 3.2 Mögliche Wege . . . 18
  - 3.2.1 Vorverarbeitung der Textdaten . . . 19
  - 3.2.2 Klassifizieren . . . 37
  - 3.2.3 Clustern . . . 44
  - 3.2.4 Auswertungsmethoden . . . 54
- 3.3 Wahl der Methode . . . 58
  - 3.3.1 Clustern vs. Klassifizieren . . . 59
  - 3.3.2 Auswahl des Clusterverfahrens . . . 62
  - 3.3.3 Datenaufbereitung . . . 63
  - 3.3.4 Vorverarbeitung . . . 65
  - 3.3.5 Auswahl der Clusterparameter . . . 69
  - 3.3.6 Auswertungsphase . . . 72

4 ERGEBNISSE 77
- 4.1 Beschreibung der Ergebnisse . . . 77
  - 4.1.1 ACM TISSEC . . . 78
  - 4.1.2 Computers & Security . . . 80
  - 4.1.3 IEEE . . . 83
  - 4.1.4 Security-Basics . . . 85
  - 4.1.5 Infosec-News . . . 88
  - 4.1.6 comp.sec.misc . . . 91
- 4.2 Analyse der angewandten Methode . . . 93
  - 4.2.1 Besondere Beobachtungen . . . 93
  - 4.2.2 Bewertung der Methode . . . 100
- 4.3 Aussagen . . . 103
  - 4.3.1 Thesen . . . 104
  - 4.3.2 Relevanz der Thesen . . . 111

5 IN WEITER FERNE 115
- 5.1 Datenaufbereitung . . . 115
- 5.2 Vorverarbeitung . . . 115
- 5.3 Clustering . . . 118
- 5.4 Auswertung . . . 119

LITERATURVERZEICHNIS 123

A ANHANG 133
- A.1 ACM TISSEC . . . 133
- A.2 Computers & Security . . . 141
- A.3 Security-Basics . . . 149
- A.4 Infosec-News . . . 157

A.5 comp.sec.misc 165
A.6 IEEE Xplore 173

ABBILDUNGSVERZEICHNIS

Abbildung 1 Beispielhafte Visualisierung optimaler Daten für die Diskursanalyse eines Themas für die Jahre 1950 bis 2009. 4
Abbildung 2 Visualisierung der Charakteristik der Publikationen in der ACM TISSEC. 7
Abbildung 3 Visualisierung der Charakteristik der Publikationen in der Computers & Security. 7
Abbildung 4 Visualisierung der Charakteristik der Publikationen in der IEEE Digital Library. 8
Abbildung 5 Visualisierung der Charakteristik der Beiträge in der Newsgroup `comp.security.misc`. 10
Abbildung 6 Visualisierung der Charakteristik der Beiträge in der Mailingliste Security-Basics. 11
Abbildung 7 Visualisierung der Charakteristik der Beiträge in der Mailingliste Infosec-News. 11
Abbildung 8 Beispiel eines Clusterings mittels DBSCAN 16
Abbildung 9 Funktionsweise der PCA im 2D-Raum anhand zufälliger Beispieldaten 35
Abbildung 10 Verteilung der Varianz über die Dimensionen nach PCA für den ACM TISSEC Corpus 36
Abbildung 11 Beispiel für das XOR-Problem 41
Abbildung 12 Beispiel für Clustering mittels K-Means 46
Abbildung 13 Beispiel für Core-, Border und Noise-Points 48
Abbildung 14 Beispiel für K-Dist-Graphen 49
Abbildung 15 Zusammenfassung der Analysemethode 60
Abbildung 16 K-Dist-Graphen für TISSEC 70
Abbildung 17 K-Dist-Graphen für Computers & Security 70
Abbildung 18 Parametergraph für den „Computers & Security“-Corpus 71
Abbildung 19 Computers & Security Cluster 11 81
Abbildung 20 Computers & Security Cluster 1 82
Abbildung 21 Computers & Security Cluster 33 83
Abbildung 22 IEEE Cluster 0 84
Abbildung 23 Security-Basics Cluster 1 86
Abbildung 24 Security-Basics Cluster 6 87
Abbildung 25 Infosec-News Cluster 26 und 0 89
Abbildung 26 Infosec-News Cluster 32 91
Abbildung 27 *Security Waves* nach von Solms (2000), übernommen von Pallas (2009, S.31) 112

TABELLENVERZEICHNIS

Tabelle 1 Beispielhafter Ausschnitt aus einer OCR-MAPPING Ersetzungstabelle 26
Tabelle 2 Beispiel einer Feature-Matrix 29
Tabelle 3 Feature-Matrix M mit Anzahl der Wörter für jedes Dokument 32
Tabelle 4 Beispiel aus Tabelle 3 nach einer PCA (gerundete Werte) 36
Tabelle 5 Dokumente des Trainingssets mit ihren enthaltenen Wörtern und Klassenzugehörigkeiten 39
Tabelle 6 Vergleich der Cluster-Algorithmen 54
Tabelle 7 Wahl der DBSCAN-Parameter für die verschiedenen Quellen 72
Tabelle 8 Anzahl von Dokumente und Dimensionen in den Corpora 78
Tabelle 9 Cluster über Basistechnologien und deren Verläufe, sortiert nach Aussagekraft 106
Tabelle 10 Cluster über höhere Technologien und deren Verläufe, sortiert nach Aussagekraft 109
Tabelle 11 Cluster über *Management-Aspekte* der Informationssicherheit und deren Verläufe, sortiert nach Aussagekraft 110

GLOSSAR

CLUSTERING [2]
: Gruppe von Clustern, die als Ergebnis der Anwendung eines Cluster-Verfahrens entstehen

CORPUS
: eine Menge von Dokumenten, die sich nach einer ersten Datenaufbereitung in einem konsistenten Zustand befinden

DATENQUELLE
: eine Menge von Dokumenten, die noch keiner weiteren Datenaufbereitung unterzogen wurden

DICTIONARY [2]
: Liste von Terms und zugehörigen Term-Häufigkeiten, siehe auch lokales/globales Dictionary

FEATURE [2]
: Eigenschaft eines Dokuments, welches zur Unterscheidung herangezogen wird, z. B. kann das Vorhandensein eines Wortes als Feature verwendet werden; üblicherweise werden Features numerisch dargestellt

FEATURE-MATRIX
: alle Feature-Vektoren eines Corpus zusammen als Spaltenvektoren einer Matrix; wird auch „Term-Document-Matrix“ genannt oder manchmal verkürzt „TD-Matrix“

FEATURE-RAUM
: der n-dimensionale Raum, der durch die n verschiedenen Features (als orthogonale, kanonische Einheitsvektoren) aufgespannt wird; jeder Feature-Vektor ist ein Ortsvektor im Feature-Raum

FEATURE-VEKTOR
: geordnete Menge von numerisch dargestellten Features als n-dimensionaler Vektor, welcher ein Dokument repräsentiert; jede der n Dimensionen steht für ein spezifisches Feature, ist ein Feature im Dokument nicht vorhanden, wird die Stelle mit dem Wert Null besetzt

GLOBALES DICTIONARY
: Dictionary, welches die Terms und Term-Häufigkeiten des gesamten Corpus enthält

K-DIST-GRAPH
: Diagramm zum Finden geeigneter Parameter für den DBSCAN-Algorithmus; gibt für jeden Punkt im Feature-Raum den Abstand zum k nächsten Nachbarpunkt an, die Abstände werden absteigend sortiert angezeigt

LOKALES DICTIONARY
: Dictionary, welches die Terms und Term-Häufigkeiten eines einzigen Dokuments enthält

2 englischer Begriff wird beibehalten, da zu einem feststehenden Fachbegriff geworden

PARAMETERGRAPH
: Graph zur Bestimmung der optimalen Parameter zum Clustering; die Anzahl der ungeclusterten Punkte (Noise) wird über die Varianz der Größe der erhaltenen Cluster aufgetragen, Optimum ist der Null-Punkt; jeder Punkt wird mit den verwendeten Parametern annotiert

POS-TAGGING [2]
: auch Part-Of-Speech-Tagging, Klasse von Verfahren welche Worten eines Satzes jeweils ihre Wortart anhand der Stellung im Satz zuordnen

PUBLIKATIONSGRAPH
: Graph welcher die Anzahl der Veröffentlichungen einer Datenquelle oder eines Clusters über die Zeit darstellt

SATZ-ZUSAMMENFASSUNG
: spezielle Form der Zusammenfassung, die aus nach Wichtigkeit sortierten Sätzen besteht

STOP-WORDS [2]
: List von Worten die keine relevante Bedeutung haben und deshalb aus den Dokumenten entfernt werden können

STEMMING [2]
: Klasse von Verfahren Worte auf ihre Wortstämme oder kürzere Wortkerne zu reduzieren

TERM [2]
: Sinneinheit aus einem Dokument, kann aus einem Wort, Wortstamm oder mehreren Wörtern bestehen

TFIDF-WERT
: spezielle Form der Berechnung von numerischen Features aus der Worthäufigkeit im Dokument und der Worthäufigkeit im gesamten Corpus; hohe TFIDF Werte bedeuten, dass ein Feature häufig im betrachteten Dokument und selten in anderen Dokumenten des Corpus' vorkommt, es ist also sehr beschreibend für das betrachtete Dokument

TITEL-ZUSAMMENFASSUNG
: spezielle Form der Zusammenfassung, die aus Titeln von nach Wichtigkeit sortierten Dokumenten besteht

TOKENIZATION [2]
: Klasse von Verfahren lange Zeichenfolgen in eine Folge kurzer Sinneinheiten zu zerlegen, oftmals an Leerzeichen

WORD-THRESHOLD [2]
: Verfahren zur Dimensionsreduktion; Features, deren Wert im Feature-Vektor nicht bestimmten Kriterien genügen, werden verworfen

WORT-ZUSAMMENFASSUNG
: spezielle Form der Zusammenfassung, die aus nach Wichtigkeit sortierten Schlüsselwörtern besteht

ZUSAMMENFASSUNG
: engl. Summary; wenn nicht in seiner normalen deutschen Bedeutung verwendet, dann ist eine Zusammenfassung ein spezielles, automatisch generiertes Exzerpt eines Textes, das diesen möglichst gut repräsentieren soll

1 ZIEL DER ARBEIT

Informationssicherheit befindet sich im Wandel. Dies betrifft sowohl die praktische Umsetzung von Informationssicherheitsmaßnahmen als auch den Diskurs darüber, wie Informationssicherheit am besten gewährleistet werden könnte. Jeder der in der Wirtschaft oder Wissenschaft für längere Zeit mit dem Problemgebiet Informationssicherheit in Berührung kommt, wird dies erleben. Man wird erleben, dass nicht mehr nur technische Maßnahmen ergriffen werden, sondern auch Regelwerke aufgestellt werden müssen und für deren Einhaltung gesorgt werden muss. Der BSI-Grundschutz-Standard 100-1 beginnt sogar bereits mit den einleitenden Worten, dass „[d]ie Praxis [...] gezeigt hat, dass eine Optimierung des Sicherheitsmanagements oftmals die Informationssicherheit effektiver und nachhaltiger verbessert als Investitionen in Sicherheitstechnik“[1]

Diese Veränderung der praktisch ergriffenen Maßnahmen müsste logischerweise von einem Diskurs der Wissenschaftler und Anwender begleitet werden. Den Standpunkt einen Wandel in der Entwicklung der Informationssicherheit beobachten zu können vertritt unter anderem auch von Solms (2000). Er postuliert gar drei „Wellen“, in denen dieser Wandel verlaufen würde: eine technische Welle, eine Management-Welle und eine Welle der Institutionalisierung. Er bleibt den Lesern jedoch ebenso ein genaue Erklärung der Wellen-Metapher schuldig wie einen Beweis für die Existenz der Wellen und beschränkt sich auf die rein subjektiv empfundenen Veränderungen. Pallas (2009) greift diese Hypothese auf, analysiert sie und kommt zu einem anderen Modell, welches sich durch historische und technisch induzierte Entwicklungen sowie ökonomische Überlegungen stützen lässt.

Eine statistische Untersuchung des tatsächlichen Diskurses über Informationssicherheit stand jedoch noch aus. Decken sich die subjektive Empfindung und die analytischen Überlegungen mit der Realität?

Ziel dieser Arbeit ist es genau dies zu überprüfen. Die Informatik und in Schnittmenge die Computerlinguistik hat Methoden entwickelt mit denen große Mengen von Dokumenten auf ihren Inhalt hin analysiert werden können. Diese Methoden ermöglichen die Einteilung von Dokumenten nach thematischer Ähnlichkeit und ermöglichen damit die Quantifizierung einer wie auch immer gearteten Wellen-Metapher. Ergebnis soll also eine statistisch fundierte Beschreibung der Themenwandel innerhalb der Informationssicherheit sein. Ein noch notwendiger Schritt dahin ist es aus den vielen möglichen Methoden die zur Verfügung stehen eine robuste Methode zur automatisierten Untersuchung von Themenwandeln in Dokumenten zu wählen. Ein weiteres Problem ist die Erfassung der Gesamtmenge. Eine statistische Aussage trifft ersteinmal nur für die Daten zu, über die sie berechnet wurde. Nur wenn die betrachtete Stichprobe repräsentativ für die Gesamtmenge ist, kann die Aussage auch zuverlässig verallgemeinert werden. Im Kapitel 2 wird die von uns betrachtete Stichprobe genau beschrieben. Hier sei nur so viel gesagt, dass sowohl wissenschaftliche Arbeiten als auch der Diskurs von Praktikern und Anwendern ausgiebig betrachtet werden.

1 Siehe Bundesamt für Sicherheit in der Informationstechnik (2008a, S.5).

Der Anspruch repräsentativ für den möglichen Gesamtdiskurs zu sein, soll jedoch nicht erhoben werden.

Was am Ende der Arbeit entsteht, ist also:

1. Eine Methode, mit der wissenschaftlicher Diskurs im allgemeinen quantitativ untersucht werden kann.
2. Eine Untersuchung des Diskurses über Informationssicherheit anhand der verfügbaren Quellen mit der erarbeiteten Methode.

2 DATEN

Vor der Betrachtung einer möglichen Methode zur Analyse des Diskurses müssen Überlegungen angestellt werden, welche Daten das Themengebiet adäquat beschreiben könnten. Das gewählte Themengebiet „Informationssicherheit“ spielt seit dem Beginn des Time-Sharing in den 1950er und 1960ern in der Rechnertechnologie und Informatik eine Rolle[1]. Die beteiligten Akteure fanden sich in Universitäten und den Unternehmen, die Rechner herstellten. Der bis heute nachvollziehbare frühe Diskurs fand in den Journalen der „Association for Computing Machinery“ (ACM) und dem „Institute of Electrical and Electronics Engineers“ (IEEE) sowie über Veröffentlichungen von Behörden[2] statt. Alles andere aus dieser frühen Zeit lässt sich wohl nur noch schwer recherchieren, geschweige denn vollständig digital erfassen.

Mit der Verbreitung der Rechnertechnologie kamen auch zunehmend Anwender und immer mehr Entwickler und Forscher mit den Fragen der Informationssicherheit in Berührung. Heute gibt es sowohl bei der ACM als auch bei der IEEE und vielen anderen Publikationsstellen eigene spezialisierte Journale, die sich diesem Thema widmen[3]. In Newsgroups und auf Mailinglisten können Anwender über die verschiedensten Probleme im Bereich der Informationssicherheit diskutieren. In den folgenden Abschnitten wird eruiert, welche Daten zur Analyse optimal wären, welche Daten verfügbar sind und welche Datenquellen zur Analyse herangezogen werden können.

2.1 DATENQUELLEN

Um eine allgemeingültige Aussage über den Diskurs über Informationssicherheit zu treffen, wäre ein umfassender Datensatz nötig. Dieses müsste sowohl den wissenschaftlichen als auch den nicht-wissenschaftlichen Diskurs kontinuierlich vom Entstehen des Themas bis zum aktuellen Stand in seiner Reichweite gleichartig erfassen. Gleichartig in dem Sinne, dass die erfasste Breite des Themas zeitlich konstant ist und nur der darin stattfindende Diskurs sich quantitativ, vereinfachend gleichzusetzen mit der Tiefe, unterscheidet (visualisiert in Abb. 1).

Ein solches optimales Datenset ist vermutlich nicht existent. Zum einen wird mündlicher Diskurs üblicherweise überhaupt nicht erfasst, es kann also prinzipiell nur schriftlich Festgehaltenes analysiert werden, zum anderen gibt es überhaupt nur wenige Quellen, die in die 1950er Jahre zurück reichen. Quellen, die in Frage kommen sind:

- Bücher
- Beiträge in wissenschaftlichen Zeitschriften
- Beiträge in nicht-wissenschaftlichen Zeitschriften

1 Vgl. McCarthy (1992).
2 Als Beispiel dazu Ware (1979).
3 Bei der ACM sind dies die „Transactions on Information and System Security“ (TISSEC), bei der IEEE finden sich die „IEEE Security and Privacy“ sowie die „IET Information Security“. Die Zeitschrift „Computers & Security“ wird schon seit 1982 herausgegeben.

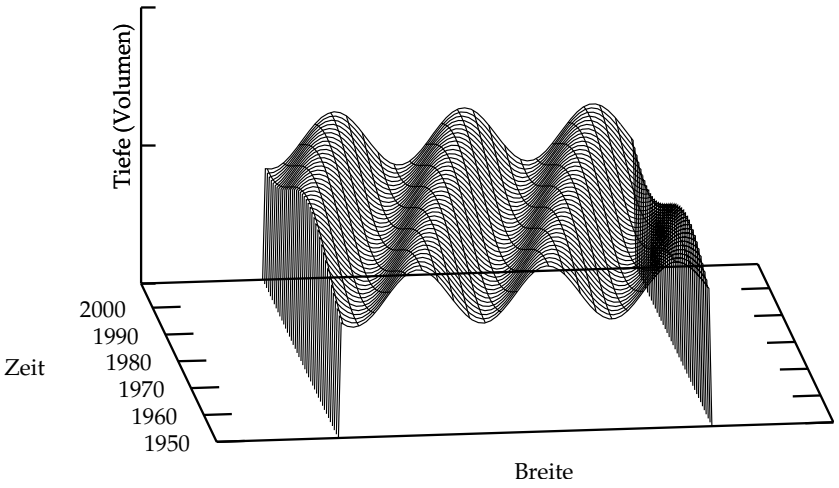


Abbildung 1: Beispielhafte Visualisierung optimaler Daten für die Diskursanalyse eines Themas für die Jahre 1950 bis 2009.

- persönliche Kommunikation (Briefe, E-Mails, Telefaxe, . . .)
- öffentliche Kommunikation (Newsgroups, Mailinglisten, Webseiten, Foren, Blogs, . . .)

Bücher sind (noch) nicht vollständig und allgemein digital verfügbar. Auch wenn ein beträchtlicher Umfang des Diskurses über Bücher stattfinden sollte, so kann dieser aus Aufwandsgründen nicht erfasst werden. Beiträge in wissenschaftlichen Zeitschriften sind zu großen Teilen digitalisiert (sofern sie archiviert wurden) und gegen Bezahlung bei den Herausgebern verfügbar. Insbesondere die digitalen Archive der IEEE und der ACM reichen weit zurück. In ebenso großem Umfang stehen die Archive großer nicht-wissenschaftlicher Zeitschriften elektronisch zur Verfügung[4]. Diese sind in ihrer Breite allerdings zu breit um sie als Quelle in Betracht ziehen zu können. Persönliche Kommunikation ist ebenso wie mündliche Kommunikation kaum archiviert und kann deshalb nicht betrachtet werden. Es bleibt öffentliche Kommunikation zwischen mehreren Kommunikationsteilnehmern. Diese Kommunikationsform ist im Prinzip erst seit der Existenz des Internets und den damit radikal gesenkten Kommunikationskosten möglich. Die frühesten elektronischen Many-to-Many Kommunikationsprotokolle waren wohl sog. Bulletin Board Systeme. Wenig später folgten die Usenet Protokolle und Mailinglisten. Da sowohl das Usenet als auch große Mailinglisten relativ gut archiviert und einfach zugänglich sind, werden diese hier genauer betrachtet.

Öffentliche Kommunikation in Form von Inhalten, die ungeordnet und ungesammelt im Internet publiziert sind (zu denken ist z. B. an Blog-Beiträge und statische Webseiten), sind zwar über Suchmaschinen auffindbar, genau diese nehmen jedoch eine schwer überschaubare Form der Sortierung und Filterung vor, so dass nicht klar ist, welche Folgen dies für eine Auswertung und die damit verbundenen Schlussfolgerungen hätte. Gestreute Informationsquellen müssen deshalb vorerst ausgeschlossen werden.

So bleibt die Untersuchung von:

4 Frankfurter Allgemeine Sonntagszeitung: ab 1993, stand 06/2009 (vgl. Frankfurter Allgemeine Zeitung 2009), New York Times: ab 1851 (vgl. New York Times 2009).

- elektronisch verfügbaren wissenschaftlichen Journalen
- elektronisch archivierten Newsgroups
- elektronisch archivierten Mailinglisten

Die verschiedenen Quellen zeigen jeweils ganz eigene Charakteristiken, die sich auf Umfang, Teilnehmer, Sprachstil und anderes beziehen. Im Folgenden werden die bearbeiteten Quellen genauer beschrieben.

2.2 CHARAKTERISTIK DER DATEN

Um dem Leser einen Überblick über die Größenordnungen und Beschaffenheiten unserer Datenquellen zu verschaffen, werden wir im Folgenden auf jede der untersuchten Quellen eingehen. Jede Quelle wird dabei mit einer groben, geschätzten Visualisierung ihrer Charakteristik über Publikationsvolumen und thematischer Breite versehen. Ebenso wird auf besondere Eigenheiten der Quellen, welche später in der Verarbeitung beachtet werden müssen, eingegangen.

2.2.1 *Journale*

Wissenschaftliche Zeitschriften sind von hohem Interesse für das Nachvollziehen eines Diskurses, da hier Fachleute miteinander diskutieren. Die Kommunikationsgeschwindigkeit ist dabei sehr langsam; eine Erwiderung kann erst im Rahmen der nächsten Ausgabe erfolgen. Zudem setzt eine Beitragsveröffentlichung in einem Journal meist ein Peer-Review vorraus, in welchem die „Fachwelt“ beurteilt, ob der Beitrag sowohl inhaltlich konsistent wie sprachlich und formal angemessen ist, als auch eine neue Erkenntnis darstellt. Ein Filter geht also der Veröffentlichung voraus. Dieser Filter kann sehr streng oder sehr locker in seiner Regulierungswirkung sein. Es lässt sich hier sogar argumentieren, dass Peer-Reviews die Veröffentlichung radikaler Neuerungen zu stark verhindern und damit grundsätzlich der Diskurs nicht in seiner Gesamtheit objektiv wiedergegeben werden kann[5].

Die genaue thematische Ausrichtung eines Journals wird üblicherweise von den Herausgebern definiert, kann sich über den Verlauf der Zeit aber natürlich auch ändern (ein für die Untersuchung unangenehmer Vorgang). Für die Untersuchung stehen drei mehr oder minder umfangreiche Quellen zu Verfügung, die sich stark in ihrer thematischen Breite, dem Publikationsvolumen und der zeitlichen Reichweite unterscheiden und die deshalb einer Beschreibung bedürfen.

Auf technischer Ebene liegen die Beiträge in den erfassten Journalen als PDF-Dokumente vor. Diese wurden üblicherweise von einer analogen Print-Ausgabe eingescannt und offensichtlich durch Optical Character Recognition (OCR), erkennbar an typischen Fehlern wie z. B. fehlenden Leerzeichen zwischen einzelnen Wörtern, in einen maschinenlesbaren Text umgewandelt. Bei der Verarbeitung der Dokumente muss auf diese Fehler geachtet werden. Auf sprachlicher Ebene genügen die Publikationen natürlich wissenschaftlichen Anforderungen.

5 Vgl. dazu auch Whitworth u. Friedman (2009), die argumentieren, dass der strenge Maßstab den Peer-Reviews induzieren zu verpassten Erkenntnissen („ommission errors“) führen kann.

2.2.1.1 *ACM TISSEC*

Die „Transactions on Information and System Security“ (TISSEC) werden seit 1998 von der ACM herausgegeben[6] – einem sehr späten Zeitpunkt in der Forschung über Informationssicherheit. Ältere Arbeiten bei der ACM sind also über allgemeiner gehaltene Journale gelaufen[7]. Die TISSEC wird hier als Low-Volume Journal trotzdem analysiert werden. Der genaue Themenfokus wird in den „Topics of Interest“[8] wie folgt beschrieben:

> „*Security Technologies*: authentication; authorization models and mechanisms; auditing and intrusion detection; cryptographic algorithms, protocols, services, and infrastructure; recovery and survivable operation; risk analysis; assurance including cryptanalysis and formal methods; penetration technologies including viruses, Trojan horses, spoofing, sniffing, cracking, and covert channels.
>
> *Secure Systems*: secure operating systems, database systems and networks; secure distributed systems including security middleware; secure web browsers, servers, and mobile code; specialized secure systems for specific application areas; interoperability, and composition.
>
> *Security Applications*: threats, system tradeoffs, and unique needs of applications; representative application areas include information systems, workflow, electronic commerce, electronic cash, copyright and intellectual property protection, telecommunications systems, wireless systems, and health care.
>
> *Security Policies*: confidentiality, integrity, availability, privacy, usage, and survivability policies; tradeoffs, conflicts and synergy among security objectives.“[8]

Die drei Themen „Security Technologies“, „Secure Systems“ und „Security Applications“ sind dabei stark technikorientiert. Es bleibt jedoch noch Raum für weitergehende, managementorienterte (wenn auch wohlmöglich nicht weniger formal ausgeführte) Beiträge, wenn im vierten Thema „Security Policies“ von „tradeoffs, conflicts and synergy among security objectives“ die Rede ist. Die Formulierung als strenge Positiv-Liste wird jedoch, auch wenn sie nicht als absolut feststehende Liste interpretiert werden sollte, absehbar dafür sorgen, dass Lösungsvorschläge für das Informationssicherheitsproblem aus gänzlich anderen Blickwinkeln schlicht nicht erfasst werden. Dass die Liste der erwünschten Themen zum letzten Mal am 11. Oktober 2001 durch die Betreiber bearbeitet wurde, zeigt vielleicht schon, dass ein Themenwechsel auch nicht erwünscht sein könnte.

Der Umfang der ACM TISSEC beträgt zum Zeitpunkt der Untersuchung 12 Jahrgänge (von 1998 bis 2009) mit jeweils vier Ausgaben bei jeweils ca. vier Beiträgen. Insgesamt stehen 177 Beiträge zur Untersuchung zur Verfügung. Die Charakteristik der ACM TISSEC ist in Abbildung 2 visualisiert.

---

6 Siehe Archiv der TISSEC aus `http://portal.acm.org/`.

7 So z. B. das HRU-Modell aus Harrison u. a. (1976) in den „Communications of the ACM“ oder das Take-Grant Modell aus Graham u. Denning (1971) auf der damals sehr bekannten und breiten „Fall Joint Computer Conference“ (Proceedings) sowie zum Take-Grant Modell später auch Lipton u. Snyder (1977) im „Journal of the ACM“.

8 Siehe ACM (2001).

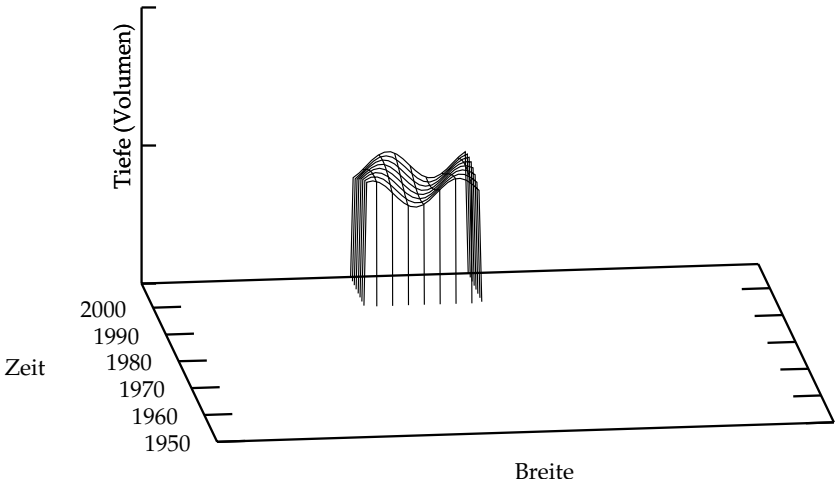


Abbildung 2: Visualisierung der Charakteristik der Publikationen in der ACM TISSEC.

2.2.1.2 *Computers & Security*

Die Zeitschrift „Computers & Security“ wird seit 1982 herausgegeben. Es ist die offizielle Zeitschrift des Technology Comittee 11 (Security and Protection in Information Processing Systems) der „International Federation for Information Processing“ (IFIP) unter dem Dach der UNESCO. Im Vergleich zur ACM TISSEC existierte diese Zeitschrift also schon deutlich länger. In der thematischen Reichweite deckt die Computers & Security nach eigener Aussage sowohl theoretische Spitzenforschung als auch praktisches Management ab.Diese große Reichtweite ist für eine Verfolgung der Themenwechsel sehr interessant.

Die Zeitschrift stand zur Untersuchung in den Jahrgängen von 1982 bis einschließlich 2002 zur Verfügung. Als in acht Ausgaben pro Jahr erscheinende Zeitschrift können ca. 2.400 Artikel analysiert werden. Die Charakteristik der Computers & Security ist in Abbildung 3 visualisiert.

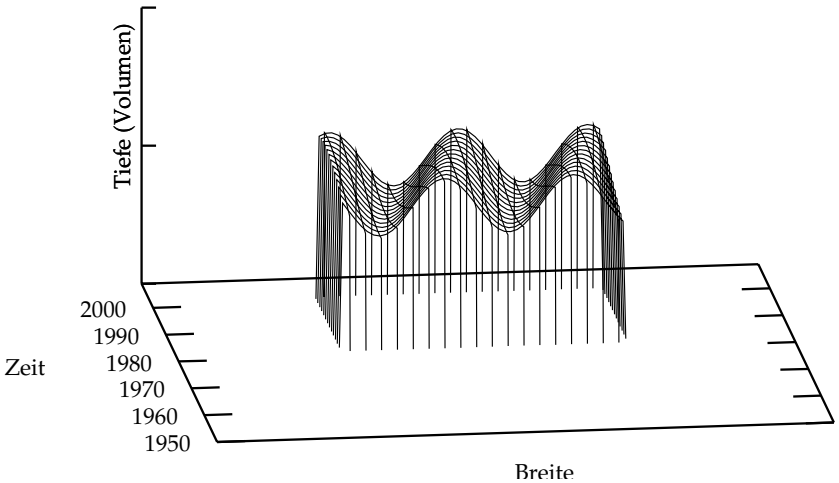


Abbildung 3: Visualisierung der Charakteristik der Publikationen in der Computers & Security.

2.2.1.3 *IEEE Digital Library*

Die IEEE ist eine große Vereinigung von Ingenieuren, die an Elektronik und vielen angrenzenden Bereichen arbeiten. Sie bietet Publikationsdienste, Konferenzen, Standardisierungsgremien und unzählige andere Dienste. Als eines der wichtigsten Foren für wissenschaftlichen Diskurs auf dem Gebiet der Elektrotechnik und Informatik archiviert die IEEE in ihrer Digital Library unzählige Beiträge[9] aus einem sehr breiten, technischen und nicht-technischen Publikationsangebot. Die thematische Reichweite in diesem Archiv hat sich natürlich mit dem Wachstum und der Ausdehnung der IEEE in neue Fachgebiete verändert. Das digitale Archiv der IEEE reicht dabei bis in die 1960er, teilweise sogar 1950er Jahre zurück.

Die IEEE hat für diese Untersuchung den Zugriff auf ca. 40.000 Beiträge freigegeben, die in der Digital Library auf das Stichwort „Security" passen. Dieses thematisch und quantitativ umfangreiche Archiv stellt vermutlich die wichtigste Ressource für das Nachvollziehen von Veränderung im Diskurs über Informationssicherheit dar. Die durch die Suche erfassten Dokumente enthalten sowohl Journal- als auch Konferenzbeiträge, sowie Standards als auch einen gewissen Anteil an Störelementen wie z. B. Werbeanzeigen. Der Signal-Rauschen-Anteil ist also etwas geringer als bei der Erfassung eines einfachen Journals. Die Charakteristik der IEEE Digital Library ist in Abbildung 4 visualisiert.

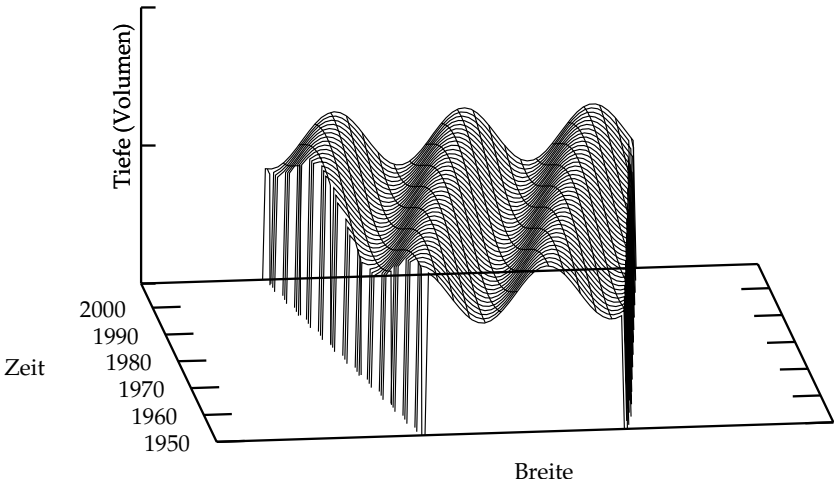


Abbildung 4: Visualisierung der Charakteristik der Publikationen in der IEEE Digital Library.

2.2.2 *Newsgroups*

Mit der Entstehung des Internets in den 60er Jahren wurden die Kommunikationskosten für Many-to-Many Kommunikation drastisch reduziert. Die ersten Formen der weltweiten Many-to-Many Kommunikation etablierten sich in den 70ern in Form von sog. Bulletin-Board-Systemen. Zu diesem System konnten sich die Benutzer mit Hilfe der ersten, langsamen Modems verbinden und Nachrichten für andere Benutzer hinterlassen. Dieser Modus der Kommunikation wurde durch

9 Laut der Ausgabe des Suchen-Formulars sind es über 2.000.000 (Stand: 10.06.2009).

die Einführung der Usenet-Protokolle in den 80er Jahren generalisiert und auf die universellen TCP/IP-Protokolle portiert[10].

Um das Usenet zu nutzen, benötigt der Nutzer Zugang zu einem Usenet-Server. Dieser stellt ihm die Nachrichten zum Abruf bereit. Alle Usenet-Server sind untereinander größtenteils synchronisiert ohne jedoch absolut zentral organisiert zu sein. Jeder Server-Administrator entscheidet selbst, welche Teile des Usenets er zur Verfügung stellt. Zentrale Abstimmung ist allerdings bei der Benennung und Gründung der sog. Newsgroups hilfreich. Alle Nachrichten im Usenet werden innerhalb einer Newsgroup platziert. Die Newsgroups werden meist in einem demokratischen Prozess durch die Benutzer hierarchisch nach Themensetzung benannt[11]. So gibt es die Newsgroups `comp.*` in denen alle Themen, die Computer-Bezug haben, aufgegliedert werden. Diese Gliederung geht weiter mit `comp.security.*` unterhalb derer sich z. B. `comp.security.firewalls`, `comp.security.ssh` und `comp.security.announce` befinden. Besonders interessant, da einen sehr großes Themengebiet umfassend, ist `comp.security.misc`. In dieser Newsgroup werden alle Themen besprochen, die in keiner der fest vordefinierten Newsgroups einen Platz finden.

Sprachlich befinden sich Beiträge in Newsgroups natürlich weit unterhalb wissenschaftlichen Standards. Schreibfehler, Umgangssprache und Internet-Jargon sind in vielen Nachrichten zu finden. Essentieller Teil des Diskurses in Newsgroups ist zudem das Zitieren von Teilen aus anderen Nachrichten, so dass der eigentlich neue Text in jedem Beitrag sehr kurz ist. Ebenso befinden sich oft am Ende von Nachrichten sog. Signaturen, kleine persönliche Textbrocken oder berühmte Zitate. Einige der Beiträge sind auch völlig außerhalb der Fachthemen, sog. Off-Topic Beiträge. Das Signal-Rauschen-Verhältnis ist also sehr schlecht. Diese Eigenheiten müssen bei der Verarbeitung beachtet werden.

Die Usegroup `comp.security.misc` ist von 1992 bis heute archiviert und umfasst ca. 25.800 Diskussionsstränge (Threads) mit insgesamt fast 100.000 Einzelbeiträgen von Fachleuten und Anwendern. Thematisch ist die Reichweite der Newsgroup jedoch über die Zeit ungefähr konstant geblieben. Die Charakteristik der Newsgroup `comp.security.misc` ist in Abbildung 5 visualisiert.

2.2.3 *Mailinglisten*

Neben dem Usenet hat sich bereits sehr früh die Verwendung von automatischen E-Mail-Verteilerlisten sog. Mailinglisten etabliert. Gerade im Bereich der Informationssicherheit sind diese ein viel genutztes Many-to-Many Kommunikationsmedium. Auf den meisten großen Mailinglisten zur Informationssicherheit werden jedoch hauptsächlich nur gefundene Schwachstellen in Softwareprodukten bekannt gegeben. So heißt es z. B. in der FAQ einer der bekanntesten Security-Mailinglisten BugTraq: „BugTraq is a full disclosure moderated mailing list for the *detailed* discussion and announcement of computer security vulnerabilities: what they are, how to exploit them, and how to fix them.“[12] oder auch in der Charta der ebenfalls sehr bekannten Security-Mailingliste

10 Zur Entwicklung der Kommunikationsprotokolle, incl. dem Usenet siehe Leiner u. a. (1997).

11 Für eine detaillierte Beschreibung der sozialen Strukturen und Dynamiken im Usenet siehe Turner u. a. (2005).

12 Siehe Security Focus (2006a).

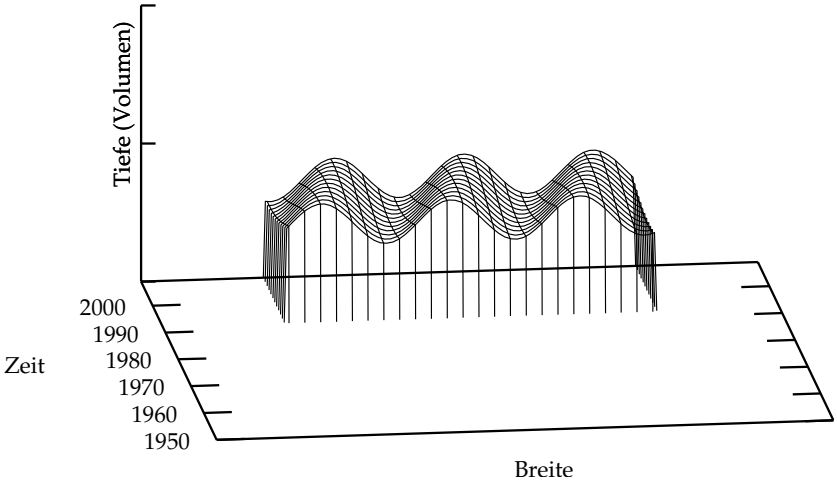


Abbildung 5: Visualisierung der Charakteristik der Beiträge in der Newsgroup `comp.security.misc`.

Full-Disclosure: „Any information pertaining to vulnerabilities is acceptable, for instance announcement and discussion thereof, exploit techniques and code, related tools and papers, and other useful information.“[13]. Obwohl die Themendefinition von Full-Disclosure weiter als die von BugTraq gefasst ist, finden doch auch dort hauptsächlich Bekanntgaben von Sicherheitslücken statt.

Sprachlich und stilistisch befinden sich die Beiträge in Mailinglisten auf einem ähnlichen Niveau wie Newsgroup-Beiträge. Auf Rechtschreibfehler und schlechtes Signal-Rauschen-Verhältnis muss geachtet werden.

2.2.3.1 *Security-Basics*

Im Gegensatz zu BugTraq und Full-Disclosure gibt es die moderierte Mailingliste „Security-Basics“ der Firma Security Focus, welche auch die BugTraq Mailingliste hostet. Auf dieser werden laut Charta ganz verschiedene Sicherheitsthemen von Fachleuten und Anfängern gemeinsam diskutiert[14]. Über das Archiv auf `insecure.org` steht eine Sammlung der Beiträge seit dem Jahrgang 2002 mit insgesamt 35.000 Nachrichten zur Verfügung. Die Charakteristik der Mailingliste ist in Abbildung 6 visualisiert.

2.2.3.2 *Infosec-News*

Eine andere thematisch breit gestreute Mailingliste ist „Infosec-News“[15]. Über diese Mailingliste werden von einigen freiwilligen Redakteuren, Nachrichten zum Thema Informationssicherheit verbreitet. Die Charta meint dazu: „This list contains: Articles catering to such topics as security, hacking, firewalls, new security encryption, forensics, critical infrastructure protection, products, public hacks, hoaxes and legislation affecting these topics [...]“. Es ist also eine breite Themenvielfalt vorgesehen. Wiederum über das Archiv auf `insecure.org` stehen 14.000

13 Siehe Cartwright (2009).
14 Siehe Security Focus (2006b).
15 Siehe Knowles (2009).

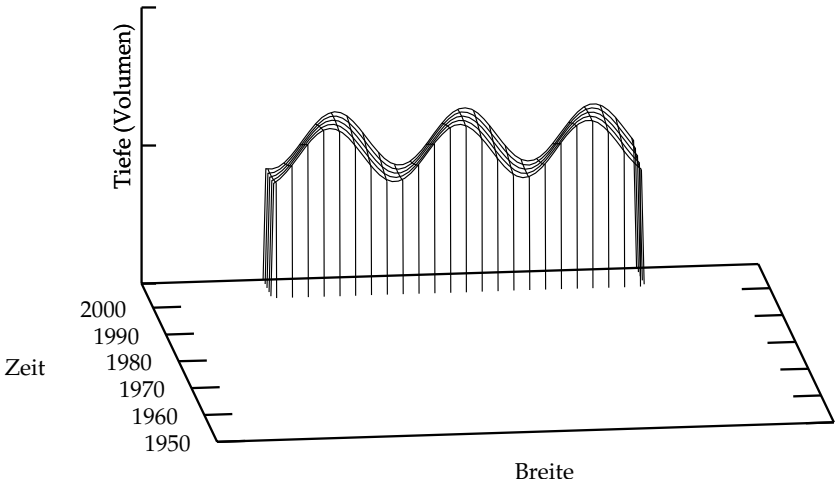


Abbildung 6: Visualisierung der Charakteristik der Beiträge in der Mailingliste Security-Basics.

Nachrichten aus den Jahren 1999 bis heute zur Verfügung. In Abbildung 7 ist die Charakteristik der Mailingliste visualisiert.

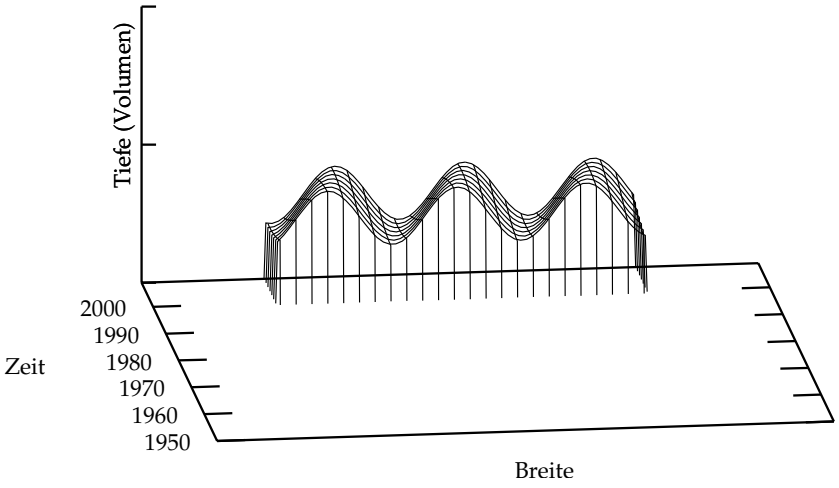


Abbildung 7: Visualisierung der Charakteristik der Beiträge in der Mailingliste Infosec-News.

All diese Datenquellen wurden gesammelt, konsolidiert und in eine zur Verarbeitung geeignete Form gebracht.

3 METHODEN

Es bieten sich viele Möglichkeiten an die Datenquellen zu verarbeiten. Um nicht die Übersicht zu verlieren, kehren wir vorerst zur ursprünglichen Fragestellung zurück: „Wie hat sich der Diskurs über Informationssicherheit verändert?“ oder mit anderen Worten: Wann wurde wieviel, worüber geschrieben?[1]

Diese letztere Frage lässt sich anhand des Datenmaterials beantworten. Oder zumindest steckt die Antwort in den Daten. Der Ansatz der hier gewählt werden soll, ist jeder Nachricht einem Thema zuzuordnen um anschließend einen zeitlichen Verlauf der Publikationshäufigkeit zu jedem Thema auftragen zu können. Die Hoffnung ist herausfinden zu können, welche Schwerpunkte wann im Interesse der Autoren auf dem jeweiligen Medium standen.

Bei mehreren zehntausend Dokumenten ist es ein beachtlicher Aufwand jedes Dokument einem Thema zuzuordnen. Um genau zu sein, ist es überhaupt schon ein Problem eine Menge von Themen zu finden, so dass jedes Dokument passend einem Thema zugewiesen werden kann. Jedoch hat die Mathematik, Informatik und Computerlinguistik in den letzten Jahren dem geneigten Forscher, Entwickler und Anwender ein beachtliches Repertoire an Verfahren und konkreten Algorithmen zur Verfügung gestellt um große Mengen von Texten zu verarbeiten und zu analysieren.

Im Verlauf der Arbeit haben wir einige Algorithmen in Betracht gezogen, ihre Stärken und Schwächen anhand von Sekundärliteratur beurteilt und gegeneinander aufgewogen um letztlich eine Auswahl zu treffen. Auf die verschiedenen Algorithmen gehen wir im Folgenden ein nicht ohne jedoch vorher noch einmal das Problem auf eine saubere Art zu strukturieren.

3.1 THE BIG PICTURE

Ziel der Arbeit ist es den Verlauf des Diskurses über Informationssicherheit zu beschreiben. Wie bereits festgestellt, wurde in den vorhandenen Quellen über das Zielthema auf verschiedene Weisen diskutiert. Der Diskurs über Informationssicherheit umfasst dabei verschiedene Unterthemen denen Nachrichten aus den vorhandenen Quellen mit ihrem Veröffentlichungszeitpunkt zugeordnet werden können. Auf diese Weise wäre man in der Lage die zeitliche Entwicklung verschiedener Bereiche und Teilthemen innerhalb der Informationssicherheit zu verfolgen. Man benötigt also zwei Verfahren. So müssen ersteinmal

1 In der Tat steckt in dieser Umformulierung schon eine implizite Definition von „Diskurs“, die natürlich keiner Überprüfungen an den Theorien von Foucault oder Habermas standhalten kann (für einen Überblick siehe dazu Angermüller 2001). Wenn hier von der quantitativen Dimension des Diskurses gesprochen wird, so entspricht dies zwar wohl eher dem Modell von Jürgen Habermas. In seiner „Theorie des kommunikativen Handelns“ würden die hier untersuchten Daten wohl hauptsächlich dem kommunikativen Handeln (im Gegensatz zum strategischen Handeln) entsprechen (zur Unterscheidung siehe Schrage 1999). Die genaue Untersuchung oder gar eine foucaultsche Diskursanalyse gehört jedoch nicht in die Arbeit zweier Informatiker und so beschränken wir uns hier mit der eingangs erwähnten Betrachtung der Breite und dem Volumen eines Diskurses über die Zeit.

Themengebiete aus den vorhandenen Quellen extrahiert werden. Dabei muss entschieden werden, wieviele Themen eigentlich untersucht werden sollen und welchen Inhalt sie haben oder anders ausgedrückt: Mit welcher Auflösung soll das Gebiet „Informationssicherheit“ eigentlich in Unterthemen aufgeteilt werden? In einem weiteren Schritt müssen dann die verschiedenen Beiträge und Nachrichten aus den Textquellen den Themen noch zugeordnet werden. Es muss also eine Methode gefunden werden, welche das Finden der Themen ermöglicht und dabei die verschiedenen Nachrichten diesen Themen automatisiert zuordnet.

Das Problem der Abbildung von Nachrichten auf Themen lässt sich auch etwas mathematischer ausdrücken. Gesucht ist nämlich im Grunde genommen eine Menge von (wie auch immer gearteten) Regeln, welche jede Nachricht einem bestimmten Thema zuweist. Eine solche Regelmenge kann auch als Modell bezeichnet werden. Es wäre denkbar eine Regelmenge zu definieren, die lediglich jede Nachricht auf das gleiche Thema abbildet, was für die Lösung unserer Fragestellung allerdings nicht sehr hilfreich sein wird. Es wird daher vielmehr ein Modell gesucht, welches jede bekannte Nachricht auf genau das generelle Thema abbildet, welches in der Nachricht behandelt wird.

Das Finden solcher Modelle oder Regelmengen gehört zum Aufgabengebiet des maschinellen Lernens. Dabei werden Algorithmen zur Verfügung gestellt, welche die gewünschten Regelmengen aus dem Gesamtraum der möglichen Regeln suchen[2]. Diese Regelmengen sollten jedoch nicht zu einfach sein, also zu stark verallgemeinern. Stellen wir uns vor, ein maschinelles Lernverfahren würde sämtliche Zeitungsartikel aus dem Politikteil einer Kategorie zuordnen, so wäre das Modell nicht in der Lage zwischen Regionalpolitik und Weltpolitik zu unterscheiden. Die Regelmengen verallgemeinern also stärker als uns lieb ist. Auf der anderen Seite ist es aber auch möglich, dass das Modell zu detailliert ist und stattdessen jedem Artikel aus dem Politikteil einer eigenen Kategorie zuordnet, da ja jeder Artikel schließlich auch ein anderes Thema beschreibt. Dieses Modell wäre zu detailliert oder zu streng und es wäre nicht in der Lage bisher unbekannte Zeitungsartikel eine der vorhandenen Kategorien zuzuordnen[3].

Anstelle von Zeitungsartikeln werden bei uns nun Texte aus wissenschaftlichen Veröffentlichungen, Mailinglisten und Newsgroup-Beiträgen betrachtet und unsere Modelle sollen diese Texte auf verschiedene Kategorien oder Unterthemen des Themas Informationssicherheit abbilden. Dabei muss für jede Textquelle ein eigenes Modell erstellt werden. Diese Textquellen werden auch häufig als Text-Corpus bezeichnet oder einfach nur Corpus, gemeint ist damit die Gesamtheit aller zur Verfügung stehenden Texte einer Quelle, die untersucht werden sollen. Diese liegen meist schon in einem digitalisierten, maschinenlesbaren Format vor. Die einzelnen Texte innerhalb eines solchen Corpus werden auch als Dokumente bezeichnet.

In den folgenden Absätzen wird ein Überblick geboten, welche grundlegenden Strategien es zum Erstellen von Modellen zur Kategorisierung von Textdaten gibt und auf welche Art Textdaten verarbeitet werden müssen, damit maschinelle Lernverfahren damit umgehen können. Anschließend wird auch auf die Visualisierung der Ergebnisse eingegangen.

2 Dieses abstrakte Verständnis von maschinellem Lernen findet sich in Witten u. Frank (2005, Kap. 1.5).

3 Das Problem eines Modells zur starken Verallgemeinerung oder zur starken Detailliertheit zu neigen wird als *Underfitting* bzw. *Overfitting* bezeichnet.

3.1.1 *Clustering und Klassifikation*

Benutzt man maschinelle Lernverfahren zum Erstellen von Modellen, so gibt es zwei Strategien, die gefahren werden können: das überwachte Lernen und das unüberwachte Lernen[4]. Werden die gewünschten Regelmengen mittels eines überwachten Lernverfahrens ermittelt, so spricht man auch von Klassifikation, im Falle von unüberwachten Verfahren von Clustern.

Bei der Klassifikation, also dem überwachten Lernen, kann auf einen Trainer (meist menschlich) zurückgegriffen werden. Dieser Trainer erzeugt sog. Trainingsdaten, die Abbildungen von Nachrichten auf Kategorien darstellen. Diese werden genutzt um das Modell zu erstellen und die Parameter anhand der Daten während des Trainings beispielhaft zu lernen. Anschließend können die restlichen Nachrichten, die nicht im Trainingsschritt verwendet wurden und deren Zugehörigkeit zu den Kategorien nicht bekannt ist, durch das trainierte Modell einer Kategorie zugeordnet werden[5].

Beim unüberwachten Lernen stehen keinerlei Trainingsdaten zur Verfügung. Die entsprechenden Algorithmen müssen einzig aus der internen Struktur der gegebenen Daten die Kategorien extrahieren. Cluster-Algorithmen finden dabei Mengen von Eingabedaten die sich untereinander ähnlich sind und fassen sie jeweils in eine Klasse zusammen. Sie Lösen also beide Probleme, das Finden der Klassen und das Abbilden der Eingabedaten auf Klassen, mit einem Schlag.

Es existieren eine Vielzahl an Cluster-Algorithmen[6]. Den meisten von ihnen ist gemein, dass sie die Ähnlichkeit zwischen Dokumenten zumindest über ein abstraktes Distanzmaß bestimmen oder jedes Eingabedatum als einen Punkt in einem Vektorraum auffassen. Die Funktion des Algorithmus zielt dann darauf ab Punktwolken zu finden. In Abbildung 8 ist beispielhaft ein zweidimensionaler Vektorraum mit Punkten und ein mögliches Clustering dargestellt . Dabei wurde der DBSCAN-Algorithmus aus Kapitel 3.2.3.2 verwendet, weshalb einige Punkte (sog. Noise-Points) keinem Cluster zugewiesen wurden.

Die Auswahl der Cluster kann z. B. nach Dichte der Punktewolken, Nähe innerhalb eines bestimmten Radius oder anderen Merkmalen erfolgen. Auf einige Varianten gehen wir im Abschnitt 3.2.3 genauer ein. Wichtig dabei ist, dass einem Cluster-Algorithmus nicht vorgegeben wird welche Unterthemen zu erwarten sind, er findet diese selbstständig. Klassifikationsverfahren hingegen besitzen ein internes, trainiertes Modell des Datenraums und können Texte anhand spezfischer Merkmale mit Hilfe dieses Modells in eine bereits bekannte Kategorie einordnen. Für das Text-Mining interessante Klassifikationsverfahren werden in Abschnitt 3.2.2 vorgestellt. In beiden Ansätzen wird meistens der Text als numerischer Vektor dargestellt. Um aber von einem Text zu einem Vektor zu gelangen, muss man jedoch erst noch in einem Vorverarbeitungsschritt einige Transformationen durchführen.

4 Es gibt außerdem noch das verstärkte Lernen, was aber im Text-Mining-Bereich wenig Beachtung findet und deshalb hier nicht betrachtet wird.

5 Streng genommen werden beim Trainingsvorgang dem Klassifikationsalgorithmus Eingabedaten gegeben aus dem dann ein Ergebnis berechnet wird, dieses wird mit dem zu erwartendem Ergebnis des Trainers verglichen. In einem sog. verstärktem Lernverfahren würde der Trainer lediglich dem Lernalgorithmus mitteilen ob sein Ergebnis richtig oder falsch war, nicht aber was die eigentlich zu erwartende Lösung war.

6 Einen Überblick hat uns Witten u. Frank (2005) und Tan u. a. (2005) geboten.

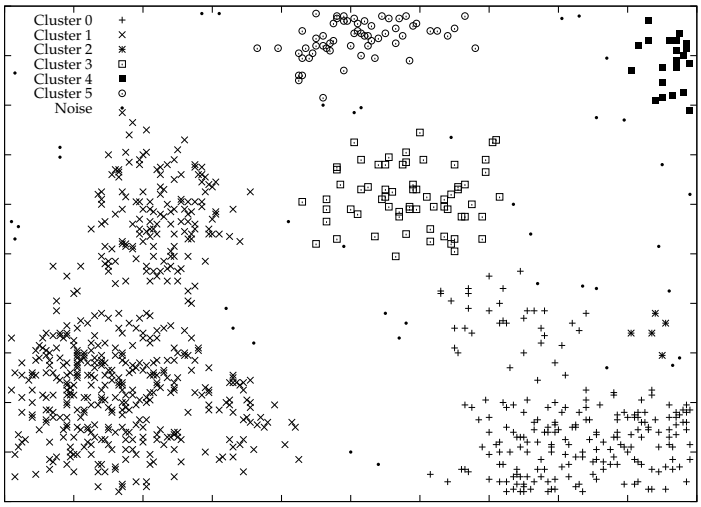


Abbildung 8: Beispiel eines Clusterings mittels DBSCAN

3.1.2 *Vorverarbeitung*

Viele (aber auch nicht alle) Cluster- und Klassifikationsalgorithmen arbeiten auf numerischen Daten. Um diese Algorithmen für Texte anwenden zu können, muss für einen Text daher eine numerische Repräsentation gefunden werden. Sie muss die Eigenschaften, anhand derer die Texte unterschieden werden sollen, geeignet wiedergeben. Reiht man diese numerischen Repräsentationen von Eigenschaften (Features) aneinander, so erhält man einen Vektor – den sog. Feature-Vektor. Die Auswahl der Features hat großen Einfluss auf die Möglichkeiten und Effektivität des Algorithmus. Ein Text enthält ein große Menge an Eigenschaften. So könnten z. B. seine Länge und die Anzahl der unterschiedlichen enthaltenen Wörter ein Merkmal für den „intellektuellen Anspruch“ eines Textes sein. Oder die Häufigkeit der Wiederholung von ähnlichen Wortfolgen könnte ein Maß für inhaltliche Redundanzen im Text darstellen. Mit Hilfe komplexer computerlinguistischer Verfahren könnten auch Aussagen über Satzbau oder sogar Veränderung des Satzbaus über die Länge des Textes als Merkmale einbezogen werden. Der Cluster- oder Klassifikationsalgorithmus kann nur anhand derjenigen Merkmale differenzieren, welche ihm auch dargeboten werden. Es ist nun zu überlegen, welche Merkmale das Thema eines Textes am besten beschreiben – eine durchaus philosophische Frage.

Wir werden hier nicht genauer hinterfragen oder untersuchen können, was denn die Essenz oder das Thema eines Textes nun genau darstellt, denn die uns zur Verfügung stehenden Mittel und Methoden für solche großen Datenmengen sind begrenzt. Die Verfahren der Computerlinguistik stellen sich größtenteils als rechnerisch zu aufwendig heraus um auf größere Datenmengen angewendet werden zu können. Ein gängiger Ansatz ist es daher einzelne Features, also Merkmale, aus dem Text zu extrahieren und unbeachtet der Reihenfolge des Auftretens zu behandeln. Features könnten dabei Begriffe wie „dangerous virus“

oder „internet security“ sein oder einfach nur einzelne Wörter wie „dangerous“, „virus“, „internet“ und „security“. Verwendet man einzelne Wörter als Features so wird auch von dem *Bag-of-Words*-Ansatz gesprochen. Dies ist ein sehr beliebter Weg um Texte darzustellen und wird auch von uns beschritten. In Abschnitt 3.2.1 gehen wir genauer auf die möglichen Wege zum Einteilen eines Textes in einzelne Features ein.

Besitzt man nun eine Liste von Merkmalen für die einzelnen Nachrichten, lassen sich daraus auch numerische Werte generieren. Nehmen wir an, es wird der Bag-of-Word-Ansatz verwendet, dann kann nun jedem Wort aus der Gesamtheit aller Texte einer Quelle (Corpus) eine Position in einem Vektor zugewiesen werden. Für jeden Text wird dann der Zahlenwert eines jeden Wortes ermittelt, dies kann eine 1 für „Wort ist im Text vorhanden“ sein oder 0 für „Wort ist nicht vorhanden“ oder auch die relative Häufigkeit des Wortes zur Gesamtlänge des Textes und in den Vektor eingetragen. Die so transformierten Texte stellen als eine Menge von Spalten-Vektoren dann auch eine Matrix dar – die Feature-Matrix. Da die Reihen der Matrix jeweils für Wörter oder Begriffe stehen und die Spalten für die Dokumente, wird häufig auch der Begriff Term-Document-Matrix verwendet.

Prinzipiell kann auf der Menge von Vektoren bereits das Cluster- oder Klassifikationsverfahren angewendet werden. Leider wird sich dabei meist noch kein gutes Ergebnis einstellen, denn Text-Daten sind stark verrauscht und viele der Wörter sind alleinstehend nicht bedeutungstragend (z. B. und, vielleicht, wir, der, die, das, ...). Rechtschreib- oder OCR-Fehler tun ihr übriges um die Dimensionalität der Vektoren, also die Anzahl der verfügbaren Textmerkmale in die Höhe zu treiben. Dieses Problem wird auch als „The Curse of Dimensionality“[7] beschrieben. Die Lösung dazu ist nur bestimmte, wichtige und charakteristische Features aus der verfügbaren Menge auszuwählen. Dieser zwischengeschaltete Vorgang wird als „Feature-Selection“ oder auch „Attribute selection“ bezeichnet[8]. Die beste Art und Weise dies zu tun, wäre die handverlesene Auswahl der wichtigsten Merkmale anhand von menschlichem Wissen über die Daten. Das dahinterliegende Problem kann aber auch als Lernaufgabe für maschinelles Lernen verstanden werden[9]. Neben der reinen Auswahl kommen auch Transformationen der Daten als Vorbereitung in Frage. Die Feature-Matrix des Corpus kann durch Operationen der linearen Algebra in Formen gebracht werden, in denen z. B. Korrelationen zwischen mehreren Zeilen oder Zeilen mit niedriger Varianz eliminiert werden und die damit besser zum Clustern oder Klassifizieren geeignet sind. Manche statistische Lernverfahren arbeiten nicht auf gebrochenen Zahlen (relative Häufigkeiten), sie erwarten diskrete Werte oder gar keine Vektoren sondern Mengen von vorhandenen Merkmalen (auf die tatsächliche Abwesenheit von nicht-vorhandenen Merkmale wird automatisch gefolgert). Liegen bereits Kennzahlen in gebrochenen Zahlen vor, müssen diese erst wieder diskretisiert werden bzw. in die geeigneten Merkmalsformate umgewandelt werden.

Meist lassen sich zudem die Algorithmen noch durch Parameter beeinflussen. Die spezifische Parametrisierung hat dann einen starken Einfluss auf die Lernergebnisse der Verfahren. Wiederholt man die

7 Dieser Begriff wurde von Richard Bellman geprägt, um die ungewöhnlichen, mathematischen Eigenschaften hochdimensionaler Räume zu beschreiben, welche Statistikern und Mathematikern häufig Schwierigkeiten bereiten.

8 Siehe dazuWitten u. Frank (2005, Kap. 7.1).

9 Vgl. ebenda, S. 289.

Verfahren mit verschiedenen Parametern und vergleicht die Ergebnisse anhand festzulegender Kriterien, so ergibt sich dann hoffentlich ein gutes Ergebnis. Formal kann die Parametrisierung als ein Optimierungsproblem begriffen werden. Eine konkrete Formulierung z. B. als lineares Optimierungsproblem gibt es für die meisten Algorithmen jedoch nicht.

3.1.3 *Visualisierung der Ergebnisse*

Die Ergebnisse sind Zuordnungen von Nachrichten zu Kategorien, wobei jede Kategorie ein inhaltliches Thema besitzt. Wurden die Kategorien wie beim Klassifizieren manuell festgelegt, so ist der Inhalt im voraus bekannt und auch ein Name für das Thema kann in der Regel leicht vergeben werden bzw. steht schon fest. Wurden die Kategorien jedoch durch ein unüberwachtes Lernverfahren ermittelt, so tragen sie keine Namen und auch die Inhalte sind unbekannt. Um das Thema zu einem Cluster zu ermitteln, könnte nun ein Mensch alle enthaltenen Dokumente lesen um daraus das Themengebiet zu erfassen. Bei einer hohen Anzahl von Dokumenten lässt sich dies allerdings nicht mehr zuverlässig realisieren. Die automatische Generierung einer Zusammenfassung von mehreren Dokumenten kann hier Abhilfe schaffen. So muss der menschliche Betrachter nur noch einen Teil der Information betrachten und kann daraus einen Namen generieren. Wie zu erwarten, verbirgt sich hinter diesem Problem wieder eine ganze Forschungslandschaft, die sich mit dem Problem der sog. „Text-Summarization" befasst.

Stehen die Namen für die Themen der Kategorien fest, so kann mit deren Visualisierung in Diagrammen begonnen werden. Es bieten sich verschiedene Darstellungen der Häufigkeitsverteilung über die Zeit an, die alle mit verschiedenen kognitiven Charakteristik behaftet sind[10]. Um jedoch zu einer solchen finalen Visualisierung zu kommen, muss erst ein konkretes Verfahren aus den verschiedenen, möglichen Ansätzen gewählt werden.

3.2 MÖGLICHE WEGE

Während im vorangegangenen Abschnitt der grobe Ablauf und der Lösungsraum für unser Problem skizziert wurde, werden nun spezifische Verfahren betrachtet und ihre jeweiligen Vorteile und Nachteile sowie gegenseitige Abhängigkeiten kurz dargestellt. Begonnen wird mit dem großen Thema der Datenvorverarbeitung in Abschnitt 3.2.1, ein wichtiger Schritt, in dem unstrukturierte Texte in numerische Daten umgewandelt werden, um mit ihnen rechnen zu können. Anschließend wird in den darauffolgenden Abschnitten detailliert auf maschinelle Lernverfahren eingegangen, die für Text-Mining-Anwendungen in Frage kommen. Klassifikationsverfahren finden sich in 3.2.2 und Cluster-Verfahren in Abschnitt 3.2.3. Zum Ende werden dann verschiedene Methoden zur Auswertung der Klassifikations-, bzw. Cluster-Ergebnisse verglichen, die in Abschnitt 3.2.4 nachzulesen sind.

10 Dazu z. B. die detaillierte Untersuchung einer aufsummierten Darstellung in Havre u. a. (2000).

3.2.1 *Vorverarbeitung der Textdaten*

Für das menschliche Gehirn ist es einfach Textdaten zu erfassen, den Inhalt zu verstehen und verschiedene Dokumente entsprechenden Themen zuzuordnen. Computer hingegen sind auf die Verarbeitung von Zahlen und abstrakten Datentypen spezialisiert. Bevor also die Dokumente von einem Programm verarbeitet und kategorisiert werden können, müssen diese in eine für den Computer lesbare Form überführt werden.

Wie bereits weiter oben erwähnt, arbeitet der Großteil aller Algorithmen zum Kategorisieren von Daten auf Vektoren, also Punkten in einem mehrdimensionalen Datenraum. Die Dokumente, die kategorisiert werden sollen, liegen allerdings nur als Textdaten vor. Bevor die Dokumente analysiert werden können, müssen diese in einem Vorverarbeitungsschritt in ein maschinenlesbares Format gebracht werden. Dieses Kapitel beschreibt verschiedene Verfahren wie Textdokumente in numerische Werte umgewandelt werden können, so dass man diese mit maschinellen Lernverfahren in Themengebiete aufteilen kann. Es muss eine Abbildung gefunden werden, welche den Text eines Dokuments auf einen Vektor abbildet. Dies ist ein wohlbekanntes Text-Mining Problem mit vielen, gut dokumentierten Lösungen, im Folgenden sollen die für diese Arbeit interessantesten Ansätze vorgestellt und diskutiert werden. Wurden die Textdaten erfolgreich umgewandelt, erhält man für jedes Dokument der Textquelle einen Feature-Vektor. Die Feature-Vektoren können dann als Spaltenvektoren einer Matrix dargestellt werden. Diese Feature-Matrix könnte weiterhin noch zur Steigerung der Ergebnisqualität mit Verfahren aus der linearen Algebra speziellen Transformationen unterzogen werden.

3.2.1.1 *Tokenization*

Für Computer stellen Textdaten lediglich eine Folge von Zeichen/Bytes ohne jegliche semantische Bedeutung dar. Der erste Schritt um einen beliebigen Text in eine aussagekräftige, numerische Darstellung zu überführen besteht darin, die Zeichen zu sinnvollen Informationseinheiten zusammen zu fassen. Aus diesen Informationseinheiten werden dann Textmerkmale generiert, welche das Dokument repräsentieren. Häufig wird dabei auf die Reihenfolge der Merkmale innerhalb des Textes keine besondere Rücksicht genommen.

Der klassische Ansatz ist dabei einzelne Wörter im Text zu so genannten Tokens zusammen zu fassen[11]. Jedes Token stellt dabei ein Feature da, welches den Inhalt oder thematischen Bezug eines Textes repräsentiert.

Für Menschen ist es wiederum recht einfach einzelne Wörter in einem Text zu erkennen, für Computer kann dies eine schwierige Aufgabe werden, abhängig von dem Grad der Genauigkeit, die der Benutzer wünscht und der Sprache in der sich der Text befindet. Für englische Texte, wie sie in dieser Arbeit vorkommen, ist es sinnvoll alle Arten von sog. *Whitespaces*, also Zeichen wie Zeilenumbruch, *Carriage-Return*[12] Tabulator- oder Leerzeichen, sowie die Sonderzeichen <>();!?“ als Trenn-

11 Dieser Vorgang wird häufig als „Tokenizing“ oder „Tokenization“ bezeichnet, im deutschen Sprachraum wird teilweise auch von „Tokenisierung“ gesprochen, im Rahmen dieser Arbeit wird durchgehend der englische Fachbegriff „Tokenization“ benutzt.

12 Wagenrücklauf, markiert den Zeilenumbruch oder die Rückführung des Cursors an den Zeilenanfang in digitalen Textdokumenten.

zeichen für Tokens zu verwenden, sowie die Zeichen .,:-' abhängig von ihrem Anwendungskontext. Es ist zu beachten, dass ein Punkt und Komma innerhalb einer Zahl keine Trennung darstellt und der Punkt auch Teil einer Abkürzung sein kann. Desweiteren muss bei der Tokenization entschieden werden, ob Zahlen, Bindestrichwörter und ähnliche Grenzfälle mitgezählt werden sollen oder verworfen werden. Die Entscheidung wie genau der Text eingeteilt werden soll, hängt davon ab was genau mit dem Text gemacht werden soll und auf welches Anwendungsgebiet er sich erstreckt. Ist man z. B. an Preisangaben innerhalb eines Textes interessiert, so dürfen Punkte und Kommas nicht nur als Satzzeichen aufgefasst werden, sondern auch als Bestandteil einer Zahl.

Ein in Tokens eingeteiltes Textdokument ist nun eine Ansammlung von Wörtern. Jedes Token stellt eine Instanz eines Features da. Taucht also in einem Text das Wort „electronic" dreimal auf, so gibt es ein Feature „electronic" mit drei Instanzen. Die Gesamtheit aller Features einer Tokenmenge nennt man „(lokales) Dictionary". Dementsprechend wird die Gesamtheit aller Features im gesamten Corpus, also aller Textdokumente einer Quelle, „globales Dictionary" genannt. In den meisten Fällen besitzt das Dictionary eines Dokuments nach der Tokenization sehr viele Features, welche reduziert werden können ohne das wesentliche Informationen verloren gehen.

Dieses klassische Vorgehen einen Textes in einzelne Wörter aufzuteilen dient dazu, diesen in kleine, aussagekräftige Informationseinheiten einzuteilen[13]. Neben dem klassischen, eben beschrieben Verfahren gibt es allerdings noch einige weitere alternative Ansätze zur Tokenization, die andere Unterteilungen in Informationseinheiten vornehmen.

N-GRAMM Eine sehr radikale Alternative zur klassischen Tokenization von Texten ist es den Text nicht nach Wörtern, sondern nach N-Grammen zu strukturieren. Ein N-Gramm ist eine Sequenz aus n verschiedenen Zeichen eines Alphabets. Ein Bigramm ist also eine beliebige Kombination aus zwei Zeichen die in einem Text vorkommt, ein Trigramm eine Kombination aus drei Zeichen und so weiter. Würde man die Zeichenkette „text" in Bigramme unterteilen, so erhielte man als Token „te", „ex", „xt". Der Text besteht im Anschluss nicht mehr aus Worten sondern einzig aus einer Menge von solchen Buchstabenbröckchen. Erstaunlicherweise lassen sich selbst mit diesen Informationseinheiten noch Ergebnisse bei der Kategorisierung von Texten erzielen[14].

Werden Texte in N-Gramm Tokens zerlegt, so hat dies den Vorteil, dass Rechtschreibfehler weniger ins Gewicht fallen und man sprachunabhängiger ist, also mehrsprachige Corpora untersuchen kann[15]. Der Nachteil ist allerdings, dass sehr viele Kombinationen von N-Grammen in Dokumenten vorhanden sind, es somit einen sehr hochdimensionalen Feature-Raum geben wird. Ein weitere Nachteil ist die Schwierigkeit für Menschen anhand einer N-Gramm-Menge auf den ursprünglichen Inhalt eines Dokuments zu schließen.

MULTI-WORD TERM EXTRACTION Häufig ist die Zerlegung von Dokumenten in einzelne Wörter zu ungenau, da dadurch zusammengehörige Wörter getrennt werden. Besonders Suchmaschinen im Internet

13 Dieser Ansatz zur Tokenization wird bei Weiss u. a. (2005) ausführlich vorgestellt.
14 Náther (2005) verwendet dieses Verfahren u. a. um Dokumente zu clustern.
15 So stellen Cavnar u. Trenkle (1994) ein auf N-Grammen basiertes Verfahren vor um mehrsprachige Newsgroupbeiträge zu klassifizieren.

lassen oft auch längere Tokens zu, man spricht dann von *Multi-Terms* oder *Multi-Word Features*. Typische Beispiele für Multi-Word Features sind Begriffe wie *Federal Republic of Germany*, *search engine* oder *text mining*. In diesen Beispielen bestehen die Begriffe aus mehreren Nomen, es ist aber auch denkbar Adjektive als beschreibende Elemente für Nomen zuzulassen, dann könnte auch *fast algorithm* als Multi-Word Feature aufgefasst werden. Folgt man dieser Idee Multi-Terms anhand ihrer grammatikalischen Komponenten zu identifizieren, muss darauf geachtet werden, dass z. B. Adjektive nicht unbedingt immer direkt vor dem Nomen stehen müssen. So wäre es denkbar *fast and efficient algorithm* und *fast algorithm* als die Ausprägung desselben Features aufzufassen. Es warten also viele Fallstricke und Mehrdeutigkeiten in der menschlichen Sprache auf die Algorithmen. In jedem Fall müssen die *Multi-Terms* jedoch in einem ersten Schritt von einem Algorithmus aus dem Corpus extrahiert werden. Es bietet sich wiederum eine ganzes Forschungsgebiet zur Recherche an[16]. Besonders im Bereich des Information Retrievals, also der Forschung im Bereich der Suchmaschinen, werden *Multi-Terms* zum speichereffizienten Indizieren von Texten verwendet und dementsprechend liefert auch dort die Forschung viele Beiträge.

Die Spannweite der Algorithmen reicht von rein statistischen und damit meist sprachunabhängigen Verfahren über rein linguistisch arbeitende Verfahren bis hin zu ausgefeilten Hybriden mit Unterstützung von maschinellen Lernverfahren. Jedoch sind bis auf wenige lobenswerte Ausnahmen keine Implementierungen von diesen Systemen verfügbar, erst recht nicht in der von uns gewählten Sprachumgebung. Der Vorteil von Multi-Word Features liegt ganz klar in ihrem starken Informationsgehalt, der Nachteil hingegen ist der enorm erhöhte rechnerische Aufwand mit der Texte vorverarbeitet werden müssen. Systeme die linguistisch arbeiten, benötigen eine aufwendige lexikalische Analyse der Texte, Hybride setzen teilweise auf Trainings-Corpora oder einen großen Thesaurus zur Unterstützung[17]. Der Implementierungsaufwand wäre zudem unumgänglich und sehr hoch.

3.2.1.2 *Stop-Words entfernen*

Dokumente die bereits in Tokens aufgeteilt wurden beinhalten häufig Features, die keinerlei Aussage über den Inhalt des Dokuments machen. Dazu zählen vor allem diejenigen Wörter, die in der Sprache in welcher die Dokumente geschrieben sind, besonders häufig auftreten. Für die englische Sprache sind das vor allem bestimmte und unbestimmte Artikel, Präpositionen, Hilfsverben, Konjunktionen und Pronomen. Es können aber auch beliebig andere Wörter mit aufgenommen werden, von dem der Nutzer weiß, dass diese keinen Informationsgehalt für die Darstellung der Dokumente haben[18].

Features die ein solches Wort repräsentieren, können normalerweise aus dem Dictionary eines Dokuments gelöscht werden. Da diese Wörter bei der Vorverarbeitung nicht weiter betrachtet werden, spricht man auch von Stop-Words. Auf der anderen Seite gibt es einige Wörter

16 Einen Überblick hat uns Jacquemin u. Bourigault (2003) und Castellví u. a. (2001) geboten.

17 z. B. setzt das System KEA++ (eines der wenigen mit verfügbarer Implementierung) auf einen kontrollierten, domänenspezifischen Wortschatz (vgl. Witten. u. a. 1999; Frank u. a. 1999).

18 Man spricht in diesem Fall auch von *domain specific stop words*, da sich diese meist nur auf sehr spezielle Arten von Texten beziehen.

die zwar auf den ersten Blick wie Stop-Words aussehen, aber unter Umständen keine sein müssen. So werden im Englischen die Verben „can“ und „may“ häufig als Stop-Words behandelt, können aber auch als Nomen aufgefasst werden, nämlich als „Kanne“ und „Mai“. Ob diese Begriffe verworfen werden sollen oder nicht, hängt häufig von dem thematischen Bezug der zu verarbeitenden Texte ab.

Stop-Word-Listen werden daher nicht maschinell, sondern per Hand erstellt und können individuell auf den eigenen Datensatz angepasst sein, aus diesem Grund gibt es auch keine offiziellen Stop-Word-Listen für bestimmte Sprachen.

3.2.1.3 *Stemming*

Nachdem der Text in verschiedene Tokens aufgeteilt wurde, wird es sehr wahrscheinlich vorkommen, das gleiche Wörter in verschiedenen Konjugationen oder Deklinationen vorliegen. Da in einer unsortierten Tokenmenge diese zusätzlichen grammatikalischen Informationen kaum einen Mehrwert darstellen, gibt es die Möglichkeit solche Wörter auf ihren Wortstamm zu reduzieren und damit die Tokenmenge zu vereinfachen ohne großen Informationsverlust in Kauf nehmen zu müssen.

Den Vorgang ein Wort auf seinen Stamm zu reduzieren wird „Stemming“ bezeichnet. Stemming-Algorithmen sind selbstverständlich sprachabhängig, da die jeweiligen Regeln einer Sprache zum Beugen von Verben und Nomen sich stark unterscheiden können. Ein Beispiel für die englische Sprache sind die Wörter „fishing“, „fishes“, „fisher“, welcher alle zu dem Wort „fish“ reduziert werden können. In dem Beispiel wird auch deutlich, das durchaus Information verloren gehen kann. Das Wort „fish“ beschreibt schließlich ein Tier, hingegen „fisher“ einen Beruf. Wenn allerdings nach dem Stemmingvorgang im Dokument nur der Begriff „fish“ zu finden ist, dann lässt sich daraus lediglich ableiten, dass der Text u.a. über Fische handelt, nicht aber auch über Fischer. In den meisten Fällen mag eine solche Vereinfachung vernachlässigbar sein, es gibt aber Anwendungsfälle wo differenzierter unterschieden werden muss ob und wie gestemmt wird. Aus diesem Grund haben sich verschiedene Techniken für das Stemmen von Wörtern entwickelt.

PORTER-STEMMER Für die englische Sprache wird häufig der weit verbreitete Porter-Stemmer-Algorithmus genutzt, welcher sehr gute Ergebnisse erzielt[19]. Der Porter-Stemmer macht sich zu Nutzen, dass sich viele Suffixe im Englischen aus mehreren kleinen Suffixen zusammensetzen. Auf diese Weise kann der Porter-Stemmer Schritt für Schritt das Wort verkleinern. Der Porter-Stemmer nutzt dabei fünf Schritte, die jeweils einen Katalog an Regeln besitzen. Jede Regel legt dabei fest, welche Eigenschaften ein Wort haben muss, also z.B. wieviele Vokal-Konsonanten-Paare auf einen Konsonanten folgen dürfen und auf welchem Suffix das Wort enden soll um diesen mit einer anderen Endung zu ersetzen oder ganz zu entfernen. In jedem Schritt wird geprüft, ob eine Regel auf das gegebene Wort angewendet werden

19 Ein Vergleich zwischen verschiedenen Stemmern findet sich in Fuller u. Zobel (1998), wobei der Porter-Stemmer als klarer Sieger hervorgeht.

kann. Ist dies der Fall, wird das Wort entsprechend geändert und dem nächstem Verarbeitungsschritt zugeführt[20].

Soll zum Beispiel das Wort „relational“ gestemmt werden, zählt der Algorithmus zuerst die Anzahl der Vokal-Konsonanten-Paare. Alle Wörter in der englischen Sprache bestehen aus diesen Paaren. Sei C eine beliebig lange Sequenz von Konsonanten und V eine ebenfalls beliebig lange Sequenz aus Vokalen, so existieren in jedem englischen Wort eine bestimmte Anzahl von VC-Paaren. Alle englischen Wörter lassen sich also wie folgt darstellen:

$$[C](VC)^m[V] \tag{3.1}$$

Die eckigen Klammern stellen dabei optionale Wortkomponenten dar, die Vokal-Konsonant-Paare werden m-mal wiederholt. Für das Beispiel „relational“ heißt das also, dass $m = 4$ ist, nämlich *EL*, *AT*, *ION* und *AL*. Der Algorithmus versucht nun verschiedene Regeln auf das Wort anzuwenden. Dabei gibt es zwei Regeln, die für dieses Wort in Frage kommen. In Schritt 2 des Algorithmus werden Wörter mit $m > 0$ und Wörter die auf das Suffix *ATIONAL* enden so geändert, das *ATIONAL* mit *ATE* ersetzt wird. Aus „relational“ wird nun also „relate“ mit $m = 2$ (*EL*, *AT*). In Schritt 4 werden dann von allen Wörtern mit $m > 1$ das Suffix *ATE* entfernt, soweit dies vorhanden ist. Aus „relate“ wird also „relat“, hingegen das Wort „rate“ würde nicht geändert werden, da $m = 1$ ist, es also nur ein Vokal-Konsonant-Paar gibt. In den übrigen Schritten können keine Regeln auf das Wort angewendet werden, deshalb werden diese Verarbeitungsschritte übersprungen.

Das Ergebnis des Porter-Stemmer ist also nicht immer ein lexikalisch-korrektes Wort, sondern lediglich sein Wortstamm. So werden die Wörtern „compute“, „computer“ und „computation“ in jedem Fall auf den Stamm „comput“ zurückgeführt.

LEMMATISATION Als Lemmatisation wird der Vorgang bezeichnet mit dem Wörter auch ihren morphologischen Wortstamm zurückgeführt werden. Normalerweise wird von Stemming-Verfahren nicht verlangt, dass das Ergebnis ein korrektes Wort ist, solange verschiedene Wörter auf dasselbe Ergebnis abgebildet werden. Bei einer Lemmatisation wird allerdings, wie der Name schon verraten lässt, auf das Lemma eines Wortes geschlossen, also die Grundform eines Wortes wie man es auch in einem Wörterbuch finden würde. Für Nomen gilt so z. B. dass sie in den Nominativ Singular überführt werden und Verben werden auf ihren Infinitiv abgebildet. Dieser Ansatz führt also zu genaueren Ergebnisse als „gewöhnliches“ Stemming[21].

Angewendet auf das vorherige Beispiel mit „fishing“, „fishes“ und „fisher“ würde das bedeuten, dass abhängig davon, ob „fishing“ ein Nomen oder Verb ist, der Algorithmus das Wort so belässt oder daraus „fish“ macht, also den Infinitiv des Wortes. Hingegen für „fishes“ ist es egal, ob es sich um ein Nomen Plural oder ein Verb in der zweiten Person Singular Präsenz handelt, in beiden Fällen wird das Wort auf „fish“ reduziert.

20 Vgl. Porter (2006) für eine ausführliche Beschreibung des Algorithmus. Zusätzliche Informationen, sowie Implementierungen für verschiedene Programmiersprachen finden sich auf Martin Porters Webseite unter `http://tartarus.org/martin/PorterStemmer`.

21 Ein sehr robustes Verfahren für Lemmatisation von englischsprachigen Texten stellt Minnen u. a. (2001) vor.

Um eine Lemmatisation vorzunehmen benötigt man die Wortartinformation (Part-of-Speech Information) der einzelnen Wörter. Diese erhält man von sog. Part-of-Speech-Taggern (POS-Tagger) spezielle Algorithmen welche Wörter in einem Satz ihre Wortart zuordnen. Leider ist die menschliche Sprache nicht immer eindeutig und besitzt viele Unregelmäßigkeiten, was ein automatisches Erkennen von Wortarten zu einem schwierigen Unterfangen machen kann. Normalerweise sind daher POS-Tagger als überwachte, maschinelle Lernverfahren implementiert, die anhand von speziellen Texten trainiert werden. Das heißt aber auch, dass wenn in einem Text viele Fachbegriffe aus dem Bereich Computersicherheit vorkommen die nicht im Trainingssatz enthalten waren, dann kann das Verfahren unter Umständen Probleme haben diesen Wörtern korrekte Wortarten zuzuordnen.

Der Vorteil von Lemmatisation ist allerdings ein differenziertes Zusammenführen von Wörtern und der Umstand, dass man korrekte Wörter erhält und keine verkürzten Wortstämme. Der Preis dafür ist allerdings hoch. Texte müssen vorher mit einem POS-Tagger analysiert werden, was ein sehr rechenintensives Vorgang ist. Tagger sollten außerdem auf einem passenden Text-Corpus traininert werden, damit sie auch mit Fremd- und Fachwörtern gut umgehen können. Vor dem Einsatz solcher Verfahren muss also genau abgewägt werden, ob der hohe Aufwand sich rechtfertigt. Wird z. B. die Information über die Wortarten nach dem Stemmen nicht mehr benötigt, dann sollte auch auf eine umfassende Lemmatisation verzichtet und auf ein einfacheres Verfahren zurückgegriffen werden.

3.2.1.4 *Rechtschreibkorrektur*

Gerade Dokumente aus Quellen in denen Rechtschreibfehler häufig vorkommen, wie in nicht-wissenschaftlichen oder nicht-journalistischen Quellen wie Webblogs, E-Mails, Newsgroups und anderen „User Generated Content“ oder auch Quellen die OCR-Fehler aufweisen können, kann es vorkommen, dass einige Features nur sehr wenige Instanzen haben. Dabei handelt es sich meistens um falsch geschriebene Wörter. Da diese Wörter für zusätzliches Rauschen sorgen, aber kaum Informationsmehrwert bereithalten, macht es Sinn diese aus der Feature-Menge zu entfernen. Zudem steigt durch die statistisch annehmbare Gleichverteilung der Fehler über die Wörter das Gesamtvokabular des Corpus nun nicht mehr logarithmisch mit der Menge der enthaltenen Wörter (Zipf's Law[22]), sondern annähernd linear. Die Dimensionalität der Feature-Vektoren würde bei großen Corpora also enorm anwachsen.

Um den durch Fehlern verursachten Dimensionalitätszuwachs zu reduzieren, gibt es dabei zwei verschiedene Ansätze. Im ersten Ansatz werden zuerst die Stop-Words entfernt, danach stellen die n häufigsten Wörter eine gute Basis da um Feature-Vektoren darauf aufzubauen und erweisen sich als erstaunlich effektiv (Weiss u. a. 2005). Die Wahl, wieviele häufigste Wörter betrachtet werden sollen, muss allerdings sorgfältig bedacht werden, da auch Wörter, die nur selten im Text vorkommen, eine gute Aussagekraft haben können.

Ein anderer Ansatz ist es deshalb nicht die häufigsten Wörter auszuwählen, sondern die seltensten Wörter zu verwerfen. Taucht ein Feature nur einmal in einem Dokument auf, so steigt die Wahrscheinlichkeit, dass es sich um ein falsch geschriebenes Wort handelt mit der Anzahl

22 Mehr dazu später in Abschnitt 4.2.1.1.

der gesamten Wörter im Text. Selbst wenn dieses Wort richtig geschrieben ist, so wird es sicherlich nicht repräsentativ für die Kernaussage des Textes sein, da es ja nur einmal vorkommt. Aus diesem Grund kann man solche Wörter verwerfen ohne einen großen Informationsverlust in Kauf nehmen zu müssen.

Ein anderer Ansatz ist das Verwenden von Rechtschreibkorrekturen. Die Anzahl der verschiedenen Wörter, also die Dimensionalität des Feature-Vektors wird dadurch zwar auch reduziert, aber nicht durch wegwerfen von Informationen, sondern durch Berichtigen von fehlerhaften Wörtern. Tippfehler in E-Mail-Dokumenten beinhalten häufig verdrehte Buchstabenreihenfolge oder den Anschlag einer falschen Taste auf der Tastatur. Mit einer Rechtschreibkorrektur ließen sich solche Wörter korrigieren. Rechtschreibkorrekturen in Computerprogrammen arbeiten allerdings nicht immer vollständig autonom, in einigen Fällen ist es schwierig zu entscheiden ob ein Wort tatsächlich falsch geschrieben ist oder es sich z. B. lediglich um einen ungewöhnlichen Eigenname handelt. Für Text-Mining gilt aber, dass die Vorverarbeitung der Dokumentendaten vollständig autonom vom Computer erledigt werden sollen, da große Mengen an Daten verarbeitet werden müssen.

Eine mögliche Lösung bestünde darin nur diejenigen Wörter zu korrigieren, bei denen sich der Computer besonders sicher ist, dass es sich um einen Rechtschreibfehler handelt. Eine Variante, gerade für eingescannte Textdokumente ist es, nur nach Fehlern Ausschau zu halten, die typisch für OCR-Programme sind. So sind sich bestimmte Buchstaben des lateinischen Alphabets in ihrer optischen Erscheinung besonders ähnlich (andere Alphabete haben offensichtlich andere Charakteristika). Die Buchstaben „e“ und „c“ oder „o“ und „a“ werden deshalb recht häufig verwechselt. Die genauen Fehlerwahrscheinlichkeiten hängen von der verwendeten OCR Software und den zu erkennenden Schriftarten ab.

OCR-MAPPING Eine einfach zu implementierende und performante Variante ist es, einmalig eine Ersetzungstabelle zu erzeugen. Jedem fehlerhaften Wort wird darin sein wahrscheinlich korrektes Pendant zugeordnet. In der anschließenden Verarbeitung aller Texte, wird dann nach der Tokenization und vor dem Stemming in jedem Dokument für jedes Wort eine eventuelle Ersetzung anhand der Ersetzungstabelle durchgeführt. Diese Operation ist sehr einfach und schnell durchführbar. Interessant ist nun wie die Ersetzungstabelle entsteht.

Wir haben dazu ein einfaches Verfahren entworfen:

1. Lese alle Wörter aus allen Dokumenten, zähle dabei die absolute, globale Häufigkeit jedes Wortes und speichere diese Information in einer Häufigkeitstabelle.

2. Vergleiche nun mit einem zu wählenden Distanzmaß jeweils ein Wort in der Häufigkeitstabelle mit jedem anderen. Wörter die eine zu bestimmende Distanz unterschreiten werden als ähnlich betrachtet.

3. Wähle aus der Menge der ähnlichen Wörter dasjenige aus, welches die höchste absolute Häufigkeit hat.

4. Trage in die resultierende Ersetzungstabelle alle als ähnlich betrachteten Wörter ein. Ihre Ersetzung ist das häufigste dieser Wörter.

5. Entferne die ähnlichen Wörter aus der zu verarbeitenden Häufigkeitstabelle.

6. Wenn die Häufigkeitstabelle noch nicht leer ist, setze das Verfahren mit dem nächsten Wort bei 2. fort.

Das Ergebnis dieses einfachen Algorithmus ist eben jene gesuchte Ersetzungstabelle. Für die genaue Spezifikation des Algorithmus muss nun noch ein Distanzmaß gewählt werden, welches die Ähnlichkeit (bzw. Entfernung) zweier Wörter beschreibt. Dazu bieten sich mehrere Alternativen, die alle Verallgemeinerungen oder Spezialisierungen der sog. Levenshtein-Distanz[23] darstellen. Dabei wird die Distanz zwischen zwei Wörtern durch die minimale Anzahl an Veränderungsoperationen beschrieben die benötigt werden um von einem Wort auf das andere zu gelangen. Die möglichen Veränderungsoperationen sind das Einfügen, Löschen oder Austauschen eines Buchstabens. Die Levenshtein-Distanz ignoriert jedoch die Länge von Wörtern vollständig. So beträgt die Levenshtein-Distanz zwischen den Worten „gehen“ und „sehen“ ebenso 1 wie die Distanz zwischen den Worten „Lehrer“ und „Lehre“. Ebenfalls ignoriert die Levenshtein-Distanz die möglichen höheren semantischen Differenzen bei Veränderungen am Wortanfang als am Wortende. Diese Nachteile greift die sog. Jaro-Winkler Distanz auf[24]. Sie beschreibt im Wertebereich zwischen Null (keine Übereinstimmung) und Eins (komplette Übereinstimmung) die relative Ähnlichkeit zur Wortlänge und gewichtet dabei Unterscheide am Wortanfang stärker als am Wortende. Die Jaro-Winkler Distanz zwischen den Worten „gehen“ und „sehen“ beträgt damit dann ca. 0,8666. Während die Distanz zwischen den Worten „Lehrer“ und „Lehre“ bei 0,9722 liegt. Ein guter Grenzwert für die allgemeine Gleichheit zweier Wörter schien die Distanz 0,985 zu sein. Zusätzlich zur Jaro-Winkler Distanz haben wir ein zweites Distanzmaß verwendet, welches typische OCR-Fehler wie das oben genannte e/c Problem aufgreift und mit der Jaro-Winkler Distanz kombiniert. Ein beispielhafter Ausschnitt einer sich ergebenden Ersetzungstabelle ist in Tabelle 1 dargestellt.

Fehlerwort	Ersetzung
youngster	youngsters
writc	write
writeprotect	writeprotcctcd
writablc	writable
whcncvcr	whenever
typc	type
sccurc	secure

Tabelle 1: Beispielhafter Ausschnitt aus einer OCR-MAPPING Ersetzungstabelle

23 Nach der grundlegenden Arbeit von Levenshtein (1966).

24 Die Jaro-Winkler Distanz in Jaro (1989) wurde ursprünglich für das Zusammenführen von verschiedenen Datenbankeinträgen über gleiche Personen (Record-Matching) in einer Volkszählung entwickelt.

VORTEILE Ein großer Vorteil dieses von uns vorgeschlagenen Algorithmus ist seine leichte Implementierbarkeit. Die Struktur des Algorithmus ist wie oben beschrieben sehr einfach. Zudem liegen für die Jaro-Winkler Distanz bereits schnelle Implementierungen vor[25]. Die innere Schleifen für den Vergleich der Wörter ließ sich zudem als C-Erweiterung für Python mit Hilfe des Cython Compilers implementieren, so dass sogar für den größten Corpus der IEEE mit ca. 1.600.000 verschiedenen, unkorrigierten Wörtern eine Ersetzungstabelle innerhalb von ca. einer Woche erstellt werden konnte[26]. Die Ersetzungstabellen erreichen dabei, je nach Fehlerquote, eine Größe von ca. 30% des unkorrigierten Gesamtwortschatzes. Die Ersetzungen stellen also durchaus eine bedeutende Verbesserung der Textqualität dar.

Ein weiterer großer Vorteil ist nicht nur die einfache Implementierbarkeit, sondern auch die einfache Realisierbarkeit. Es sind keinerlei Trainingsdaten, Wörterbücher, semantische Netze oder andere Informationen von außen notwendig um den Algorithmus effektiv ausführen zu können. Er ist deshalb auch vollständig sprachunabhängig[27].

NACHTEILE Obwohl die meisten Einträge der entstehenden Ersetzungstabelle die Fehlerwörter in das richtige oder zumindest ein ähnliches aber orthographisch korrektes Wort übersetzen, so muss dies nicht immer gegeben sein. Wenn in der Menge der ähnlichen Wörter eine Falschschreibung dominiert, so wird auch ein richtiges Wort durch ein falsches ersetzt. In Tabelle 1 wird z. B. das (fast) korrekte Wort „write-protect“ in das völlig zerstörte Wort „writeprotcctcd“ übersetzt. Dies ist nicht grundsätzlich schädlich, denn die statistischen Eigenschaften des Wortes bleiben trotzdem erhalten, nur heißt es nun anders. Nur wenn linguistische Methoden (wie z. B. Stemming) auf die ersetzten Worte angewendet werden, spielt die Orthographie eine Rolle. Da der überwiegende Teil der Wörter jedoch verbessert und nicht verschlechtert wird, schätzen wir die Wirkung in dieser Richtung als eher positiv ein. Durch die Art der Wahl des von nun an „richtigen“ Wortes tritt jedoch der Effekt auf, dass sich die einzelnen Dokumente nach der OCR-Korrektur mehr ähneln als vorher. Die globale Mehrheitsentscheidung bei der Auswahl des richtigen Wortes führt dazu, dass eventuell relevante Unterschiede zwischen den Dokumenten etwas verwischt werden. Dieser Effekt tritt bei den vorherigen Methoden, in denen einfach selten Worte verworfen werden, nicht auf.

ANDERE OCR-KORREKTURMETHODEN Neben unserem sehr einfachen Versuch gibt es natürlich noch komplexere Systeme[28]. Um nur eines davon herauszugreifen, betrachten wir nun kurz das System aus Tong u. Evans (1996) in dem die Autoren beschreiben wie OCR Fehler durch ein hybrides Modell automatisch korrigiert werden können. Sie betrachten dabei Wörter als eine Menge von N-Grammen. Über gefundene, wahrscheinliche Korrekturen mittels eines externen Wörterbuches wird eine zusätzliche Tabelle mit Fehlerwahrscheinlichkeiten

25 Wir haben auf PyLevenshtein von `http://code.google.com/p/pylevenshtein/` zurückgegriffen. PyLevenshtein ist eine schnelle, in C implementierte, Python Erweiterung. Zur weiteren Optimierung könnte zur Berechnung des Unterproblems der *edit-distance* noch der $O(n)$ Algorithmus von Wagner u. Fischer (1974) verwendet werden.

26 Dabei wurden ca. $128 \cdot 10^{10}$ Vergleichsoperationen bei einer Geschwindigkeit von ca. zwei Millionen Vergleichen pro Sekunde auf einem normalen Desktop-PC durchgeführt.

27 Mit Ausnahme der Distanzberechnung die statistische Annahmen über die Sprache einfließen lässt.

28 Kukich (1992) gibt einen hervorragenden Überblick über das Thema.

für bestimmte Buchstabenverwechselungen aufgebaut. In einem neuen Durchgang wird diese gelernte Tabelle dann in die Fehlerkorrektur mit einbezogen, so dass sich die Wahl der Korrekturen verbessert. Zusätzlich werden aus einem Trainingscorpus bedingte Wahrscheinlichkeiten gelernt. Die beschreiben die Wahrscheinlichkeit dass ein bestimmtes Wort verwendet wird, unter der Voraussetzung, dass das vorausgehende Wort bereits bekannt ist. Es fließen also auch Informationen über die Struktur der Sprache mit in die Korrekturentscheidung ein.

Dieses Modell scheint sehr vielversprechend zu sein und zeigt in den Experimenten der Autoren gute Ergebnisse. Jedoch hat es als Voraussetzung ein bestehendes Wörterbuch der Sprache und einen korrekten Trainingscorpus zum Erlernen der Sprachstruktur. Beides stand uns jedoch nicht direkt zur Verfügung, da davon ausgegangen werden muss, dass die Corpora über Informationssicherheit über ein extrem domänenspezifisches Vokabular verfügen.

3.2.1.5 *Entfernen von Synonymen*

Viele Begriffe in unserer Sprache kennen Synonyme, also andere Wörter, die die gleiche Bedeutung haben. Im Bereich der Informatik wäre ein bekanntes Synonym der Begriff „Laptop“ und „Notebook“, beide Begriffe meinen das gleiche, nämlich einen handlichen, mobilen Computer der an Rechenleistung einer Desktopmaschine in etwa ebenbürtig ist. Diese Synonymität muss nicht in jeder Domain gelten, es kann auch Bereiche geben, in denen diese Wörter nicht synonym sind. Dies gilt zum Beispiel für die beiden Wörter „student“ und „pupil“, welche beide einen Schüler bezeichnen könnten, aber im Kontext eines biologischen Textes, könnte das zweite Wort auch „Pupille“ bedeuten. Da sich eine Menge von zu verarbeitenden Dokumenten meistens auf einen begrenzten Fachbereich begrenzen, ist es möglich Synonymlisten zusammen zu stellen, die sehr effektiv Features zusammenfassen können und so helfen die Anzahl der Dimensionen weiter zu reduzieren.

3.2.1.6 *Erstellen der Feature-Matrix*

Sobald ein Text in Tokens unterteilt und die Anzahl der Features mit Hilfe von Stemming, dem Entfernen von Stop-Words und anderer Maßnahmen reduziert wurde, kann der Text in eine Zahlendarstellung überführt werden. Zuerst wird ein globales Dictionary mit allen Features aller Dokumente erstellt. Jedes Feature stellt damit eine Dimension in einem mehrdimensionalen Raum da. Die Idee ist es nun, Dokumente als Punkte in diesem Raum darzustellen, wobei dies mit Hilfe der Darstellung durch Vektoren geschieht. So ein Vektor beschreibt die Merkmale eines Textes und wird daher auch Feature-Vektor genannt. Besitzt nun ein Dokument ein bestimmtes Wort aus dem globalen Dictionary, so wird an der entsprechenden Stelle innerhalb des Feature-Vektors ein Zahlenwert gesetzt; besitzt es ein Wort nicht, so ist der Wert an dieser Stelle Null. Zur Berechnung des genauen Zahlenwertes ergeben sich mehrere Möglichkeiten, die im Folgenden noch vorgestellt werden sollen. Da alle Feature-Vektoren eines Corpus gleich lang sind, können sie auch als Spaltenvektoren einer sog. Feature-Matrix aufgefasst werden. In Tabelle 2 findet sich eine beispielhafte Feature-Matrix bestehend aus einem Corpus von drei Dokumenten d_1, d_2 und d_3. Das globale Dictionary umfasst vier Wörter, nämlich „virus“, „trojan“, „pgp“ und „encrypt“, die Vektoren der einzelnen Dokumente besitzen

Wörter \ Dokumente	d_1	d_2	d_3
virus	x	0	0
trojan	x	0	0
pgp	0	x	0
encrypt	0	x	x

Tabelle 2: Beispiel einer Feature-Matrix

also vier Dimensionen. Nun werden im Dokument d_1 zwei Begriffe genannt, nämlich „virus“ und „trojan“[29], also nimmt der Vektor an diesen Stellen den Wert x an, der einen beliebigen Zahlenwert ungleich Null darstellt. Das Dokument d_3 hingegen enthält lediglich das Wort „encrypt“ und besitzt daher auch nur an einer Stelle einen Wert ungleich Null. Momentan wurde nur von einem nicht näher definierten Zahlenwert x gesprochen der die Ausprägung eines Features in einem Dokument beschreibt, wie genau aber dieser Wert berechnet werden kann, soll nun näher betrachtet werden.

BINÄRE VEKTOREN Sofern eine gründliche Vorverarbeitung der Textdokumente stattgefunden hat und strenge Kriterien angelegt wurden um nur aussagekräftige Wörter in das globale Dictionary zu übernehmen, kann es Sinn machen, beim Generieren der Feature-Vektoren lediglich nach der Existenz oder Nicht-Existenz eines Features zu unterscheiden. Das heißt man erstellt für jedes Dokument einen Vektor, dessen Werte entweder 1 oder 0 annehmen, je nachdem ob das entsprechende Wort, welches durch das jeweilige Feature dargestellt wird in dem Dokument vorkommt oder nicht.

Der Vorteil dieser Darstellung ist vor allem, dass sich binäre Vektoren sehr speicherarm darstellen lassen, was bei der Verarbeitung von großen Datenmengen von Vorteil sein kann. Die Vorhersagekraft binärer Vektoren ist aber natürlich eingeschränkt, da ein Wort das weniger häufig in einem Dokument vorkommt nun die gleiche Wichtigkeit eingeräumt wird wie ein Wort das sehr häufig vorkommt.

ZÄHLEN DER WORTHÄUFIGKEITEN Eine Erweiterung zu binären Vektoren, die *Weiss2005* vorschlägt, ist das Verwenden von Vektoren mit drei Werten. Dabei wird ein Schwellenwert festgelegt, ab wann ein Wort häufig vorkommt. Taucht ein Wort also nicht im Dokument auf, so ist der Wert an der entsprechenden Stelle im Vektor 0, taucht es nur selten auf, d.h. das Wort kommt vor, aber seine Häufigkeit liegt nicht über dem Schwellenwert, dann ist sein Wert 1. Für Wörter deren Häufigkeit dagegen oberhalb des Schwellenwert liegen, ist der Wert 2. Dieser Weg kann als ein Kompromiss zwischen dem Zählen der reinen Worthäufigkeit und binären Vektoren angesehen werden.

Eine noch höhere Genauigkeit lässt sich durch die Angabe reiner Worthäufigkeiten erreichen. Wird also ein Wort 10 mal im Text genannt, dann beträgt auch sein Wert im Feature-Vektor 10. Auf diese Weise können aber längere Dokumente bevorzugt werden. Dies könnte dann passieren, wenn ein sehr langer Text von über zweihundert Wörtern einen Begriff sieben mal nennt und das gleiche auch in einem sehr

29 Man kann auch sagen, das lokale Dictionary des Dokuments d_1 umfasst die beiden Features „virus“ und „trojan“.

kurzen Text von nun zwanzig Wörtern passiert. Ist ist daher üblich die Worthäufigkeit nocheinmal durch die Anzahl aller Wörter im Text zu teilen um so den prozentualen Anteil eines Features zu erhalten.

TFIDF Alle bisher vorgestellten Verfahren haben das Problem, dass sie nur Aussagen darüber machen, ob ein Wort in einem Text vorkommt und evtl. wie häufig dies geschieht. Benötigt wird aber vielmehr eine Aussage darüber wie wichtig ein Wort für ein Dokument ist oder anders formuliert, wie gut ein Wort den Inhalt eines Textes repräsentiert. Ein sehr ausgereifter Weg um Feature-Vektoren zu erstellen ist daher den sogenannten TFIDF-Wert zu berechnen. TF steht als Abkürzung für *term frequency* also die Häufigkeit eines Wortes in einem Text und IDF für *inverse document frequency*, die inverse Häufigkeit eines Wortes in allen Dokumenten.

Die grundlegende Idee dieser Berechnung ist, dass Wörter die besonders häufig in einem Text genannt werden, allerdings kaum in anderen Dokumenten auftreten, für den Text sehr wichtig sein müssen und ihn deshalb gut beschreiben. Solche Wörter sollen also einen hohen Wert zugewiesen bekommen. Tritt hingegen ein Wort sehr häufig in allen Dokumenten auf, so ist es kein besonders wichtiges Merkmal und soll weniger stark bewertet werden.

Formal ausgedrückt nach Yates u. Neto (1999) ist der TFIDF das Produkt zweier Faktoren. Der erste Faktor, also die Worthäufigkeit wird wie in Formel 3.2 dargestellt berechnet, wobei $n_{d,t}$ die Anzahl von Wort t im Dokument d bezeichnet und berechnet sich im Grunde genommen aus der normalisierten Worthäufigkeit wie sie im vorherigen Abschnitt vorgestellt wurde.

$$\mathrm{tf}_{t,d} = \frac{n_{t,d}}{\sum_t n_{t,d}} \tag{3.2}$$

$$\mathrm{idf}_t = \log\left(\frac{|D|}{|\{d : w_t \in d\}|}\right) \tag{3.3}$$

Der zweite Faktor ist die inverse Dokumentfrequenz und ist in Formel 3.3 beschrieben. D bezeichnet die Menge aller Dokumente, diese wird geteilt durch die Menge aller Dokumente in denen der Term w_t vorkommt. Es ist darauf zu achten, dass jedes Wort mindestens in einem Dokument vorkommt, da es sonst zu einer Division durch Null kommen wird. Je häufiger also nun ein Wort in mehreren Dokumenten eines Corpus auftritt, desto kleiner wird sein IDF.

Der TFIDF-Wert ist jetzt, wie beschrieben, dass Produkt aus diesen beiden Faktoren. Die Formel 3.4 zeigt, wie der TFIDF-Wert für einen Term t im Dokument d berechnet wird.

$$\mathrm{tfidf}_{t,d} = \mathrm{tf}_{t,d} \cdot \mathrm{idf}_t = \frac{n_{t,d}}{\sum_t n_{t,d}} \cdot \log\left(\frac{|D|}{|\{d : w_t \in d\}|}\right) \tag{3.4}$$

Die Berechnung dieses Wertes ist ein sehr bekanntes und weit verbreitetes Verfahren um die Features eines Dokuments zu bewerten. Die Berechnung ist einfach durchzuführen und bewertet die Textmerkmale auf sinnvolle Weise. Die Darstellung ist allerdings für Menschen unge-

wohnt und unintuitiv und daher schwerer lesbar als z.B. die rein binäre Darstellung oder prozentuale Worthäufigkeit.

3.2.1.7 *Latente Semantische Analyse*

Die Latente Semantische Analyse (LSA) ist ein Verfahren zum Finden von allgemeinen Konzepten in Textdaten. Als ein Konzept kann ein Begriff verstanden werden, welcher mehrere thematisch ähnliche Begriffe zusammenfasst. So kann das Konzept *malware* Begriffe wie *virus*, *botnet*, *trojan* und *infection* umfassen. Durch das Finden solcher Konzepte ist es möglich die Anzahl der Dimensionen in eine Feature-Matrix stark zu reduzieren, da sich nach der LSA die Dimensionen für die vier Wörter des Konzeptes *malware* zu einem zusammenfassen lassen.

Aus mathematischer Sicht wird versucht die Punkte eines hochdimensionalen Raum möglichst gut in einem Raum mit weniger Dimensionen darzustellen. Dabei sollen aber wichtige Informationen erhalten bleiben und die unwichtigen Informationen verworfen werden. Man kann diesen Vorgang mit der Fotografie von Luftballons vergleichen, die in unterschiedlicher Entfernung durch einen dreidimensionalen Raum schweben. Möchte man also auf einem zweidimensionalen Foto möglichst viele Luftballons abbilden, so muss darauf geachtet werden, dass ein Ballon keinen anderen verdeckt. Die Fotografie stellt in gewissem Sinne eine Abbildung des dreidimensionalen Raums auf den zweidimensionalen Raum da. Die LSA ist in der Lage viel höher dimensionierte Räume und ihre enthaltenen Datenpunkte auf niedrigere Dimensionen abzubilden, ohne dass dabei wichtige Informationen verloren gehen[30].

Um eine solche Transformation durchführen zu können, wird demnach eine Darstellung von den zu untersuchenden Textdokumenten in einem Raum benötigt. Diese Darstellung wird z.B. durch die Feature-Matrix gegeben, auf dessen Erstellung in Absatz 3.2.1.6 genauer eingegangen wird. Die LSA benutzt diese Matrizen, wobei sie entweder TFIDF-Werte oder einfache Worthäufigkeiten besitzen können. Die Idee ist es nun mit Hilfe einer Singulärwertzerlegung (SVD)[31] die Singulärwerte der Matrix zu erhalten um damit Aussagen über die Korrelation von Wörtern und Dokumenten zu machen. Eine solche Zerlegung kann für jede beliebige Matrix erzeugt werden. Das Ergebnis der Singulärwertzerlegung sind drei spezielle Matrizen, die in Formel 3.5 für eine beliebige Feature-Matrix M dargestellt sind.

$$M = U\Sigma V^T \tag{3.5}$$

Die Matrix Σ ist dabei eine Diagonalmatrix und beinhaltet die in absteigender Reihenfolge sortierten Singulärwerte der Matrix M. Diese Werte stellen die Wurzel aus den Eigenwerten von M^TM da. Die Matrizen U und V^T sind jeweils orthogonale Matrizen, welche die Eigenvektoren von M^TM bzw. MM^T beinhalten. V^T ist dabei besonders interessant, da hier die Hauptkomponenten der ursprünglichen Matrix enthalten sind und diese verschiedenen Konzepte innerhalb des Corpus repräsentieren. Konzepte stellen also besonders aussagekräftige Linearkombinationen von gewichteten Wörtern über Dokumenten da. Nun

30 Eine ähnliche Metapher mit Fischen in einem Aquarium findet sich in weit ausführlicherer Form und tiefer gehenden Beispielen in Yu u. a. (2002).

31 Engl. für Singular Value Decomposition.

lassen sich aus Σ die niedrigsten k Singulärwerte streichen indem man diese auf Null setzt und dann mit dem so modifizierten Σ' anhand der oben genannten Formel M' berechnet. Nun gilt das $M \approx M'$ ist. Da allerdings nur die dominantesten Eigenvektoren von $M^T M$ bzw. MM^T in M' zusammengesetzt werden, sind in der neue Matrix die Korrelationen zwischen Wörtern die zusammen sehr häufig auftreten auch wesentlich höher. Auf diese Weise können unwichtige Informationen verworfen werden um wichtigere Zusammenhänge deutlicher zu machen. Da die neue Matrix M' die gleichen Dimensionen wie M hat, lässt sich mit ihr auch problemlos weiterrechnen.

Allerdings wurde auf diese Weise noch keine Dimensionen reduziert, sondern lediglich unwichtige Information verworfen. Möchte man tatsächlich eine Matrix erhalten, die zwar genauso viele Spalten (Dokumente) wie die ursprüngliche Matrix hat, aber weniger Reihen (Wörter), muss nach der Singulärwertzerlegung keine neue Matrix erstellt werden. Eine solche Dimensionsreduktion kann durchgeführt werden, indem man die untersten k Reihen der Matrix V^T wegschneiden. Mit k werden so die Anzahl der entfernten Dimensionen bezeichnet. Auf diese Weise erhält man eine neue Matrix V_k^T, welche nur diejenigen Hauptkomponenten, also Konzepte, von M beinhaltet, die besonders ausdrucksstark sind. Mit anderen Worten, V_k^T enthält in jeder Reihe den Featurevektor eines Konzepts und zwar genau die Konzepte die sich besonders eindeutig aus der gegebenen Feature-Matrix ableiten lassen.

BEISPIEL Zur besseren Veranschaulichung nochmal ein Beispiel dazu. Gegeben seien vier Dokumente $d_1, d_2, \ldots d_4$ in denen jeweils die fünf Begriffe *virus*, *trojan*, *pgp*, *encrypt* und *crack* vorkommen können. Eine Feature-Matrix M die nach Worthäufigkeiten zusammengestellt ist, ist in Tabelle 3 dargestellt. Dort ist zu erkennen, dass die Dokumente d_1 und d_2 besonders häufig die Wörter *virus* und *trojan* beinhalten, während die anderen Dokumente eher über *pgp*, *encrypt* und *crack* reden.

Term	d_1	d_2	d_3	d_4
virus	2	2	0	0
trojan	1	2	0	0
pgp	0	0	1	2
encrypt	0	0	2	2
crack	0	0	1	1

Tabelle 3: Feature-Matrix M mit Anzahl der Wörter für jedes Dokument

Nach der Zerteilung der Matrix M mittels SVD ergeben sich drei Matrizen U, Σ und V^T, welche in Gleichung 3.6 dargestellt sind. Nun sollen in einem nächsten Schritt M' berechnet werden, indem die untersten $k = 2$ Reihen von Σ auf Null gesetzt werden und es soll eine Dimensionsreduktion von ebenfalls $k = 2$ auf die Matrix V^T angewendet werden.

$$
\begin{aligned}
M &= U\Sigma V^T \qquad (3.6)\\
&= \begin{pmatrix} 0 & -0,8 & 0 & -0,6 \\ 0 & -0,6 & 0 & -0,8 \\ -0,6 & 0 & 0,8 & 0 \\ -0,7 & 0 & -0,5 & 0 \\ -0,4 & 0 & -0,3 & 0 \end{pmatrix} \cdot \begin{pmatrix} 3,8 & 0 & 0 & 0 \\ 0 & 3,6 & 0 & 0 \\ 0 & 0 & 0,6 & 0 \\ 0 & 0 & 0 & 0,6 \end{pmatrix} \cdot \\
&\quad \begin{pmatrix} 0 & 0 & -0,6 & -0,8 \\ -0,6 & -0,8 & 0 & 0 \\ 0 & 0 & -0,8 & 0,6 \\ -0,8 & 0,6 & 0 & 0 \end{pmatrix}
\end{aligned}
$$

In Gleichung 3.7 ist die bereinigte Matrix M' zu sehen. Deutlich ist zu erkennen, dass sich die Werte im Vergleich zur Feature-Matrix stark verändert haben. So betrug früher die euklidische Distanz zwischen den Dokumenten d_1 und d_2 $1,0$ und jetzt nur noch $0,64$. Das gleiche gilt für die beiden anderen Dokumente, dessen Distanz von $3,31$ auf $3,25$ sank. Zur gleichen Zeit sind auch die Abstände zwischen den Dokumenten, welche die beiden verschiedenen Konzepte umfassen, gestiegen. Das heißt die LSA hilft in diesem Fall die Dokumente besser unterscheidbar zu machen.

$$
\begin{aligned}
M' &= U\Sigma' V^T \qquad (3.7)\\
&= \begin{pmatrix} 0 & -0,8 & 0 & -0,6 \\ 0 & -0,6 & 0 & -0,8 \\ -0,6 & 0 & 0,8 & 0 \\ -0,7 & 0 & -0,5 & 0 \\ -0,4 & 0 & -0,3 & 0 \end{pmatrix} \cdot \begin{pmatrix} 3,8 & 0 & 0 & 0 \\ 0 & 3,6 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \cdot \\
&\quad \begin{pmatrix} 0 & 0 & -0,6 & -0,8 \\ -0,6 & -0,8 & 0 & 0 \\ 0 & 0 & -0,8 & 0,6 \\ -0,8 & 0,6 & 0 & 0 \end{pmatrix} \\
&= \begin{pmatrix} 1,7 & 2,2 & 0 & 0 \\ 1,3 & 1,7 & 0 & 0 \\ 0 & 0 & 1,4 & 1,7 \\ 0 & 0 & 1,8 & 2,2 \\ 0 & 0 & 0,9 & 1,1 \end{pmatrix}
\end{aligned}
$$

Um eine bessere Vorstellung von der Wirkung der LSA zu bekommen, sei hier nocheinmal M' gegeben unter der Voraussetzung, dass das Dokument d_2 den Begriff *crack* einmal verwendet. Dann ergibt sich folgendes Ergebnis:

$$
M' = \begin{pmatrix} 1,6 & 2,3 & 0 & -0,03 \\ 1,3 & 1,8 & 0,03 & 0 \\ -0,1 & 0,1 & 1,4 & 1,7 \\ -0,1 & 0,1 & 1,8 & 2,2 \\ 0,4 & 0,7 & 0,9 & 1,1 \end{pmatrix}
$$

Es ist deutlich zu sehen, wie nun auch Wörter gewichtet werden, die das Dokument ursprünglich nicht besaß. Auch treten nun negative Werte auf, die signalisieren, dass das Fehlen eines bestimmten Wortes ein ausschlaggebendes Merkmal eines Dokuments ist.

Die Datenreduktion hat zur Folge, dass aus V^T die letzten beiden Reihen weggeschnitten werden und man so $V_2^T = \begin{pmatrix} 0 & 0 & -0,6 & -0,8 \\ -0,6 & -0,8 & 0 & 0 \end{pmatrix}$ erhält. Diese Matrix enthält zwei Reihen für jeweils zwei Konzepte und vier Spalten für jedes Dokument. Der Datenraum der Feature-Vektoren wurde also von fünf auf nunmehr zwei Dimensionen reduziert. Deutlich ist zu erkennen, wie dabei die ersten beiden Spalten, also Dokument d_1 und d_2, auf das gleiche Konzept verweisen, genauso wie die Dokumente d_3 und d_4. Auf diese Weise wurde der Featureraum stark verkleinert ohne wesentliche Informationen zu verlieren.

VOR- UND NACHTEILE Die Vorteile einer LSA ist die Möglichkeit den Feature-Raum um viele Dimensionen einschränken zu können und so statistisches Rauschen zu eliminieren, was die Qualität von maschinellen Lernverfahren einschränkt. Selbst wenn die LSA nicht benutzt wird um Dimensionsreduktion zu betreiben, ist es zumindest möglich die Gewichtung von häufig zusammen auftretenden Wortgruppen in den Dokumenten zu stärken. Der Nachteil der LSA ist der hohe Speicherverbrauch des Verfahrens und die enorme Rechenzeit. Die Zeitkomplexität für die Singulärwertzerlegung beträgt $O(n^2)$, wobei n die Anzahl der Dokumente addiert mit der Anzahl aller Wörter ist. Bei der Berechnung der Singulärwertzerlegung entstehen außerdem drei sehr große Matrizen, zwar lässt sich die Diagonalmatrix Σ speichereffizient speichern, aber dies trifft nicht für die anderen Komponenten der SVD zu. Ein weiterer Nachteil bei der Dimensionsreduktion durch die LSA ist, dass Wörter durch Konzepte ersetzt werden und es einen eigenen Nachverarbeitungsschritt bedarf diesen Konzepten konkrete Begriffe aus dem Corpus zuzuordnen.

3.2.1.8 *Hauptkomponentenanalyse*

Ein artverwandtes Verfahren zur LSA ist die Hauptkomponentenanalyse oder auf Englisch „Principal Component Analysis (PCA)“, die bereits 1901 von Pearson skizziert[32], jedoch erst mit der Verbreitung der Computer populär wurde. Bei der PCA werden ebenfalls Matrizen im euklidischen Raum transformiert. Hier wird jedoch nicht nach gleichen Konzepten gesucht, sondern untersucht, mit welchen Achsen im Raum sich ein Großteil der Varianz der Datenpunkte bereits erklären lässt. Achsen, die nur wenig der Varianz darstellen, können verworfen werden. In Abbildung 9 ist dieses Prinzip dargestellt. Die ursprüngliche Menge der 2D-Punkte lässt sich durch zwei Achsen vollständig beschreiben. Die längere der Achsen beschreibt dabei schon etwas mehr als 95% der Varianz, während auf die kürzere Achse nur die restlichen 4,93% der Varianz entfallen. Nach der Ermittlung der Hauptkomponenten können die Punkte dann in das neue, von den beiden Vektoren aufgespannte Koordinatensystem transformiert werden.

Zur Berechnung der PCA stehen mehrere Algorithmen zur Verfügung. So kann zum einen über die Optimierung der Kovarianzmatrix durch Eigenwertberechnung der Ausgangsdaten, zum anderen über die oben beschriebene Singulärwertzerlegung oder über komplexere aber effizientere Algorithmen wie NIPALS[33] eine Hauptkomponentenanalyse durchgeführt werden[34]. Im Folgenden wird die einfachste Methode über die Eigenwertberechnung und Optimierung der Covarianzmatrix angerissen.

BERECHNUNG Sei M die ursprüngliche Feature-Matrix reduziert um die arithmetischen Mittel entlang der Dimensionen (Zentrierung auf den Mittelpunkt des Koordinatensystems) und M' die gesuchte transformierte Matrix (in der nach Wunsch Zeilen verworfen werden können). Es existiert eine transformierende Matrix P, so dass $M' = PM$. Sei $C_{M'} = \frac{1}{n-1} M'M'^{T}$ die Covarianzmatrix mit n als der Anzahl der

32 Vgl. Pearson (1901).
33 „Nonlinear Iterative Partial Least Squares“
34 Für eine kurze Einführung in die Methoden eine PCA zu berechnen siehe Smith (2002) oder Shlens (2005).

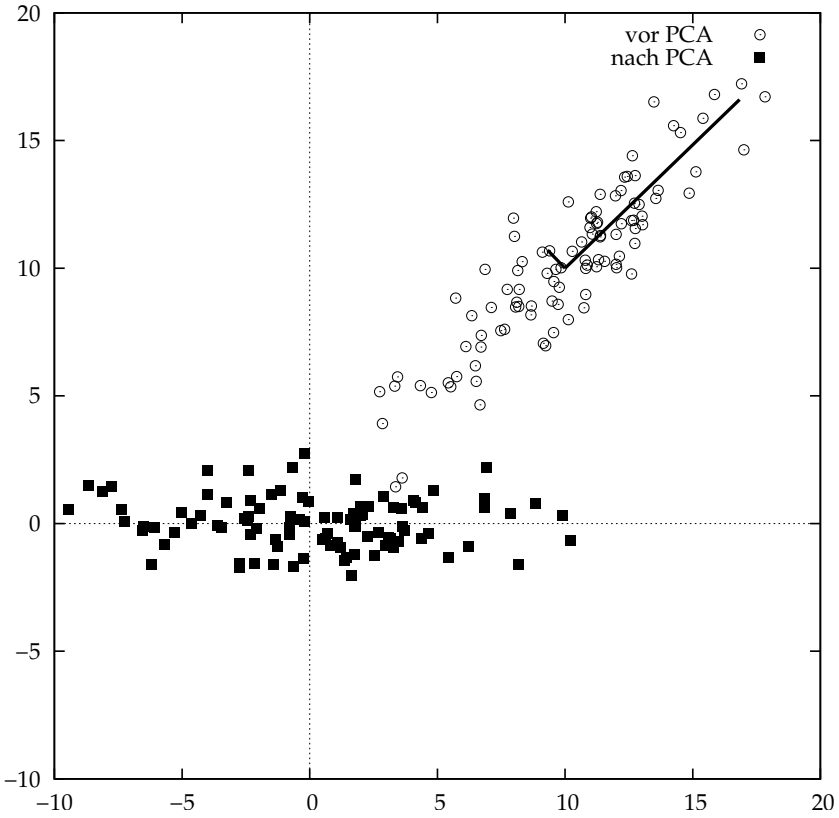


Abbildung 9: Funktionsweise der PCA im 2D-Raum anhand zufälliger Beispieldaten

Dimensionen jedes Vektors in M. Die Covarianzmatrix beschreibt auf der Diagonalen die Varianz der Spalten von M' und auf allen anderen Positionen das Rauschen der Daten. Wird P so gewählt, dass $C_{M'}$ zu einer Diagonalenmatrix wird (alle Werte außer der Diagonalen sind Null), so bedeutet dies, dass die Abbildung durch P zu einer Maximierung der Varianz auf den Achsen führt. Es lässt sich zeigen, dass dies genau dann der Fall ist, wenn die Zeilenvektoren p_i von P die Eigenvektoren der Matrix MM^T sind[35]. Die Berechnug der Eigenvektoren und Eigenwerte ist nun nur noch ein Standardverfahren aus der Linearen Algebra.

BEISPIEL Das oben aufgeführte Beispiel in Tabelle 3 würde durch eine PCA zu dem in Tabelle 4 dargestellten Ergebnis werden. Die Achsen setzen sich in der neu erhaltenen Tabelle aus jeweils mehreren alten Achsen zusammen und tragen jeweils eine bestimmte Fähigkeit die Varianz in den Daten zu erklären. Zu erkennen ist, dass bereits eine Achse ausreicht um 94,3% der Gesamtvarianz zu beschreiben. Diese Achse unterscheidet offensichtlich zwischen den Terms *virus* und *trojan*, sowie den Terms *pgp*, *encrypt* und *crack*. Die beiden weiteren Achsen tragen im Prinzip nicht mehr nennenswert zur Erklärung der Daten bei. Es wäre also ohne weiteres möglich, die Daten auf eine Dimension zu reduzieren. Die Punkte d_1 und d_2 liegen den Punkten d_3 und d_4 auf dieser neuen Achsen nahezu gespiegelt gegenüber. Die Verschiedenartigkeit der Punkte ist deutlich erkennbar.

35 Der Beweis findet sich unter anderem bei Shlens (2005).

Achsenbeschreibung	erklärte Varianz	d_1	d_2	d_3	d_4
$\begin{pmatrix} -0{,}53 & \text{virus} \\ -0{,}42 & \text{trojan} \\ 0{,}42 & \text{pgp} \\ 0{,}54 & \text{encrypt} \\ 0{,}27 & \text{crack} \end{pmatrix}$	94,3%	-1,63	-2,05	1,62	2,05
$\begin{pmatrix} \sim 0 & \text{virus} \\ 0{,}71 & \text{trojan} \\ 0{,}71 & \text{pgp} \\ \sim 0 & \text{encrypt} \\ \sim 0 & \text{crack} \end{pmatrix}$	3,4%	-0,35	0,35	-0,35	0,35
$\begin{pmatrix} -0{,}39 & \text{virus} \\ 0{,}57 & \text{trojan} \\ -0{,}57 & \text{pgp} \\ 0{,}39 & \text{encrypt} \\ 0{,}2 & \text{crack} \end{pmatrix}$	2,3%	-0,32	0,25	0,32	-0,25

Tabelle 4: Beispiel aus Tabelle 3 nach einer PCA (gerundete Werte)

VOR- UND NACHTEILE Der Vorteil der PCA ist die enorme Reduzierung der Dimensionalität, denn um die vollständige Varianz von n Vektoren zu erklären sind maximal $n - 1$ Dimensionen notwendig. In einem Corpus mit wenigen Dokumenten, aber bereits sehr hoher Dimensionalität, reduziert allein die PCA die Dimensionalität auf die Anzahl der Dokumente. Die einzelnen dabei erhaltenen Dimensionen haben dabei jeweils einen gewissen Anteil an der Erklärung der kompletten Varianz. Für die TFIDF-gewichtete Feature-Matrix der ACM TISSEC haben wir in Abbildung 10 beispielhaft die sich ergebende Verteilung der Varianz über die Dimensionen aufgetragen und die jeweils erreichte Summe der erklärten Varianz. Man sieht, dass ca. 80 Dimensionen eine höhere Erklärungskraft besitzen als die verbleibenden ca. 90 Dimensionen. Allein mit diesen 80 Dimensionen ließen sich dann auch knapp 70% der gesamten Varianz erklären.

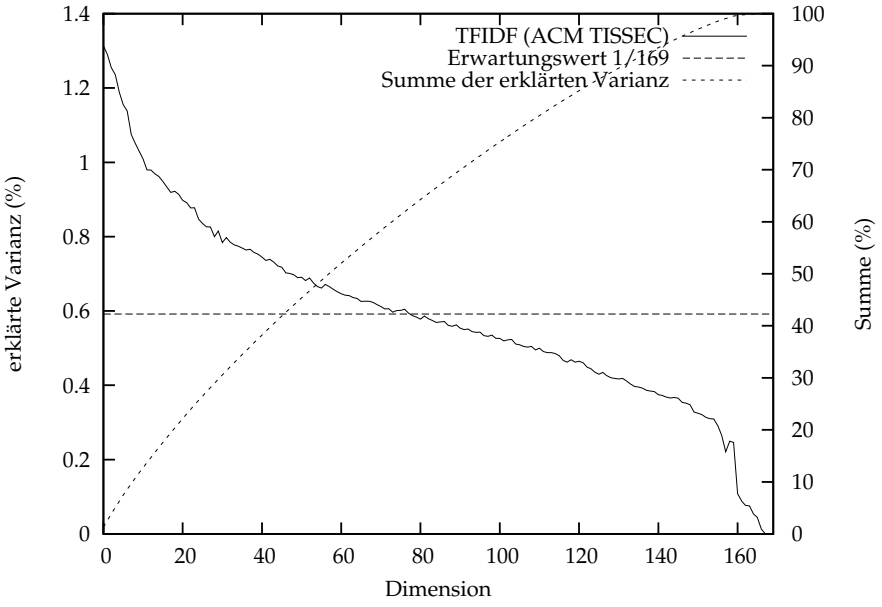


Abbildung 10: Verteilung der Varianz über die Dimensionen nach PCA für den ACM TISSEC Corpus

In dieser extremen Dimensionsreduktion liegt jedoch auch eine schwer absehbare Gefahr: Daten aus einem ca. 12.000-dimensionalen Raum in eine 170-dimensionalen Raum herunter zu transformieren bringt schwer vorhersehbare Effekte auf die Distanzen zwischen den Punkten. Ebenso nimmt die PCA an, dass sich die Daten als Linearkombinationen der Achsen darstellen lassen. Diese Linearitätsannahme muss nicht unbedingt zutreffen. Abseits der theoretischen Probleme ist auch die praktische Umsetzung nicht ohne weiteres durchführbar, obwohl fertige Implementierungen für die von uns verwendete Programmiersprache Python existieren. Je nach gewähltem Algorithmus werden in der Berechnung teilweise dicht besetzte Matrizen von hoher Dimensionalität (Anzahl der Punkte oder Anzahl der Dimensionen) benötigt. Die Ergebnismatrizen sind zwar in ihrer Dimensionalität reduziert, jedoch durch die neue Basis des Koordinatensystems nun nicht mehr dünn, sondern sehr dicht besetzt. Der Speicherverbrauch wächst also auch in dieser Richtung in jedem Fall enorm. Ebenso ist der Berechnungsaufwand für das in jedem Falle zu lösende Eigenwertproblem sehr hoch. Für die großen Datenquellen unserer Arbeit wäre eine Umsetzung der PCA nur parallelisiert mit enormen Implementierungs- und Rechenaufwand möglich.

3.2.2 *Klassifizieren*

Nachdem die Textdaten vorverarbeitet wurden, besteht die Möglichkeit die Daten mit maschinellen Lernverfahren thematisch zu gruppieren. Das Klassifizieren ist dabei ein überwachtes Lernverfahren, bei dem Daten bereits vordefinierten Kategorien zugeordnet werden. Das heißt, das Klassifizierungsalgorithmen zuerst trainiert werden müssen um ein Modell der Eingabedaten zu erstellen, anhand dessen dann weitere Daten klassifiziert werden. Es wird also gelernt, welche Eigenschaften der gegebenen Daten zu welcher Klasse gehört. Für gewöhnlich stellt man dazu ein sog. Trainingsset zusammen. Im Rahmen von Text-Mining könnte dies also eine Menge an Textdokumenten, wie z.B. Zeitungsartikel sein, bei denen bereits bekannt ist, welcher Klasse sie angehören. Für Zeitungsartikel könnten diese Klassen möglicherweise „Politik“, „Sport“ und „Kultur“ sein. Anhand des Trainingsset lernt also das Verfahren die Textdokumente den Klassen zuzuordnen, am Anfang wird dies noch zufällig sein, da aber die richtige Klasse für jedes Dokument bekannt ist, können Klassifizierungsalgorithmen erkennen, ob sie richtig lagen oder nicht und ihre internen Parameter entsprechend ändern um das nächste Mal ein Dokument mit gleichen Eigenschaften der korrekten Klasse zuzuordnen. Nachdem ein Klassifizierer ausreichend trainiert wurde, ist es möglich seine Qualität anhand von Testdaten zu überprüfen, um eine Aussage über die Genauigkeit des Verfahrens zu erhalten. Danach werden dann die Dokumente mit unbekannter Klasse durch den Algorithmus den korrekten Kategorien zugeordnet. Für Text-Mining-Anwendungen gibt es zwei sehr weitflächig eingesetzte Klassifizierungsverfahren, die hier vorgestellt werden sollen. Dies sind Naive-Bayes-Klassifizierer in Abschnitt 3.2.2.1 und Support-Vektormaschinen in Abschnitt 3.2.2.2. Eine kurze, vergleichende Zusammenfassung beider Verfahren findet sich bei 3.2.2.3.

3.2.2.1 *Naive Bayes*

Das nach dem englischen Mathematiker Thomas Bayes benannte Naive-Bayes-Verfahren ist ein probabilistischer Ansatz um Daten zu klassifizieren. Die Methode nutzt dabei das Bayestheorem um die Wahrscheinlichkeit zu berechnen mit der gegebene Eingabedaten zu einer der vordefinierten Klassen gehört. Das Verfahren ist besonders im Bereich des Text-Minings auf Grund seiner guten Performance und geringen Anzahl von Trainingsschritten sehr beliebt und findet tagtäglich Anwendung in E-Mail-Programmen um Spamnachrichten von gewünschten Nachrichten zu unterscheiden und auszusortieren.

Das Naive-Bayes-Verfahren kann als eine Funktion angesehen werden, welche eine Menge von gegebenen Attributen auf Klassen abbildet. Für Textdaten werden gewöhnlich binäre Attribute verwendet, d.h. die Eingabedaten bestehen aus einem Vektor welcher Einträge für jedes Wort des gesamten Corpus hat, wobei der Wert 0 eines Feldes bedeutet, dass das Wort nicht in dem Dokument vorhanden ist, welches durch den Vektor dargestellt wird und 1 weist auf das Vorhandensein des Wortes hin. Der Algorithmus geht dabei davon aus, dass alle Attribute voneinander statistisch unabhängig sind, sich also nicht gegenseitig bedingen[36]. In Text-Corpora gibt es allerdings eine Menge von unterschiedlichen Abhängigkeiten zwischen den Wörtern, dennoch erzielt das Verfahren erstaunlich gute Ergebnisse beim Klassifizieren von Texten.

ALGORITHMUS Der Naive-Bayes-Algorithmus benutzt den Satz von Bayes um Wahrscheinlichkeitsaussagen über die Zugehörigkeit von Inputdaten zu Klassen zu machen. Dieser Satz sagt aus, wie die Wahrscheinlichkeit für das Eintreten eines Ereignisses A, unter der Bedingung das B eingetreten ist, berechnet werden kann:

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)} \tag{3.8}$$

Für die Berechnung werden also drei Wahrscheinlichkeiten benötigt, die jeweiligen Wahrscheinlichkeiten für die Ereignisse A und B, sowie die Wahrscheinlichkeit für das Eintreten von Ereignis B unter der Bedingung das A vorher eingetreten ist.

Um nun Aussagen über die Kategorisierung von Textdaten zu machen, kann man unter K_c eine Variable für die Klassenzugehörigkeit zur Klasse c verstehen und D_i als ein Attribut auffassen, welches das Vorhandensein des i-ten Wortes aus dem globalen Dictionary in einem bestimmten Dokument beschreibt. Die Wahrscheinlichkeit, dass ein Dokument welches das i-te Wort beinhaltet in einem Dokument der Klasse c auftritt, ist demnach $P(D_i|K_c)$. Möchte man nun diese Aussage für ein bestimmtes Dokument konkretisieren, müssen alle Wörter in diesem Dokument betrachtet werden und die Wahrscheinlichkeit, dass ein Dokument D in einer Klasse c enthalten ist, ergibt:

$$P(D|K_c) = \prod_i P(D_i|K_c) \tag{3.9}$$

36 Damit stellt das Naive-Bayes-Verfahren eigentlich nur eine spezielle Form von Bayesschen Netzwerken dar. Rine detaillierte Beschreibung findet sich in Russell u. Norvig (2004, Kapitel 20.2).

Dokument	*virus*	*worm*	*wireless*	*network*	Klasse
1	1	1	0	0	m
2	1	0	0	0	m
3	1	0	1	0	w
4	0	1	0	1	m
5	0	0	0	1	w

Tabelle 5: Dokumente des Trainingssets mit ihren enthaltenen Wörtern und Klassenzugehörigkeiten

Dies gilt natürlich nur unter der Voraussetzung, dass die Wörter untereinander bedingt unabhängig sind, was der „naiven" Voraussetzung für Naive-Bayes-Klassifikation entspricht. Was für die Klassifizierung von Texten benötigt wird, ist eine Aussage über die Wahrscheinlichkeit, dass ein gegebenes Dokument zu einer bestimmte Klasse c gehört. Mathematisch lässt sich dies also als die Wahrscheinlichkeit $P(K_c|D)$ ausdrücken. Nun lässt sich der Satz von Bayes anwenden um diesen Wert zu berechnen, wie in Formel 3.10 dargestellt.

$$\begin{aligned} P(K_c|D) &= \frac{P(D|K_c) \cdot P(K_c)}{P(D)} \\ &= \frac{\prod_i P(D_i|K_c) \cdot P(K_c)}{P(D)} \end{aligned} \tag{3.10}$$

Mit dieser Formel lassen sich jetzt die Wahrscheinlichkeiten für die Trainingsdaten berechnen. Initial gilt für $P(K_c)$ die A-Priori-Wahrscheinlichkeit, diese kann aber später über den Anteil der in dieser Kategorie befindlichen Dokumente angegeben werden. Soll ein Dokument kategorisiert werden, so muss für jede erwartete Klasse $P(K_c|D)$ errechnet werden. Die Ergebnisse werden verglichen und das Dokument derjenigen Klasse mit der höchsten Wahrscheinlichkeit zugeordnet.

BEISPIEL Angenommen es gibt einen Text-Corpus mit verschiedenen Dokumenten, die sich lediglich aus den vier Begriffen *virus*, *worm*, *wireless* und *network* zusammensetzen, so machen die Variablen D_i für $i \in \{1, 2, 3, 4\}$ also eine Aussage über das Vorhandensein dieses Wortes in einem Dokument. Es soll nun ein Dokument auf die Zugehörigkeit seiner Klasse überprüft werden. Es stehen zwei Klassen zur Auswahl, zum einen Klasse m zum Themengebiet „Malware" und Klasse w zum Themengebiet „Wireless". Nach einem ersten Trainingsdurchlauf wurden bereits fünf Dokumente kategorisiert, die Dokumente mit ihren enthaltenen Wörtern und ihrer Klassenzugehörigkeit sind in der Tabelle 5 aufgelistet.

Sei $D = \{D_1 = 1, D_2 = 0, D_3 = 0, D_4 = 1\}$ ein Dokument welches lediglich den Begriff *network* und *virus* beinhaltet und nun klassifiziert werden soll. Die Wahrscheinlichkeit, dass dieses Dokument einer bestimmten Klasse angehört, lässt sich durch die Gleichung 3.10 bestimmen, gesucht sind also $P(K_m|D)$ und $P(K_w|D)$. Anhand den Trainingsdaten lassen sich nun die Wahrscheinlichkeiten für die Klassen bestimmen, diese sind $P(K_m) = \frac{3}{5}$ und $P(K_w) = \frac{2}{5}$. Unter der Voraussetzung dass alle Wörter gleichverteilt in den Dokumenten vor-

kommen, ist $P(D)$ konstant und muss zum Vergleichen der Ergebnisse daher nicht bestimmt werden. Nun ergibt sich nach Anwendung von Gleichung 3.10 auf das Dokument D folgendes:

$$\begin{aligned}
P(K_m|D) &= \frac{\prod_i P(D_i|K_m) \cdot P(K_m)}{P(D)} \\
&= \frac{P(D_1 = 1|K_m) \cdot P(D_2 = 0|K_m)}{P(D)} \cdot \\
&\quad \frac{P(D_3 = 0|K_m) \cdot P(D_4 = 1|K_m) \cdot P(K_m)}{P(D)} \\
&= \frac{\frac{2}{3} \cdot \frac{1}{3} \cdot 1\frac{1}{3} \cdot \frac{3}{5}}{P(D)} \\
&= \frac{0,044}{P(D)} \\
P(K_w|D) &= \frac{\prod_i P(D_i|K_w) \cdot P(K_w)}{P(D)} \\
&= \frac{P(D_1 = 1|K_w) \cdot P(D_2 = 0|K_w)}{P(D)} \cdot \\
&\quad \frac{P(D_3 = 0|K_w) \cdot P(D_4 = 1K_w) \cdot P(K_w)}{P(D)} \\
&= \frac{\frac{1}{2} \cdot 1 \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{2}{5}}{P(D)} \\
&= \frac{0,05}{P(D)}
\end{aligned}$$

Das Ergebnis zeigt, dass die Wahrscheinlichkeit für Dokument D zur Klasse w zum Thema „Wireless“ zu gehören mit $P(K_w|D) = \frac{0,05}{P(D)}$ geringfügig höher ist als mit $P(K_m|D) = \frac{0,044}{P(D)}$ zur Klasse m zum Thema „Malware“, man würde das Dokument also entsprechend einordnen und danach die Wahrscheinlichkeiten für die Ereignisse K_w und K_m neu anpassen.

VOR- UND NACHTEILE Klassifikation mit Hilfe des Naive-Bayes-Ansatzes ist für Text-Mining-Anwendungen sehr gut geeignet. Das Verfahren ist sehr schnell, effizient und liefert gute Ergebnisse. Besonders verrauschte Daten können von Naive-Bayes-Klassifizierer gut gehandhabt werden. Irrelevante Attribute haben kaum Einfluss auf die Klassifikation, was das Verfahren auch für verrauschte, hochdimensionale Daten wirksam macht. Desweiteren gibt es keine Parameter, die für das Verfahren bestimmt werden müssen. Auf der anderen Seite ist die Annahme, dass alle Wörter eines Attributs bedingt unabhängig sind, gefährlich, da in Texten durchaus Abhängigkeiten auftreten können. Diese wiederum können die Qualität der Klassifikation verschlechtern. Dennoch stellt das Naive-Bayes-Verfahren eine empfehlenswerte und zuverlässige Methode zur Textklassifikation dar.

3.2.2.2 *Support-Vektormaschinen*

Ein weiteres bekanntes Verfahren zum Klassifizieren von Daten sind Support-Vektormaschinen (SVMs). Dabei handelt es sich um einen relativ neuen Ansatz aus der Gruppe statistischer Lernverfahren. Support-Vektormaschinen erwarten als Eingabe numerische Vektoren aus einem

beliebig dimensionierten Feature-Raum und stellen eine Funktion da, welche diese Vektoren auf eine von zwei Klassen abbildet. SVMs sind sehr robust gegenüber verrauschte Daten und können auch mit hochdimensionierten Vektoren umgehen, was sie für Text-Mining interessant macht.

Mathematisch gesprochen, versuchen Support-Vektormaschinen eine lineare Separierung der gegebenen Vektoren im Feature-Raum zu finden. Für den zweidimensionalen Raum würde dies bedeuten, dass die Support-Vektormaschine während ihres Trainings eine lineare Funktion zu finden versucht, welche zwischen den Vektoren der beiden Klassen verläuft. Möchte man dann anschließend einen neuen Vektor klassifizieren, muss lediglich geprüft werden, ob dieser „rechts“ oder „links“ von der Linie liegt, welche die Klassen separiert. Ein Problem an dieser Stelle ist, dass es prinzipiell unendlich viele Möglichleiten gibt eine lineare Funktion zu definieren, welche den Raum trennt. Um eine optimale Trennung zu finden benutzen Support-Vektormaschinen sog. Support-Vektoren, also Punkte die besonders nahe an der Grenze liegen. Es wird dabei versucht die Trennung der beiden Klassen so zu legen, dass diese einen maximalen Abstand zu allen Support-Vektoren besitzt. Dadurch wird sichergestellt, dass die Grenze besonders „mittig“ zwischen beiden Klassen verläuft.

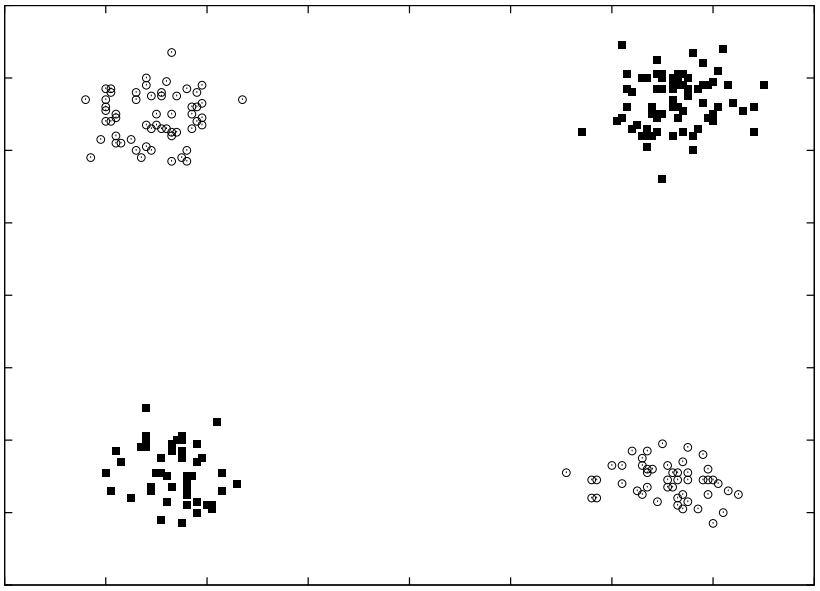


Abbildung 11: Beispiel für das XOR-Problem

Ein weiteres Problem ist, dass bereits für den zweidimensionalen Raum sehr leicht Fälle denkbar sind, bei denen keine lineare Separierung möglich ist. Ein klassisches Beispiel sind Datenpunkte, welche durch die XOR-Funktion in einen zweidimensionalen Raum abgebildet werden. Die XOR-Funktion liefert zwei Klassen, einmal mit den Punkten $(0,0)$ und $(1,1)$, sowie eine Klasse mit $(1,0)$ und $(0,1)$. In Abbildung 11 ist ein Beispiel mit mehreren Datenpunkten gezeigt, welche diesen beiden Klassen angehören. Wie man dort leicht sieht, ist es nicht möglich eine Linie durch den Graph zu ziehen, welche die Punkte der beiden Klassen trennt. Eine Trennung dieser Punkte mit Hilfe einer linearen Funktion ist also nicht möglich. Diese Aussage gilt

allerdings nur für den zweidimensionalen Raum. Die Idee ist nun die vorhanden Datenpunkte als Vektoren in einem höher-dimensionalen Raum darzustellen. In dem Beispiel mit der XOR-Funktion reicht schon ein dreidimensionaler Raum aus. Man könnte sich also vorstellen die Punkte der einen Klasse höher als die der anderen zu legen, man kann dann eine Ebene zwischen beiden Klassen legen. Dazu muss nun eine Abbildung gefunden werden, welche die zweidimensionalen Punkte aus dem Eingaberaum in den neuen dreidimensionalen Raum überführt. Dieser Vorgang wird auch „Kernel-Trick" genannt und geschieht mit sog. Kernel-Funktionen. Durch den neuen höherdimensionalen Feature-Raum kann nun eine Hyperebene gelegt werden, welche die Eingabedaten linear separiert, auf diese Weise können Support-Vektormaschinen auch Daten klassifizieren, welche auf komplexe Weise zusammenhängen.

ALGORITHMUS Anhand einer Menge von Trainingsdaten wird eine Hyperebene bestimmt, welche diese Daten linear separiert. Bei den Trainingsdaten handelt es sich dabei um numerische Vektoren, im Falle von Textdokumenten also zum Beispiel um deren Feature-Vektor mit TFIDF-Werten, sowie um deren Zugehörigkeit zu einer Klasse. Betrachtet werden dabei nur zwei Klassen. Die Trainingsdaten sind demnach für den d-dimensionalen Eingaberaum X und der Klassen $C = -1, 1$ eine Menge von Tupeln: $\{(x_i, c_i)|x_i \in \mathbb{R}^d, c_i \in C\}$ mit $i = \{1, 2, 3, \ldots, n\}$, wobei n die Anzahl der Testdatensätze beschreibt.

Gesucht ist nun eine Hyperebene, welche die Punkte in X so teilt, dass sie diejenigen Punkte mit $c_i = -1$ von denen mit $c_i = 1$ trennt. Zur Beschreibung dieser Ebene wird der Normalenvektor w und der Bias b benötigt, die so gewählt werden müssen, dass der folgende Ausdruck erfüllt wird:

$$w \cdot x + b = 0 \tag{3.11}$$

Nun ist es auch möglich c_i genauer zu beschreiben. Da die Punkte nämlich entweder ober- oder unterhalb der Ebene liegen müssen gilt:

$$c_i = \mathrm{sign}(w \cdot x + b) \tag{3.12}$$

Das Finden der Parameter der optimalen Hyperebene im Sinne eines maximalen Abstandes zu den Support-Vektoren lässt sich nun als Optimierungsproblem beschreiben, bei dem der Ausdruck $\frac{\|w\|^2}{2}$ minimiert werden muss. Dabei muss immer beachtet werden, dass folgende Bedingung nicht verletzt wird[37]:

$$c_i(w \cdot x + b) \geqslant 1 \tag{3.13}$$

Da der zu optimierende Ausdruck quadratisch ist und die Beschränkung linear, handelt es sich um ein konvexes Optimierungsproblem, das sich mit Standardverfahren wie der Lagrange-Multiplikatorenregel umformulieren und lösen lässt. Auf diese Weise lässt sich der Normalen-

37 Warum dies so ist, kann detailliert in Tan u. a. (2005, Kapitel 5.5) nachgelesen werden.

vektor w als Linearkombination der Trainingsdaten zusammensetzen, so dass gilt:

$$w = \sum_{i=1}^{m} \lambda_i c_i x_i \tag{3.14}$$

Der Vektor λ wird dabei durch das Optimierungsverfahren bestimmt, so dass nun eine konkrete Hyperebene durch den Raum gelegt werden kann. Möchte man nun einen neuen Datenpunkt klassifizieren, so kann dies mit der Klassifikationsfunktion f geschehen, die in 3.15 definiert ist.

$$f(z) = \mathrm{sign}(w \cdot z + b) = \mathrm{sign}(\sum_{i=1}^{n} \lambda_i c_i x_i + b) \tag{3.15}$$

Diese Klassifikationsfunktion ist allerdings nur für linear separierbare Daten anwendbar. Für komplexere Probleme, wie sie in den meisten realen Anwendungen vorkommen, werden daher, wie oben ausgeführt, Kernel-Funktionen benutzt um die Daten aus dem Eingaberaum X auf einen höher dimensionalen Featureraum F abzubilden. Eine Kernel-Funktion ist also eine Abbildung $K : X \times X \to \mathbb{R}$ mit $K(x, y) = \varphi(x) \cdot \varphi(y)$ für alle $x, y \in X$ und einer Feature-Mapping-Funktion $\varphi : X \to F$, sofern K das Mercer-Theorem nicht verletzt. Beispiele für solche Funktion sind $K(x, y) = (x \cdot y + 1)^p$ oder $K(x, y) = e^{-\|x-y\|^2/(2\rho^2)}$. Wie man sieht, müssen für diese Funktionen ggf. geeignete Parameter gefunden werden. Durch das Anwenden der Kernelfunktion lässt sich nun der Normalenvektor im neuen Featureraum, wie in Formel 3.16 gezeigt, darstellen, sowie die Klassifikationsfunktion umformulieren, siehe dazu Formel 3.17.

$$w = \sum_{i=1}^{m} \lambda_i c_i \varphi(x_i) \tag{3.16}$$

$$f(z) = \mathrm{sign}(w \cdot z + b) = \mathrm{sign}(\sum_{i=1}^{n} \lambda_i c_i K(x_i, z) + b) \tag{3.17}$$

Ein Nachteil klassischer Support-Vektormaschinen ist der Umstand, dass diese nur zwei Klassen handhaben können. In vielen Fällen möchte man aber Datenpunkte in mehr als nur zwei Klassen einteilen können. Eine Möglichkeit dieses Problem anzugehen ist die Verwendung von binären Klassen, d.h. für jede Klasse wird mittels der SVM ein Modell erstellt, welches entscheidet, ob ein Punkt zu dieser Klasse gehört oder nicht. Dieser Ansatz wird auch *one-against-all* genannt und erlaubt mit wenig Rechenaufwand Daten auch mehreren Klassen zuordnen zu können. Allerdings büßt man dadurch auch einiges an Genauigkeit ein, so dass es günstiger sein kann Modelle für jede Kombination von Klassen mit Hilfe der sog. *one-against-one*-Methode zu erstellen. Dadurch werden also bei k verschiedenen Klassen $k(k-1)/2$ Modelle erstellt. Der Rechenaufwand ist daher höher, die Qualität der Klassifizierung kann allerdings so verbessert werden. Ein Vergleich zwischen diesen Methoden und auch anderen Ansätzen für das Verarbeiten mehrerer Klassen findet sich bei Hsu u. Lin (2002).

VOR- UND NACHTEILE Das Lernen von Daten kann bei Support-Vektormaschinen als konvexes Optimierungsproblem formuliert und auf diese Weise durch effiziente Algorithmen gelöst werden. Der Vorteil dieses Ansatzes im Gegensatz zu anderen maschinellen Lernverfahren ist, dass SVMs dadurch viel wahrscheinlicher ein globales Optimum für eine Trennung der Datensätze finden. Support-Vektormaschinen sind in der Lage relativ schnell und effizient Modelle aus Trainingsdaten zu erstellen, anhand derer sich neue Daten klassifizieren lassen. Dabei sind SVMs relativ robust gegenüber verrauschten Daten und können auch mit hochdimensionalen Daten effizient arbeiten, was sie für Text-Mining-Anwendungen interessant macht.

Der Nachteil von Support-Vektormaschinen ist die Parameterwahl. Es muss zuerst eine geeignete Kernel-Funktion gefunden werden, die je nach Beschaffenheit der Daten verschiedene Eigenschaften besitzen muss. Kernel-Funktionen besitzen selbst noch unterschiedliche Parameter, die einen nicht unerheblichen Einfluss auf die Performance des Verfahrens haben. Ein weiteres Problem ist, dass SVMs lediglich binäre Klassifizierer sind und Datenpunkte nur in eine von zwei Klassen einordnen können. Möchte man mehrere Klassen verwenden, muss das klassische Verfahren erweitert werden wobei zwischen verschiedenen Ansätzen mit unterschiedlicher Performance abgewogen werden muss.

3.2.2.3 *Zusammenfassung*

Es wurden zwei klassische Klassifikationsverfahren vorgestellt, die häufig im Bereich des Text-Minings angewendet werden. Beide Ansätze sind dabei grundlegend verschieden. Der Naive-Bayes-Klassifikator untersucht Dokumente nach statistischen Merkmalen, wobei die Worthäufigkeit eine ausschlaggebende Rolle spielt. Support-Vektormaschinen hingegen untersuchen numerische Vektoren in Hinblick auf ihre lineare Separierbarkeit, die ggf. durch das Transformieren in einen höheren Raum gewährleistet wird. Dokumente können bei diesem Ansatz auch durch andere Arten von Feature-Matrizen dargestellt werden. Die Implementierung von Support-Vektormaschinen ist im Vergleich zu der Implementierung eines Bayes-Klassifikators sehr aufwendig und benötigt das Verwenden eines extra Optimierungsverfahren zum Finden der Eigenschaften der Hyperebene, welche die Klassen trennt. Auf der anderen Seite ist letzteres Verfahren so verbreitet, dass gute Programmbibliotheken für die meisten Programmiersprachen bereits vorhanden sind. Ein weiterer Unterschied ist, dass Bayes-Klassifikatoren sehr leicht für das Klassifizieren von Daten in mehr als zwei Klassen verwenden lässt, während SVMs nur binär klassifizieren können, so dass hier noch einmal zusätzlicher Aufwand besteht, wenn mehr Klassen verwendet werden sollen. Beide Verfahren eignen sich für Text-Mining-Anwendungen, die Genauigkeit mit denen beide Verfahren klassifizieren ist ebenfalls vergleichbar[38].

3.2.3 *Clustern*

Ein weiteres maschinelles Lernverfahren um Textdaten automatisiert zu gruppieren, ist das sog. Clustern. Dabei handelt es sich im Gegensatz zur Klassifikation um ein unüberwachtes Lernverfahren, was bedeu-

38 Bei Huang u. a. (2003) findet sich ein Vergleich beider Ansätze in Hinblick auf AUC (*area under the curve*) und Genauigkeit.

tet, dass beim Einsatz von Cluster-Verfahren keine Modelle in einem extra Trainingsvorgang erstellt werden müssen, sondern dass alle Eingabedaten automatisch in Kategorien zugeordnet werden. Ein weiteres Merkmal dieser unüberwachten Lernverfahren ist, dass der Inhalt der Kategorien nicht vorgegeben werden muss. Je nach Verfahren ist es zwar möglich eine bestimmte Anzahl von Clustern zu bestimmen, die erwartet werden, aber die eigentlichen Inhalte dieser Cluster werden von dem Algorithmus automatisiert ermittelt. Bei einem Klassifizierer würde man also z.B. Zeitungsartikel den Kategorien „Sport", „Politik" und „Kultur" zuordnen, beim Clustern hingegen würde der Algorithmus selbstständig herausfinden, welche Kategorien und ggf. wie viele es gibt, so dass das Ergebnis eine Einteilung in „Sport", „Innenpolitik", „Außenpolitik" und „Kultur" sein könnte. Es werden drei verschiedene Clusterverfahren betrachtet. In Abschnitt 3.2.3.1 wird K-Means vorgestellt, ein Verfahren das häufig im Bereich Text-Mining angewendet wird. Danach wird auf in Absatz 3.2.3.2 auf ein Cluster-Verfahren namens DBSCAN eingegangen, welches Form und Anzahl der Cluster eigenständig feststellen kann und abschließend wird ein eher unbekanntes, allerdings speziell für Text-Mining-Anwendungen ausgerichtetes Verfahren in 3.2.3.3 dargestellt. Abschließend findet in Abschnitt 3.2.3.4 ein ausführlicher Vergleich und eine Zusammenfassung der vorgestellten Verfahren statt.

3.2.3.1 *K-Means*

K-Means ist ein klassischer Ansatz um Daten in verschiedene Partitionen einzuteilen und wird häufig auch für Text Mining verwendet. Der Algorithmus ist weit verbreitet, gut erforscht und kennt verschiedene Erweiterungen und Verbesserungen, die vor allem die Wahl der Parameter betrifft.

ALGORITHMUS Die grundlegende Idee ist es eine vorhandene Menge von Daten in k verschiedene Cluster einzuteilen. Dies geschieht indem zunächst alle Dokumente zufällig auf die vorhandenen Cluster verteilt werden. Nun wird für jeden Cluster der Mittelwert der von ihm enthaltenen Daten berechnet. Dieser Mittelwert stellt jetzt eine Art Repräsentant für diesen Cluster da. Da als Abstandsmaß in der klassischen Variante von K-Means auf die euklidische Distanz zurückgegriffen wird, bildet dieser Mittelwert das Zentrum einer Kugel im Feature-Raum in der alle Daten des Clusters enthalten sind. Es wird daher auch von einem *Centroid* gesprochen.

Als nächstes werden alle Daten in diejenigen Cluster verschoben, mit dessen Repräsentanten sie am ähnlichsten sind. Für die nun veränderten Cluster müssen jetzt die neuen Repräsentanten berechnet werden. Ab jetzt wiederholt sich der Vorgang der Zuordnung von Daten zu Clustern bis keine Daten mehr verschoben werden können oder nachdem eine zuvor festgelegte Anzahl maximaler Durchläufe erreicht wurde.

Das Ergebnis sind Cluster deren enthaltenen Daten minimalen Abstand zum Cluster-Mittelwert haben. Dabei ist darauf zu achten, dass der unmodifizierte K-Means in einem lokalen Minimum anhalten kann, da die initiale Zuordnung von Daten zu Clustern zufällig erfolgt. Eine Strategie um dies zu verhindern ist es, den Algorithmus mehrere Male mit verschiedenen Startkonfigurationen auszuführen bis bessere Ergebnisse erzielt werden. Eine andere Lösung ist, einen globalen Mittelwert zwischen allen Daten zu berechnen und die Daten dann in

sortierter Reihenfolge gemäß ihres Abstands zum globalen Mittelwert der Reihe nach auf die Cluster zu verteilen, so dass es initiale Cluster gibt, die Daten mit weiter Entfernung und geringerer Entfernung zum Mittelwert enthalten. Nun werden nach dem oben beschriebenen Verfahren die Daten in den Clustern umverteilt wobei als Einschränkung hinzukommt, dass Daten nur in anliegende Cluster verschoben werden dürfen.

Betrachtet man diesen Ansatz genauer so kann man K-Means auch als Optimierungsproblem auffassen, wobei die Summe der Abstände aller Punkte im Datenraum zu ihrem nächst gelegenen Centroid minimiert werden sollen. Die einzelnen Schritte des Algorithmus sind hier nocheinmal stichpunktartig aufgeführt:

1. Verteile alle Vektoren gleichmäßig und zufällig auf k Cluster
2. Berechne für alle Cluster den Mittelpunkt, also den Durchschnitt aller Vektoren
3. Verschiebe alle Vektoren in denjenigen Cluster, dessen Durchschnitt dem Vektoren am ähnlichsten ist
4. Wurde kein Vektor verschoben, beende das Programm, ansonsten mache bei Schritt 2 weiter.

In Abbildung 12 ist ein mögliches Clustering mittels K-Means angegeben. In dem abgebildeten Datenraum gibt es insgesamt drei Punktwolken, die man intuitiv auch drei Clustern zuordnen würden. Es wurde nun aber mit $k = 4$ geclustert. Die Punktwolke links oben im Bild wurde damit auf zwei Clustern verteilt. Das Beispiel zeigt wie ein unglücklich gewähltes k unerwartete Ergebnisse hervorbringen kann. Bei dem Einsatz von K-Means muss also das k weise gewählt werden.

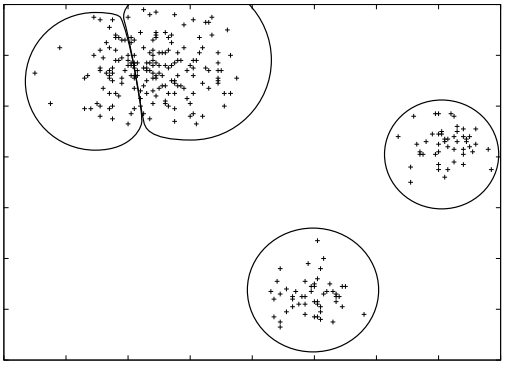


Abbildung 12: Beispiel für Clustering mittels K-Means

VOR- UND NACHTEILE Die Stärke des K-Means-Algorithmus liegt vor allem in seiner Effizienz, sofern k, also die Anzahl der Cluster, weit geringer als die Anzahl der zu partitionierenden Daten ist, was in den meisten Anwendungen der Fall sein dürfte, besitzt K-Means einen Zeitaufwand von $O(n)$. Ein weiterer Vorteil ist, dass jeder Cluster einen leicht zu berechnenden Repräsentanten besitzt, der den Cluster zutreffend beschreibt.

Der größte Nachteil dieses Ansatzes ist vor allem die Wahl des Parameters k. Der Algorithmus ist nicht in der Lage die Anzahl der Cluster

selbstständig festzulegen, diese muss also vom Anwender explizit gewählt werden und setzt Wissen des Anwenders über den ungefähren Aufbau des zu untersuchenden Feature-Raums vorraus. Häufig gibt es aber nur eine ungefähre Schätzung wie viele Partitionierungen zu erwarten sind, so dass verschiedene, statistische Methoden angewendet werden müssen um diese Abschätzung zu konkretisieren. Ein weiteres Problem ist, dass nur kugelförmige Cluster erkannt werden können und K-Means ungenau für Daten ist, die verschieden große Cluster mit unterschiedlicher Punktdichte aufweisen[39].

3.2.3.2 *DBSCAN*

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) ist ein weiterer, häufig genutzter Cluster-Algorithmus, welcher selbstständig die Anzahl von Clustern und deren Größe bestimmen kann. Das Verfahren versucht dabei Regionen im Datenraum mit hoher Dichte auszumachen und ordnet diese einem Cluster zu. DBSCAN ist nach Ester u. a. (1996) insbesondere für das Verarbeiten von vielen Daten geeignet, da beim Clustern solcher Datensätze häufig nur wenig Kenntnisse über die räumliche Verteilung der Daten vorliegen, so dass man nur schwer Aussagen über die Form und Größe der Cluster oder ihrer Anzahl machen kann. Desweiteren läuft DBSCAN auch auf vielen Daten sehr effizient.

DBSCAN erhält als Eingabedaten eine Menge von numerischen Vektoren, im Falle von Textdaten bietet es sich daher an eine TFIDF-Gewichtung der Feature-Vektoren zu verwenden oder aber mit reinen Worthäufigkeiten zu arbeiten. Die grundlegende Idee ist es dabei diejenigen Punkte zu finden, die besonders viele Nachbarn in ihrer unmittelbaren Umgebung haben. Da diese Punkte sich in besonders „dichten“ Regionen im Feature-Raum befinden, bilden sie die Zentren der Cluster. Eine Besonderheit von DBSCAN ist, dass er Daten verwirft, welche keinem Cluster eindeutig zuzuordnen sind. Diese Daten werden als *Noise*, also Rauschen betrachtet.

ALGORITHMUS Um Bereiche besonders hoher Dichte im Datenraum ausfindig zu machen, benutzt DBSCAN einen zentrumsbasierten Ansatz um Dichte zu definieren. Jeder Datenpunkt wird daher in eine von drei möglichen Kategorien eingeteilt. Die erste Kategorie umfasst alle Punkte innerhalb einer Region von dicht aneinander liegenden Punkten. Da diese Punkte innerhalb von Clustern liegen, werden sie auch als *Core-Points* bezeichnet. Die zweite Kategorie von Punkten umfasst alle Daten, welche am Rand eines Clusters liegen und werden dementsprechend als *Border-Points* bezeichnet. Die übrigen Punkte, die sich also nicht im Zentrum eines Clusters befinden oder zumindest am Rand eines solchen, werden nicht geclustert und daher verworfen. Alle Punkte, die also in diese Kategorie fallen, werden *Noise-Points* genannt.

Etwas formaler ausgedrückt sind Core-Points all diejenigen Datenpunkte, deren Anzahl von Nachbarpunkten, die innerhalb eines vom Nutzer bestimmten Radius *Eps* liegen, einen gegebenen Grenzwert *MinPts* übersteigt. Alle Punkte innerhalb dieses Radius werden auch die Nachbarschaft eines Punktes genannt. Border-Points sind nun alle Punkte, die keine Core-Points sind, allerdings Core-Points in ihrer

39 Ausführlich wird K-Means mit seinen Stärken und Schwächen in Tan u. a. (2005, Kapitel 8.2) behandelt.

Nachbarschaft besitzen. Alle anderen Punkte sind dann Noise-Points und spielen für das Clustern keine Rolle mehr. Ein Beispiel für die Zuteilung von Datenpunkten zu einer der genannten Kategorien ist in der Abbildung 13 zu sehen. Dort sind sechs verschiedene Punkte abgebildet, die geclustert werden sollen. Der Punkt *p1* besitzt in seiner Nachbarschaft mit dem Radius der Länge *Eps* genau vier andere Punkte. Nehmen wir nun an, dass der Wert für den Parameter *MinPts* drei beträgt, dann muss *p1* ein Core-Point sein, da er mindestens drei Punkte in seiner Nachbarschaft besitzt. Für Punkt *p4* gilt dies allerdings nicht, dieser Punkt besitzt lediglich zwei Punkte in seiner Nachbarschaft, da aber einer davon ein Core-Point ist, so zählt *p4* als Border-Point und markiert den Rand eines Clusters. Der Punkt *p6* besitzt keine Punkte in seiner Nachbarschaft und ist damit ein Noise-Point.

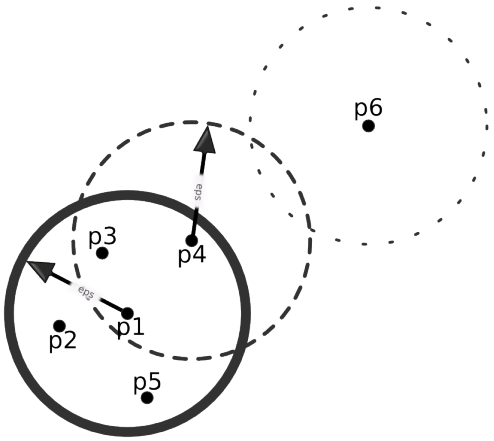


Abbildung 13: Einteilung der Datenpunkte, *p1* ist ein Core-Point, *p4* ein Border-Point und *p6* ein Noise-Point

Der gesamte Algorithmus lässt sich wie folgt in drei Schritte zusammen fassen:

1. Teile alle Punkte in Core-, Border- und Noise-Points ein.
2. Verbinde alle Core-Points die in der selben Nachbarschaft liegen. Jeder Core-Point-Graph stellt nun ein Cluster dar.
3. Ordne alle Border-Points dem ihm am nächsten gelegenen Cluster zu.

Der DBSCAN-Algorithmus ordnet dabei in einem ersten Schritt alle Datenpunkte einer der drei vorgestellten Kategorien zu. Dies geschieht in dem zuerst die Anzahl der Nachbardaten für jeden Punkt ermittelt wird. Wird dabei der Grenzwert *MinPts* überschritten, kann der Punkt gleich als Core-Point markiert werden. Da nun alle Core-Points bekannt sind, können nun die Border-Points ermittelt werden. Dafür muss für die übrigen Punkte überprüft werden, ob sie einen Core-Point in ihrer Nachbarschaft besitzen. Ist dies der Fall, handelt es sich um einen Border-Point, wenn nicht, können sie als Noise-Points verworfen werden. Als nächstes werden alle Core-Points verbunden, die sich innerhalb der *Eps*-Nachbarschaft befinden. Dies lässt sich effizient mit einer Breitensuche über die Core-Points realisieren, wobei immer dann eine Kante zwischen zwei Core-Points angenommen wird, wenn dessen

Entfernung im Feature-Raum weniger als der angegebenen *Eps*-Wert beträgt. Auf diese Weise entsteht ein (graphentheoretischer) Wald, wobei jeder in ihm enthaltene Graph von verbunden Core-Points nun einen Cluster darstellt. Als letztes werden nun die Border-Points dem Cluster des ihm am nächsten gelegenden Core-Points zugeordnet.

Für den DBSCAN-Algorithmus müssen zwei Parameter gewählt werden. Ester u. a. (1996) schlagen dabei ein heuristisches Verfahren vor um *Eps* und *MinPts* des „thinnest cluster" einer Datenbank zu finden. Dabei wird eine Funktion *k-dist* definiert, welche die Distanz eines Punktes zu seinem k-ten Nachbarn berechnet. Wählt man nun ein konstantes k und berechnet für alle Punkte diese Distanzen mittels *k-dist*, welche in aufsteigender Reihenfolge in einem Graph dargestellt werden, erhält man einen sog. *K-Dist-Graphen* oder auch *sortierten K-Dist-Graphen*. In Abbildung 14 ist ein Beispiel für einen K-Dist-Graphen gezeigt. Dieser Graph hilft Aussagen über die Dichteverteilung der Daten zu machen. Wählt man einen beliebigen Punkt p auf dem Graphen und setzt eine absteigende Sortierung der Distanzen voraus, dann sind alle Punkte, welche sich im Graphen rechts vom Punkt p befinden, sehr wahrscheinlich Core- oder Border-Points und alle Punkte links von Punkt p sind wahrscheinlich Noise-Points. Auf diese Weise lässt sich ein geeigneter Parameter finden, um die Anzahl der Noise-Points gering zu halten, aber dennoch sinnvolle Cluster zu erhalten. Ester u. a. (1996) schlagen dabei vor, den Punkt p im *„first "valley" of the sorted k-dist graph"* zu suchen. Für den Beispielgraphen wäre dies etwa ein *Eps* zwischen $0,4$ und $0,5$.

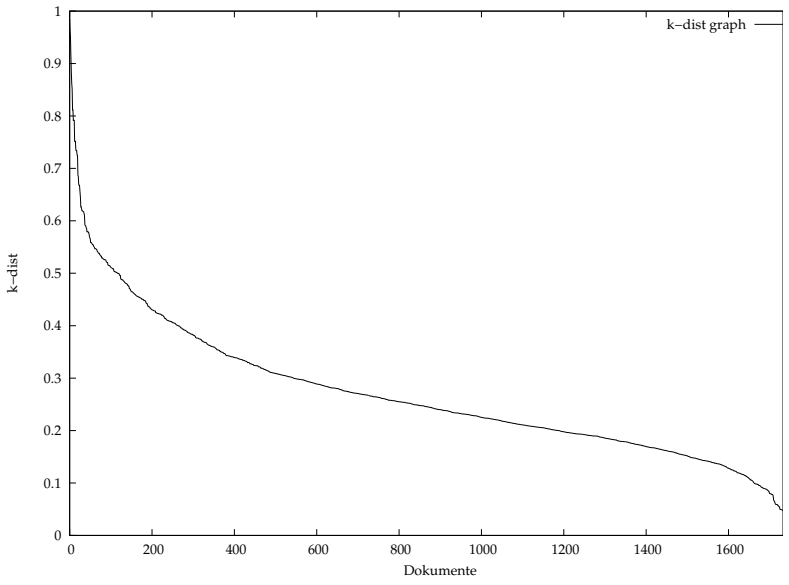


Abbildung 14: Beispiel für K-Dist-Graphen

DISTANZMASS Dichte-basierte Cluster-Verfahren wie DBSCAN benutzen für gewöhnlich das euklidische Distanzmaß um die Abstände zwischen den Datenpunkte anzugeben. Dies mag für viele Einsatzgebiete ausreichend sein, allerdings wird speziell bei Text-Mining mit sehr hohen Dimensionen gerechnet. Da jedes Wort eine Dimension darstellt,

ist es nichts Ungewöhnliches mehr als 10000 Wörter zu haben, welche den Feature-Raum aufspannen.

Aus diesen Eigenschaften der Eingabedaten bei Text-Mining-Applikationen ergibt sich das Problem, dass für jede Dimension, welche die zu clusternden Daten besitzen, das Volumen des Feature-Raums exponentiell wächst. Wenn also die Anzahl der Punkte im Raum nicht auch exponentiell wächst, dann nimmt die Dichte der Datenpunkte sehr schnell ab. Dies wiederum wirkt sich negativ auf die Perfomanz des Cluster-Algorithmus aus. Denn wenn alle Dokumente im Feature-Raum weit voneinander entfernt sind, verringern sich die Dichteunterschiede und der Begriff der Nähe verliert die Bedeutung in einem solchen Raum. Ein weiteres Problem ist, dass im Falle von Text-Mining-Anwendungen die Vektoren nicht nur sehr groß sind, sondern erfahrungsgemäß auch viele Nullwerte beinhalten, so dass ein Maß für die Ähnlichkeit von Vektoren gewünscht ist, welches mit großen Vektoren umgehen kann und die Nullwerte ignoriert.

Der klassische Lösungsansatz, um diesem Problem zu begegnen, ist es anstatt dem euklidischen Längenmaß die Datenpunkte mit der Cosinusdistanz zu messen. Die Cosinusdistanz (oder auch Cosinus-Ähnlichkeit) setzt sich aus dem Punktprodukt geteilt durch den multiplizierten Betrag beider Vektoren zusammen, siehe Gleichung 3.18 für die genaue mathematische Definition der Distanz für zwei Vektoren x und y. Auf diese Weise wird die Entfernung durch den Winkel bestimmt, das Ergebnis ist so normalisiert, dass es zwischen 0 (absolut unterschiedlich) und 1 (gleich) liegt[40]. Dies hat den positiven Nebeneffekt, dass der *eps*-Radius von DBSCAN unabhängig von der eigentlichen Dimension der Daten gewählt werden kann, was im euklidischen Raum nicht möglich wäre. Die Haupteigenschaft der Cosinusdistanz ist es allerdings Vektoren unabhängig ihrer Länge betrachten zu können. Das erlaubt es zwei Vektoren mit Werten an der selben Stelle (also gleichen Wörtern) auch gleich zu behandeln. Die Cosinusdistanz ist deshalb besonders im Bereich des Text-Minings sehr beliebt und findet dort auch weite Anwendung [41].

$$\mathrm{cossim} = \frac{x \cdot y}{|x| \cdot |y|} = \frac{\sum_i x_i \cdot y_i}{\sqrt{\sum_i (x_i^2)} \cdot \sqrt{\sum_i (y_i^2)}} \tag{3.18}$$

VOR- UND NACHTEILE Da DBSCAN einen Dichte-basierten Ansatz zum Ermitteln der Cluster benutzt, ist das Verfahren recht unanfällig für Rauschen und kann Cluster in verschiedenen Größen und Formen erkennen. Allerdings weist der Algorithmus dann Schwächen auf, wenn Daten hochdimensioniert sind, da dafür die klassische Dichtedefinition nur unzureichend genau ist. Desweiteren sind Datensätze mit unterschiedlichen Dichten pro Cluster schwer zu erkennen, da der Nachbarschaftsradius *Eps* einmal vom Benutzer definiert wird und dann unverändert bleibt. Eine weitere Eigenschaft von DBSCAN ist, dass

40 Häufig wird auch dieser Wert von 1 subtrahiert um die Null mit Gleichheit zu assoziieren und 1 mit Ungleichheit, dies entspricht dann auch eher dem Verhalten einer Distanzfunktion und wird auch von uns so durchgeführt.

41 So empfehlen Tan u. a. (2005) das Cosinusdistanzmaß für große, mit vielen Nullen versehende Vektoren, bei Strehl u. a. (2000) findet sich ein Vergleich der Performance verschiedener Cluster-Algorithmen für Text-Mining in Bezug auf verschiedene Distanzmaße, wobei neben der erweiterten Jaccard-Distanz ebenfalls die Cosinusdistanz empfohlen wird.

nicht alle Punkte geclustert werden, diejenigen Vektoren welche nicht genügend Punkte in ihrem Nachbarschaftsradius haben (sog. Noise-Points), werden keinem Cluster zugeordnet. DBSCAN ist allerdings ein sehr effizientes Verfahren und die Implementierung von DBSCAN, wie sie Ester u. a. (1996) beschreiben, hat einen Zeitaufwand von lediglich $O(n \cdot \log(n))$.

3.2.3.3 *Frequent Term-Based Text Clustering*

Frequent Term-Based Text Clustering (FTC) ist ein Cluster-Algorithmus, welcher speziell für das Kategorisieren von Textdokumenten entwickelt wurde. Im Gegensatz zu alternativen Verfahren werden dabei keine Vektoren, also numerische Darstellungen von Textdokumenten, sondern der Text direkt verwendet. Jedes Dokument wird dabei von FTC als eine Menge von Wörtern betrachtet[42]. Der Algorithmus versucht nach Beil u. a. (2002) dabei drei typischen Problemen von Text-Cluster-Algorithmen beizukommen:

1. „Very high dimensionality of the data (10,000 terms dimensions): this requires the ability to deal with sparse data spaces or a method of dimensionality reduction."
2. „Very large size of the databases (in particular, of the world wide web): therefore, the clustering algorithms must be very efficient and scalable to large databases."
3. „Understandable description of the clusters: the cluster descriptions guide the user in the process of browsing the clustering and, therefore, they must be understandable also to non-experts."

ALGORITHMUS Jeder Cluster wird durch FTC von einer Menge an Wörtern, einem sog. *Term-Set*[43] repräsentiert. Der eigentliche Cluster besteht dann aus allen Dokumenten, welche alle Wörter des Term-Sets enthalten, also wo das Term-Set eine Untermenge des Dokuments ist. Die Term-Sets bestehen dabei aus Wörtern, welche statistisch besonders häufig zusammen in Dokumenten auftreten. Wenn also Dokumente, in denen von „worm" die Rede ist, häufig auch „e-mail" als Wort vorkommt, so könnte dann „worm" und „e-mail" zusammen ein Term-Set darstellen und alle Dokumente welche ebenfalls diese Wörter enthalten einen Cluster.

Beil u. a. (2002) stellen einen Greedy-Algorithmus vor, der Term-Sets findet, welche besonders gut unterschiedliche Dokumentencluster repräsentieren. Als Ergebnis wird also eine Liste von Term-Sets zurückgegeben, welche Cluster repräsentieren, die sich möglichst gering überlappen.

Sei $D = (D_1, D_2, \ldots, D_n)$ ein Corpus mit n verschiedenen Dokumenten, welche durch die Wörter repräsentiert werden, die im Dokument vorkommen, dann ist T die Menge aller Wörter in allen Dokumente von D. Es lässt sich dann die Abdeckung (engl. coverage) definieren,

42 Dies ist natürlich nur die halbe Wahrheit, theoretisch könnte man die Dokumente auch als ein Feature-Vektor mit binären Werten darstellen, da der Algorithmus aber viele Operationen auf Mengen verwendet, ist diese Darstellung der Daten nicht zu empfehlen.

43 Der Begriff Term-Set wird eigentlich nur im Zusammenhang mit Textdaten verwendet, eine allgemeinere Bezeichnung von Mengen aus Daten mit gemeinsamen, statistischen Eigenschaften ist *Item-Set*. Das Finden von Item-Sets ist eine bekannte Aufgabe des Data-Minings.

welche die Menge aller Dokumente beschreibt, die eine Menge von Wörtern $S \subseteq T$ beinhaltet, wie in Formel 3.19 dargestellt.

$$cov(S) = \{D_i \in D | S \subseteq D_i\} \tag{3.19}$$

Nun seien $F = \{F_1, F_2, \ldots F_k\}$ alle Term-Sets von D, die mindestens so viele Dokumente in ihrer Abdeckung haben wie ein benutzergewählter Parameter $0 \leqslant minsupp \leqslant 1$ (minimal support). Die Menge F stellt damit die frequenten Term-Sets von D bezüglich *minsupp* da und ist formal in der Gleichung 3.20 beschrieben.

$$F = \{F_i \subseteq T || cov(F_i)| \geqslant minsupp \cdot |D|\} \tag{3.20}$$

Das Finden der frequenten Term-Sets kann ein sehr aufwendiges Unterfangen sein, da die Anzahl aller möglichen Kombinationen an Wörtern innerhalb von T mit der Gesamtanzahl der Wörter exponentiell steigt. Wenn allerdings die Abdeckung eines Term-Sets S_i über *minsupp* liegt, dann gilt dies auch für alle Term-Sets, die eine Obermenge von S_i sind. Diese Monotonieeigenschaft lässt sich beim Finden der frequenten Term-Sets ausnutzen. Es gibt mehrere unterschiedliche Algorithmen, die dieses tun um möglichst effizient Term-Sets zu generieren. Einer der prominentesten Vertreter solcher Algorithmen ist dabei der Apriori-Algorithmus. Dieser kann als ein Standardverfahren für das Finden von frequenten Term-Sets angesehen werden. Seine Laufzeitkomplexität beträgt allerdings $O(n^2 d)$, wobei n die Anzahl unterschiedlicher Wörter im gesamten Corpus bezeichnet und d die Anzahl der Dokumente[44]. Es gibt zwar verschiedene Verbesserungen für Apriori, welche die Laufzeitkomplexität mit Hilfe von Vorverarbeitungsschritten weiter reduzieren können, diese Algorithmen verschlechtern aber entweder die Qualität des Clusterings oder haben selbst eine hohe Laufzeitkomplexität[45].

Besitzt man eine Menge F von frequenten Term-Sets, so besteht der nächste Schritt daraus, innerhalb dieser Menge dasjenige Term-Set zu finden, welches die geringste Überlappung hat. Um ein sinnvolles Maß für die Überlappung eines Term-Sets zu berechnen, schlagen Beil u. a. (2002) folgende Definition vor. Sei f_j die Anzahl aller frequenten Term-Sets, in deren Abdeckung sich das Dokument D_j befindet, also:

$$f_j = |\{F_i \in R | F_i \subseteq D_j\}| \tag{3.21}$$

Die Menge R ist dabei eine Untermenge von F und wird in jedem Cluster-Schritt verkleinert. Sie beinhaltet alle verbleibenden frequenten Term-Sets, Beil u. a. (2002) beschreibt sie als „subset of remaining frequent term sets, i.e. the difference of F and the set of the already selected frequent term set“. Um nun ein Maß für die Überlappung von

44 Genauere Angaben zur Berechnung der Zeitkomplexität von Apriori, mit Hilfe der Verwendung verschiedener Datenstrukturen, findet sich in Hegland (2007).

45 Mehr Informationen zu möglichen Vor- und Nachbearbeitungsschritten finden sich bei Jovanoski u. Lavrac (2001).

Cluster zu finden wird empfohlen die Berechnung der *entropy overlap* zu nutzen, welche besonders gute Cluster-Ergebnisse erzielt:

$$EO(C_i) = \sum_{D_j \in C_i} -\frac{1}{f_j} \cdot \ln(\frac{1}{f_j}) \tag{3.22}$$

Die Überlappung ist 0 solange alle Dokumente im Cluster durch nur ein frequentes Term-Set dargestellt werden, steigt hingegen die Anzahl von zusätzlichen Term-Sets, so steigt auch der Grad der Überlappung. In jedem Cluster-Schritt wird nun dasjenige Term-Set ausgewählt, welches die geringste Überlappung hat. Dieses Term-Set stellt nun ein Cluster da und wird aus der Menge verfügbarer frequenter Term-Sets R entfernt. Ebenfalls werden alle Dokumente aus D entfernt, welche sich in der Abdeckung des ausgewählten Term-Set befinden, sowie alle Cluster-Kandidaten, welche eines der entfernten Dokumente in ihrer Abdeckung haben. Diese Dokumente werden damit auch aus der Abdeckung der noch verfügbaren Term-Sets entfernt, um nichtüberlappende Cluster zu gewährleisten. Der Vorgang wird fortgesetzt, bis die ausgewählten Term-Sets, also der Cluster-Kandidaten, alle Dokumente abdecken. Der Ablauf des Algorithmus ist hier noch einmal in Pseudo-Code dargestellt.

- $C = \{\}$, $R = F$ für ein gegebenes *minsupp*
- Solange $cov(C) \neq |D|$
  - Berechne die Überlappung aller Mengen in R
  - S = Term-Set mit minimaler Überlappung aus R
  - $C = C \cup \{S\}$
  - $R = R \cap \{S\}$
  - entferne alle Dokumente in $cov(S)$ aus D
  - entferne alle Term-Sets in R die in ihrer Abdeckung Dokumente aus $cov(S)$ haben
- Gebe die Abdeckung aller Elemente in C als Cluster aus

VOR- UND NACHTEILE Die Vorteile dieses Verfahrens sind, das die Anzahl der Cluster selbstständig gefunden wird und eine sinnvolle Beschreibung von Clustern in Form von frequenten Term-Sets zurückgegeben wird. Der Nachteil ist, dass das Finden aller frequenten Term-Sets in einer Anzahl großer Dokumente, sehr aufwendig ist. In diesem Fall muss eine effiziente Implementierung des Suchalgorithmus erfolgen, was mit einem erhöhten Programmieraufwand einhergeht. Vergleiche mit anderen Algorithmen zeigen aber, dass bei einer geringen Anzahl von Wörtern pro Dokument und einer entsprechend effizienten Implementierung klassische Verfahren wie K-Means in Sachen Geschwindigkeit geschlagen werden können, die Clusterqualität aber mit denen von traditionellen Verfahren vergleichbar ist[46].

3.2.3.4 *Zusammenfassung*

Es wurden verschiedene Algorithmen zum automatischen Kategorisieren von Daten vorgestellt. Um die verschiedenen Verfahren besser

46 Dies wurde ebenfalls von Beil u. a. (2002) beobachtet.

vergleichen zu können, hier noch einmal eine Zusammenfassung. Es wurden drei sehr unterschiedliche Verfahren vorgestellt. K-Means ist ein Algorithmus, bei dem eine vorgegebene Anzahl von Clustern gefunden wird, ein Cluster wird durch den Mittelwert aller seiner Vektoren dargestellt. Daraus folgt, dass K-Means lediglich runde Cluster identifizieren wird. DBSCAN hingegen ermittelt die Anzahl der Cluster selbst und kann beliebig geformte Cluster erkennen, der Nachteil ist allerdings, dass DBSCAN von sich aus keine Cluster-Beschreibung bietet. Frequent Term-Based Text Clustering (FTC) ist ein Verfahren, dass speziell für Text-Mining-Anwendungen entwickelt wurde. Im Gegensatz zu den beiden anderen Ansätzen operiert FTC nicht auf numerischen Werten, sondern benötigt lediglich die Wörter, welche in einem Dokument vorkommen. FTC untersucht, welche Wortpaare besonders häufig zusammen auftreten und gibt dies als eine Cluster-Beschreibung zurück. Der Vorteil von FTC ist, dass die Qualität des Clusterings nicht beeinträchtigt wird, wenn viele Dokumente und Wörter verarbeitet werden müssen. Allerdings besitzt der FTC-Algorithmus eine sehr hohe Laufzeitkomplexität für Textdaten mit vielen Wörtern, die nur mit einem hohen Implementierungsaufwand und evtl. Qualitätsverlust verringert werden kann.

Algorithmus	K-Means	DBSCAN	FTC
Laufzeitkomplexität	$O(n \cdot \log(n))$	$O(n \cdot \log(n))$	$O(n^2 \cdot d)$
Implementierungsaufwand	gering	gering	hoch
Eingabedaten	numerische Vektoren	numerische Vektoren	Menge von Wörtern
Clusterbeschreibung	durch Centroid	keine	durch Term-Sets
Vorteil		resistent gegen Rauschen	
Nachteil	Anzahl der Cluster muss vorher feststehen		hohe Laufzeitkomplexität

Tabelle 6: Vergleich der Cluster-Algorithmen

3.2.4 *Auswertungsmethoden*

Nachdem die einzelnen Dokumente jeweils einem Thema zugeordnet wurden, müssen aus den erhaltenen Klassen Schlüsse gezogen werden. Dies kann nur geschehen, wenn der Inhalt der Klassen sowohl auf den Achsen der thematischen Breite als auch der quantitativen Verteilung (Publikationsvolumen) über die Zeit verortet werden kann. Beides erfordert vorherige Überlegungen.

3.2.4.1 *Thema*

Steht einmal eine Zuteilung der Dokumente zu Themengebieten fest, so muss nun noch veranschaulicht werden, welche Dokumente genau zu einem Themengebiet zusammengefasst wurden, bzw. was die Essenz dieser Themengebiete ist. Dies kann natürlich nur gelingen, wenn das verwendete Verfahren eine einigermaßen homogene Kategorie erzeugt hat. Ist dies der Fall, so stellt sich das Problem der *Multi-Document-Summarization*, der automatischen Zusammenfassung von mehreren Dokumenten[47]. Grundsätzlich gibt es dazu zwei verschiedene Lösungsansätze: Die generative Zusammenfassung und die extraktive Zusammenfassung. Bei der generativen Zusammenfassung werden über Methoden der *Information Extraction* Schlüsselinformationen aus den Texten gewonnen und dann mittels computerlinguistischer Textsynthese daraus Zusammenfassungen generiert. Bei extraktiven Zusammenfassungen werden hingegen wichtige und in sich möglichst abgeschlossene Sinneinheiten (Wörter, Wortgruppen, Sätze, Absätze, ...) nach ihrer Repräsentativität bewertet, sortiert und präsentiert.

Wie unsere Leser vielleicht schon erraten haben, erzeugen heutige Systeme für generative Zusammenfassung mehr oder minder unlesbaren Kauderwelsch aus den eingegeben Texten[48] oder sind nur für sehr spezielle Textsorten und Corpora praktisch verwendbar[49]. Dies ist nicht verwunderlich, denn menschliche Sprache ist alles andere als eindeutig lesbar und eignet sich denkbar schlecht um Fakten und Zusammenhänge automatisch zu erfassen. Die extraktiven Zusammenfassungen sind schon erfolgreicher, jedoch fehlen auch in ihnen oft wichtige Informationen aus innertextuellen Diskurselementen. So stellen sie zwar vermeintlich wichtige Textbröckchen dar, aber die Verbindungen zwischen diesen fehlen meist.

Einige der bereits ersonnenen Zusammenfassungs-Systeme beachten bei der Erstellung der Extrakte auch positionelle oder temporale Informationen, sprich – an welcher Stelle im Text sich eine Sinneinheit befindet oder in welcher Reihenfolge die Texte geschrieben wurden (in der Annahme, dass spätere Veröffentlichungen frühere vervollständigen oder korrigieren).

Positionelle Informationen zu verwenden ist sinnvoll, denn in vielen Textsorten wird dem Leser vom Autor bereits eine menschlich generierte Zusammenfassung vorangestellt (wissenschaftliche Artikel) oder wichtige Information bereits gleich zu Anfang dargeboten (Zeitungs-/Newsartikel). Da die Dokumente in unserem größten Corpus, dem IEEE Corpus nach der Permutation der Sätze jedoch keine positionellen Informationen mehr enthält[50], sind all diese Systeme wegen ihrer Voraussetzung für unsere Zwecke nicht geeignet. Ebenso hängen von der Wahl des Clustering-Algorithmus auch die Möglichkeiten zur automatischen Zusammenfassung ab. So liefert z. B. der K-Means-Algorithmus immer eine runde Punktwolke, dessen Mittelpunkt oder der Punkt, welcher der Mitte am nächsten ist, auf natürliche Weise das Thema

47 Einen Einstieg in diese eigenständige Forschungsrichtung hat uns Hovy (2003, Kap. 32) gegeben.

48 Vgl. ebenda, S.585.

49 Wie z. B. die generativen Zusammenfassungen aus Barzilay u. a. (1999) für kurze News-Artikel.

50 Dieser Umstand ist der Tatsache zu verschulden, dass wir den unversehrten Corpus nur bis zu einem sehr frühen Termin aufheben durften. Um eine spätere Wiederholung der Vorverarbeitung zu erlauben, wurden alle Dokumente des IEEE Corpus an ihren Satzgrenzen getrennt und diese Sätze zufällig permutiert. Dabei bleiben die meisten statistischen Eigenschaften erhalten, nicht jedoch positionelle Informationen.

des Clusters repräsentiert[51]. Auch die Art der Vorverarbeitung hat großen Einfluss auf die Möglichkeiten. Bei Zhang u. a. (2004) findet sich z. B. ein Verfahren, welches die wichtigsten Sätze aus einer Menge von Web-Seiten extrahiert. Sie greifen dabei aber auf die in der Vorverarbeitung der Daten gewonnenen Multi-Terms und die Zwischenwerte aus der Cluster-Berechnung (C-Value/NC-Value) zur Bewertung der Multi-Terms zurück. Die Art der Zusammenfassung hängt also eng mit der Art der Clustering zusammen und muss passend für die Vorverarbeitung gewählt werden.

TFIDF-ZUSAMMENFASSUNGEN Werden z. B. bei der Vorverarbeitung die Feature-Vektoren aus TFIDF-Werten zusammengesetzt, so hat man bereits ein enormes Informationspotential. Die Höhe jedes TFIDF-Werts beschreibt, wie charakteristisch das zugrundeliegende Wort für das Dokument ist. Wörter, welche in diesem Dokument häufig auftauchen, in anderen Dokumenten aber nicht, haben einen hohen TFIDF-Wert. Wörter, die entweder in diesem Dokument nicht oder auch in vielen anderen Dokumenten auftauchen, haben einen niedrigen TFIDF-Wert. Ein an Zhang u. a. (2004) angelehntes Verfahren[52] könnte so aussehen, dass aus jedem Cluster drei Zusammenfassungen erzeugt werden. Diese sind:

- Eine Wort-Zusammenfassung, welche aus den wichtigsten Terms aus allen Dokumenten des Clusters besteht.
- Eine Titel-Zusammenfassung, welche aus den wichtigsten Dokumenten-Titeln des Clusters besteht.
- Eine Satz-Zusammenfassung, welche aus den wichtigsten Sätzen aus allen Dokumenten des Clusters besteht.

Für jede dieser Zusammenfassungen lässt sich einfach jeweils eine Menge aller Terms, Titel und Sätze aller Dokumente des Clusters erzeugen. Wichtig ist nun die Sortierung der Reihenfolge. Für die Sortierung der Wörter werden aus jedem Dokument die n Wörter mit dem größten TFIDF-Wert gewählt und gespeichert. Ist in einem anderen Dokument das gleiche Wort ebenfalls unter den n größten Wörtern, so wird der neue TFIDF-Wert auf den alten addiert. Es entsteht so ein verkürzter Feature-Vektor für den gesamten Cluster. Aus diesem können dann wiederum die n Worte mit dem höchsten gesamt-TFIDF gewählt und dargestellt werden.

Für die Titel-Zusammenfassungen müssen die Titel der Dokumente bewertet werden. Dafür werden für jedes Dokument die Einträge im Feature-Vektor des Dokuments betrachtet, welche auch in die Wort-Zusammenfassungen aufgenommen wurden. Je öfter ein solches Keyword im Dokument vertreten ist (also sein TFIDF-Wert größer Null ist), desto repräsentativer ist das Dokument auch für den Cluster. Die Anzahl der Vorkommenden Keywords in den Dokumenten wird also gezählt und ihre Titel nach dieser Reihenfolge sortiert. Die ersten n Titel werden wiedergegeben.

Für die Satz-Zusammenfassungen müssen in ähnlicher Weise die Sätze in den Dokumenten bewertet werden. Dabei sei die Repräsentativität

51 In der Tat gibt es z. B. den MEAD-Summarizer von Radev u. a. (2004) der sich diese Tatsache zu Nutze macht.

52 Zhang u. a. (2004) arbeiten mit C-Value/NC-Value, Multi-Terms und gewichten Sätze komplizierter als im Folgenden beschrieben.

eines Satzes um so höher, je repräsentativere Wörter in ihnen vorkommen. Es wird also für jedes Dokument jeder Satz nach vorkommenden Terms mit hohem TFIDF-Wert durchsucht. Der höchste vorkommende TFIDF-Wert wird als Bewertung dieses Satzes verwendet. Die n wichtigsten Sätze jedes Dokuments werden einer Liste mit wichtigen Sätzen hinzugefügt. Aus dieser werden dann wiederum die n wichtigsten Sätze des ganzen Clusters gewählt.

Dieses Verfahren ist bei gegebener TFIDF-Vorverarbeitung einfach implementierbar. Eine formale Untersuchung der Leistungsfähigkeit steht aus. Da das Verfahren aber sehr ähnlich zu anderen ist, ist eine akzeptable Leistungsfähigkeit zu erwarten. Ebenso ist zu beachten, dass es bei der gegebenen Größe der zu verarbeitenden Text-Corpora eine gute Geschwindigkeit zeigt. Einziger Schwachpunkt ist, dass keinerlei Redundanzen reduziert werden. Bei der geringen Textmenge, die dem Betrachter aus hunderten von Dokumenten dargeboten wird, stellen ähnliche Terms, Titel sehr ähnlicher Dokumente oder sehr ähnliche Sätze mit redundanten Informationen eventuell eine schlechte Wahl dar, da so die Abdeckung des ganzen Themas nicht immer gewährleistet werden kann.

K-MEANS ZUSAMMENFASSUNGEN Ein anderes Verfahren zur Zusammenfassung, gerade von Density-Based Clustern, könnte die zum Cluster gehörenden Dokumente noch einmal mit einem beliebigen k mit dem oben beschriebenen K-Means-Algorithmus clustern. Die so entstehenden, runden und unterschiedlich großen Sub-Cluster haben durch die Eigenheiten des K-Means-Algorithmus jeweils einen natürlichen Mittelpunkt. Aus diesem ließen sich repräsentative Zusammenfassungen generieren. Das zusätzliche K-Means-Clustering erzeugt jedoch einen enormen neuen Berechnungs- und Implementierungsaufwand. Es ließe sich jedoch vermutlich eine hohe Qualität der Zusammenfassungen erreichen[53].

3.2.4.2 *Publikationsvolumen und Zeit*

Wie im Ziel der Arbeit vorgesehen, können mit einer gegebenen Zuordnung der Dokumente zu Themen Graphen der quantitativen Verteilung der Publikationshäufigkeit über die Zeit erstellt werden. Die Gestaltung der Graphen hat maßgeblich Einfluss auf die Art der Aussagen, die abgeleitet werden können. Zuvorderst sollte bedacht werden, ob die quantitative Verteilung über die Zeit an der Gesamtpublikationshäufigkeit der Datenquelle normalisiert werden soll. Der Vorteil einer Normalisierung ist, dass ein solcher Graph eher die relative Aufmerksamkeit einer Gemeinschaft für ein Thema beschreibt. So kann sich zwar das gesamte Aufkommen an Publikationen stark erhöhen, der relative Anteil für ein Thema aber gleich bleiben. Dieser Vorteil ist natürlich auch gleichzeitig ein Nachteil, denn auch die absoluten Publikationshäufigkeiten können von Interesse sein, denn sie überlassen zusammen mit einer Kurve der Publikationshäufigkeit die Interpretation dem Leser. Wenn zu einer Normalisierung gegriffen wird, so ist zudem noch

53 Ein solches Verfahren wäre eine Kombination der Vorschläge aus Goldstein u. a. (2000), die vorerst nur vorgeschlagen haben eine breit gestreute Dokumentenmenge vor der Zusammenfassung erst zu Clustern, sowie Radev u. a. (2004), die K-Means Centroids verwenden um Cluster zusammenzufassen, und auch Boros u. a. (2001), die die Sätze der Dokumente nach Ähnlichkeit clustern um eine große Abdeckung des Themengebiets zu erhalten.

wichtig, ob nur an der Publikationshäufigkeit der Dokumente in den Klassen oder auch an Dokumenten ohne zugewiesene Klasse normalisiert wird. Dokumente ohne zugewiesene Klassen können je nach verwendeten Algorithmen wirklich uninteressante Dokumente sein, die eigentlich nicht zur Gesamtpublikationshäufigkeit gezählt werden können (Spam, Werbung, ...) oder sie können nur durch die Eigenschaften des Algorithmus nicht zugewiesen werden, könnten aber von einem Menschen durchaus einem Thema zugeordnet werden und sollten so in den Gesamtpublikationshäufigkeit eingehen.

Desweiteren sind in einigen Text-Mining-Publikationen, speziell in solchen die sich mit Themenwandeln beschäftigen, die Themenverteilung über die Zeit aufsummiert dargestellt[54]. Dies ermöglicht dem Leser die schnelle Übersicht über absolute Häufigkeiten und ermöglicht gleichzeitig die Beurteilung der relativen Anteile. Allerdings können in der aufsummierten Darstellung auch leicht kleinere aber interessante Schwankungen übersehen werden. Ebenso stellt sie ganz profan bestimmte Anforderungen an den Reproduktionsmechanismus, der geeignet sein muss verschiedenfarbige Flächen darzustellen.

Wichtig ist ebenso, welche Klassen in eine Auswertung mit eingehen. Einige Algorithmen erzeugen auch bei guter Parametrisierung, gerade auf großen Corpora, immer noch über einhundert verschiedene Klassen. Es ist zu erwägen nur die quantitativ größten dieser Klassen zu wählen, da die quantitative Größe ein Maß für die Aussagekraft und Bedeutung der Volumen-Zeit Graphen ist. Dabei können jedoch kleinere aber sehr charakteristische Klassen verloren gehen.

3.3 WAHL DER METHODE

In diesem Abschnitt wird die letztendlich verwendete Methode zur Analyse der Textquellen vorgestellt. Dabei werden zuerst die verschiedenen zur Verfügung stehenden Techniken aus dem vorhergehenden Abschnitt diskutiert und die Entscheidung für bestimmte Verfahren begründet. Eine schematische Darstellung der gesamten Verarbeitungskette findet sich in Abbildung 15.

Um die gegebenen Textquellen zu analysieren und deren Inhalt thematisch zu gruppieren, haben wir uns für ein Cluster-Verfahren entschieden, wobei für jede vorhandene Textquelle über den gesamten zur Verfügung stehenden Zeitraum ein separates Clustering erstellt wird. Um keine Annahmen über die konkrete Anzahl von Cluster machen zu müssen und möglichst effizient die große Anzahl der Daten zu verarbeiten, wird der Cluster-Algorithmus DBSCAN verwendet. Das heißt also, dass für jede betrachtete Datenquelle eine Menge von Clustern gesucht wird, welche sich über den gesamten Zeitraum erstrecken. Über die zeitliche Verteilung der Beiträge innerhalb eines Clusters lässt sich dann auch eine zeitliche Entwicklung des Themas ableiten auf das sich der Cluster bezieht. Im Abschnitt 3.3.1 wird nocheinmal genauer auf die Entscheidung für das Clustern und gegen das Klassifizieren eingegangen und auch die konkrete Cluster-Strategie begründet. Gleich darauffolgend findet sich in 3.3.2 auch eine genaue Begründung, warum wir uns für DBSCAN als Cluster-Algorithmus entschieden haben.

Um die einzelnden Texte in den Quellen gut verarbeiten zu können, müssen diese noch in eine konsistente Form gebracht werden, dazu zählt u.a. das Entfernen von Duplikaten und anderen störenden

54 Zur kognitiven Analyse einer solchen Darstellung siehe z. B. Havre u. a. (2000).

Merkmalen wie zum Beispiel E-Mail-Signaturen, welche die Qualität der Textanalyse einschränken könnten. Mehr Information dazu finden sich in den Absätzen 3.3.3. Sobald dieser Schritt erledigt ist, können die Textdaten in numerische Werte während der sog. Vorverarbeitung transformiert werden. Dabei werden zuerst die Texte durch einen Tokenizer in einzelne Wörter zerlegt aus denen dann statistische Merkmale zu dem Inhalt des Textes generiert werden. Der Tokenizer trennt den Text an allen Sonderzeichen und Leerstellen und verwirft sämtliche Zahlen. Man erhält also für jeden Text nun eine Liste aus Worten, den sog. Features. Aus dieser Wortliste werden alle Stop-Words entfernt, also Begriffe ohne semantische Bedeutung für den Text, wie z.B. Präpositionen oder Pronomen. Anschließen werden die übriggebliebenen Wörter auf ihren Wortstamm reduziert um so verschiedene Konjugationen und Deklinationen eines Wortes zusammenfassen zu können. Während des gesamten Vorgangs wird dabei kein Unterschied zwischen Groß- und Kleinschreibung gemacht. Aus der nun stark bearbeiteten Wortliste werden anschließend alle Wörter verworfen, die entweder zu lang oder zu kurz sind um sinnvolle Wörter darstellen zu können oder aber zu selten im Text vorkommen (Word-Threshold); auf diese Weise kann die Anzahl der OCR- und Rechtschreibfehler stark reduziert werden. Nun existiert für jedes Dokument eine Liste von Begriffen, die dort enthalten sind, das sog. lokale Dictionary. Daraus lässt sich nun auch leicht eine Liste von allen Wörtern aller Dokumente, also das globale Dictionary erstellen. Diese Information wird genutzt um einen statistischen Messwert zu berechnen, der die Wichtigkeit eines Wortes für den Inhalt eines Dokuments wiederspiegelt, dieser Messwert ergibt sich aus dem TFIDF-Wert eines Wortes. Man erhält also für jede Wortliste eines Dokuments einen numerischen Vektor, mit dem nun der Cluster-Algorithmus umgehen kann. Die genaue Vorgehensweise bei der Vorverarbeitung wird detailliert im Abschnitt 3.3.4 vorgestellt.

Die Vektoren aller Dokumente werden verwendet um mit DBSCAN in Clustern zusammengefasst zu werden. Um gute Parameter für den Cluster-Algorithmus zu finden, wird für eine Auswahl verschiedener Parameter Cluster erstellt und diejenigen Parameter gewählt, welche die Dokumente am geeignetsten im Bezug auf die Standardabweichung der Clustergrößen und der Anzahl der entstandenen Noise-Points zusammengefasst haben. Diese Vorgehensweise wird in 3.3.5 noch einmal vertieft behandelt.

Aus der Cluster-Einteilung, der Feature-Matrix und den ursprünglichen Corpora werden dann kurze Zusammenfassungen der Cluster-Inhalte generiert und dazugehörige Graphen des anteiligen Publikationsvolumens über die Zeit erstellt (Abschnitt 3.3.6).

3.3.1 *Clustern vs. Klassifizieren*

Das Kategorisieren von Daten mittels Cluster-Algorithmen oder Klassifizierungsverfahren haben verschiedene Vor- und Nachteile. Klassifikation von Daten ist besonders dann geeignet, wenn bereits bekannt ist, welche Klassen es gibt und in diese dann neue Daten eingeordnet werden sollen. Der Nachteil von Klassifikation ist allerdings, dass erst in einem aufwendigen Trainingsschritt ein Modell erstellt werden muss.

Cluster-Verfahren hingegen wählen selbst aus, welche konkreten Kategorien sich in den gegebenen Daten befinden. Einige Varianten

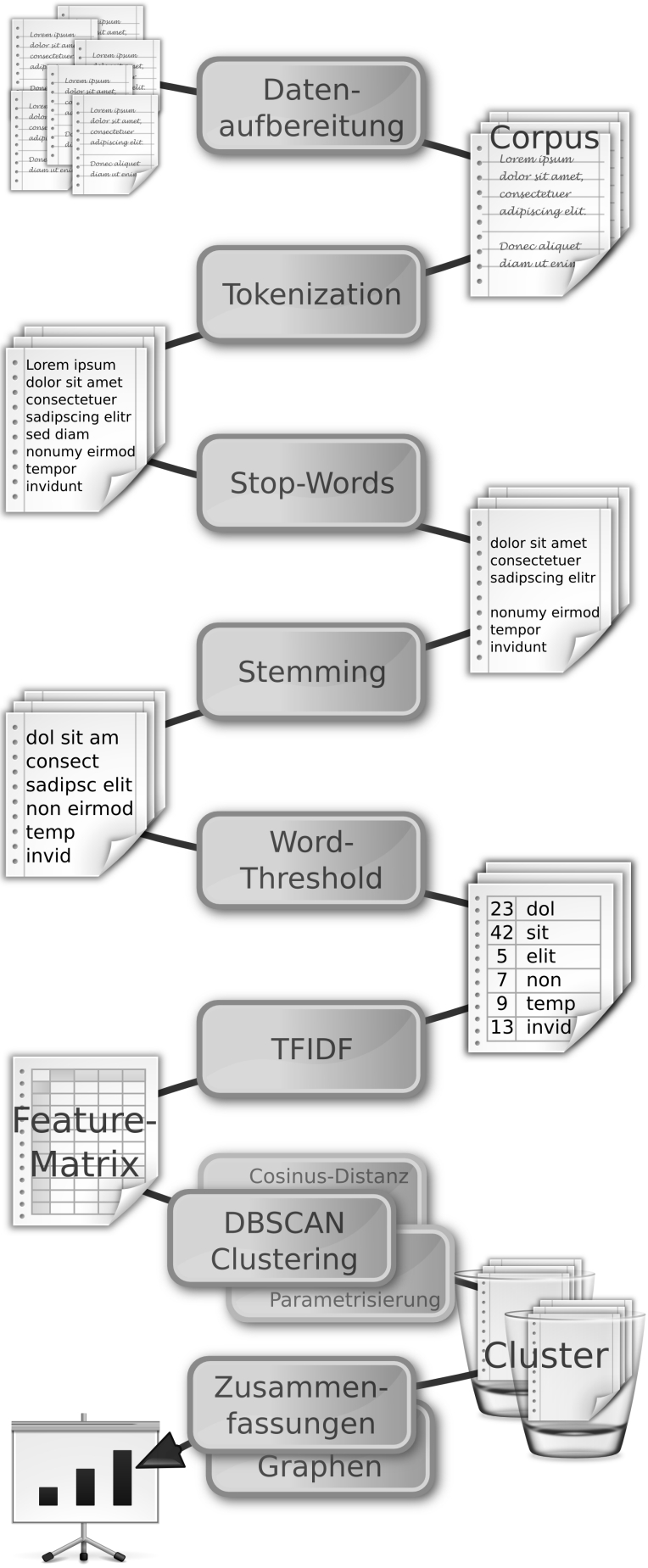


Abbildung 15: Zusammenfassung der Analysemethode

können auch die Anzahl solcher Kategorien eigenhändig bestimmen. Der Nachteil ist allerdings, dass bei den meisten Ansätzen in einem extra Nachbearbeitungsschritt herausfinden muss, warum gewissen Daten in einem Cluster zusammengefasst wurden um so einen Bezeichner über das inhaltliche Thema eines Clusters konstruieren zu können.

Möchte man nun den Diskurs über Informationssicherheit anhand von verschiedenen Textquellen mit Hilfe von Klassifikationsverfahren untersuchen, so muss zuerst geklärt werden, in welche Art von Klassen die Texte eingeteilt werden sollen. Speziell auf die Wellenthese von Solms bezogen, wäre es möglich eine Klasse für jede Welle einzurichten. Nachdem alle Dokumente aus den Textquellen klassifiziert wurden, lässt sich der Verlauf der Veröffentlichungen über die Zeit einer jeden Klasse analysieren. Der Nachteil an diesem Ansatz ist aber, dass es keinen Trainingssatz an Textquellen gibt, die sich bereits einem der drei Wellen eindeutig zuordnen lassen. Ein weiteres Problem ist, dass mit diesem Ansatz lediglich festgestellt werden kann, ob es tatsächlich einen Diskurs gab, der sich in drei Wellen ereignete. Sollte es keine Wellen geben oder es lediglich zwei oder gar vier Wellen gegeben haben, so lässt sich dies nicht mit Hilfe von Klassifizierern herausfinden.

Unsere Problemstellung erlaubt also keinen Rückschluss auf die Anzahl von Kategorien die bei der automatisierten Gruppierung der Daten zu erwarten sind. Das Thema „Informationssicherheit“ lässt sich zwar in viele verschiedene Unterthemen aufteilen, aber die endgültige Anzahl der Themen hängt stark von der Auflösung ab, mit der man das Thema Informationssicherheit betrachtet. So stellt sich z.B. die Frage, ob das allgmeine Unterthema „Verschlüsselung von Daten“ noch weiter aufgeteilt werden kann in „Verschlüsselung von E-Mails“, „Sicherheit von Kryptoverfahren“ und ähnlichen weiteren Kategorien.

Um also zusammenhängende Themen in den Dokumenten zu finden, muss ein maschinelles Lernverfahren in der Lage sein diese auch selbstständig ausfindig zu machen, damit fallen alle Ansätze, die vorgegebene Kategorien benötigen, aus. Dies betrifft vor allem Klassifizierungsverfahren, aber auch spezielle Cluster-Algorithmen wie K-Means, die Dokumente in eine vorgegebene Anzahl von Kategorien einteilen.

Das Clustern der Textdokumente kann auf verschiedene Weise erfolgen. Ein Ansatz wäre das Clustern von Daten in bestimmten Zeitfenstern. So ließe sich z.B. für jeden Monat des zu betrachtenden Zeitraums ein Clustering bilden. Auf diese Weise lassen sich die Veröffentlichungen während dieser Zeit thematisch gruppieren und man kann so die Entstehung neuer Diskussionsthemen über die Zeit entdecken, sowie die Anzahl von Textquellen für jedes Thema über die Zeit darstellen. Der Vorteil von diesem Ansatz wäre, dass nur jeweils wenige Dokumente in einem Schritt geclustert werden müssen, was bei Verfahren mit hoher Komplexität, wie FTC, die Performance deutlich verbessern kann. Auf der anderen Seite ist es schwer Clustern ein genaues Thema zuzuordnen. Möchte man aber die Entwicklung von Themen über die Zeit betrachten, so ist es nötig ein Vergleichsmaß zu haben, das angibt, wie ähnlich zwei Cluster sind um diese ggf. miteinander zu identifizieren zu können. Ein nachträgliches Identifizieren von ähnlichen Clustern, das nicht automatisiert geschieht, schränkt aber wiederum die Objektivität des Verfahrens und damit die Aussagekraft des Ergebnis stark ein.

Um dieses Problem zu umgehen, ist es möglich über den gesamten betrachteten Zeitraum Cluster zu suchen und so Kategorien von

Textdaten zu erstellen, welche Dokumente mit unterschiedlichem Veröffentlichungsdatum zusammenfassen. Nachträglich ließe sich dann der zeitliche Verlauf eines jeden Clusters aus dem in ihm enthaltenen Textdaten ablesen und grafisch darstellen. Der Nachteil ist allerdings, dass viele Dokumente aus den Textquellen gleichzeitig geclustert werden, was ein sehr rechenintensiver Vorgang ist. Dennoch haben wir uns im Rahmen dieser Arbeit für dieses Verfahren entschieden, da so eine möglichst objektive Aussage der Clusterergebnisses erreicht werden kann.

Wie genau die Daten vorverarbeitet werden, hängt u.a. stark vom verwendeten Clusterverfahren ab. So benötigt FTC lediglich die Information ob ein Wort in einem Dokument vorkommt, aber nicht wie häufig dies geschieht. Prinzipiell ist aber zu erwarten, dass die Daten stark verrauscht sein werden. Die älteren Journalausgaben sind mit Hilfe von OCR digitalisiert worden, also ein Verfahren, dass durchaus Fehler machen kann. Bei den E-Mail- und Newsgroupdatensätzen muss hingegen mit Tippfehlern gerechnet werden, weshalb unabhängig vom verwendeten Cluster-Verfahren verschiedene Arten von Störmerkmalen aus den Text-Corpora vor der Vorverarbeitung entfernt und weitere Dimensionsreduktions-Maßnahmen verwendet werden müssen.

3.3.2 *Auswahl des Clusterverfahrens*

Aus der Begründung der Entscheidung für ein Cluster-Verfahren im Abschnitt 3.3.1 folgt, dass der Algorithmus in der Lage sein muss selbstständig die Anzahl der Cluster festzulegen. Es kommen daher nur DBSCAN und FTC in Frage, da bei K-Means eine erwartete Cluster-Anzahl angegeben wird. Für DBSCAN spricht seine Eigenschaft Cluster beliebiger Größe und Form zu finden, sowie seine Performance für viele Dokumente und die Einfachheit seiner Implementierung. Der Nachteil von DBSCAN ist die Schwierigkeit hochdimensionale Daten ordentlich zusammenzufassen, allerdings lässt sich dieses Problem mit der Verwendung der Cosinusdistanz und verschiedenen Maßnahmen zur Dimensionsreduktion in Griff bringen.

FTC ist speziell für das Clustern von Textdokumenten kreiert worden. Die Vorteile sind, dass Cluster ähnlich wie bei DBSCAN automatisch gefunden werden ohne das eine spezielle Anzahl genannt werden muss und dass zusätzlich automatisch eine Liste von Wörtern generiert wird, welche die Cluster gut beschreiben. Der Nachteil von FTC ist der hohe Implementierungsaufwand. Hinzu kommt, dass das Finden von frequenten Term-Sets sehr aufwendig und rechenintensiv ist. Standardverfahren wie Apriori sind zwar durch die Ausnutzung der Monotonieeigenschaft frequenter Term-Sets recht Laufzeiteffizient, verbrauchen aber sehr viel Speicher bei der Suche in einer großen Anzahl von Dokumenten mit vielen Wörtern.

Aus diesem Grund haben wir uns für DBSCAN entschieden, diese Entscheidung beeinflusst nun auch die Wahl der Vorverarbeitungsschritte. So betrachtet DBSCAN numerische Vektoren, während FTC lediglich eine Menge von Wörtern benötigt.

DBSCAN operiert gewöhnlicherweise im euklidischen Raum, da wir es aber mit hochdimensionalen Vektoren zu tun haben, bei denen viele Komponenten Null sind, benutzen wir die Cosinusdistanz um die Ähnlichkeit zwischen Featurevektoren zu messen.

3.3.3 *Datenaufbereitung*

Da nun der Clusteralgorithmus feststeht, kann die weitere Verarbeitung der Daten beschrieben werden. Ein erster Schritt dazu ist die ursprünglichen Datenquellen in konsistente Text-Corpora umzuwandeln. Konsistent heißt dabei, dass jedes Dokument einen ordentlichen Inhalt, Titel und Erstellungsdatum hat, sowie dass keine identischen Dokumente unter verschiedenen Titeln vorliegen. Zudem tragen bestimmte Typen von Dokumenten besonders markante Störmerkmale, die manuell entfernt werden können. Die Datenaufbereitung läuft deshalb in drei Phasen ab – der Entfernung von Störmerkmalen, der Entfernung von Duplikaten und gegebenenfalls der Aggregation der Dokumentinhalte.

3.3.3.1 *Störmerkmale*

E-Mails und ähnliche Newsgroup-Beiträge entsprechen historisch begründet einer bestimmten Form[55]. Zitate werden z. B. durch das „>“ abgesetzt und manchmal mit einer Zeile der Form „Am ... schrieb ...:“ eingeleitet. Am Ende einer E-Mail oder eines Newsgroup-Beitrags steht häufig eine sog. Signatur. Diese wird per Konvention durch die Zeichen „- - “ vom Text abgesetzt. Insbesondere diese Signaturen müssen entfernt werden, da sie sonst Ähnlichkeit zwischen verschiedenen Dokumenten nur anhand des Autors, nicht anhand des Inhalts erzeugen. Da Zitate keine originäre Beiträge sondern nur Kopien sind, haben wir uns ebenso entschlossen diese vollständig zu entfernen, so dass eine E-Mail-Nachricht nach der Entfernung der Störmerkmale nunmehr aus dem vom Autor geschriebenen Text besteht. Einer ähnlichen Behandlung wurden die Titel der E-Mails unterzogen, die nach Möglichkeit aller Vorkommen von diskursstrukturierenden Elementen wie „Re:“, „Fwd:“, etc. entledigt werden sollten.

Die Journal-basierten Quellen (Computers & Security, ACM TISSEC und IEEE) hatten keinerlei solcher offensichtlichen und leicht erkennbaren Strukturmerkmale[56]. Deshalb wurde dieser Schritt nur für E-Mail- und Newsgroup-Beiträge durchgeführt.

3.3.3.2 *Duplikate*

Es hat sich gezeigt, dass in den Datenquellen der IEEE und der Computers & Security teilweise identische PDFs zu unterschiedlichen Artikeln vorhanden waren. Dieses Phänomen tritt vornehmlich bei sehr alten Beiträgen auf, welche scheinbar nicht korrekt digitalisiert wurden. Dies könnte zur Bildung von besonders dichten Clustern führen, die dann die Existenz von weniger dichten Clustern verdecken. Zudem würden die zeitlichen Publikationshäufigkeiten durch diese Duplikate verzerrt werden. Ein weiteres damit im Zusammenhang stehendes Problem ist, dass diese identischen PDFs der unterschiedlichen Artikel nicht ganz identisch sondern leicht modifiziert sind. Der Test der Gleichheit zweier Dokumente musste diesen minimalen Änderungen gegenüber auch noch neutral sein.

55 Nichts was nicht spezifiziert wäre und so ist in Gellens (2004, RFC3676) der empfohlene Zitationsstil und die empfohlene Signaturform für E-Mails und Newsgroup-Beiträge beschrieben.

56 Wie später noch festgestellt wird, gibt es doch welche, so z. B. Editorials. Allerdings sind diese Strukturmerkmale recht kompliziert zu erkennen und nicht ohne weiteres aus den Texten entfernbar.

Immer wenn ein Element einer Liste der Länge n mit einem jedem anderen verglichen wird, so hat man es mit einem $O(n^2)$ Aufwand zu tun. Dies bedeutet, für jedes weitere Element in der Liste müssen n-mal so viele Vergleichsoperationen durchgeführt werden, was schnell zu enormen Laufzeiten führt. Dieses Problem tritt bei der Berechnung der Distanzmatrix für den Clustering-Algorithmus ebenso auf wie bei dem nun vorausgehenden notwendigen Vergleich aller Dokumente untereinander. Die Anforderung ist, dass die Vergleichsoperation besonders einfach und effizient gestaltet werden muss. Da es bei diesem Vergleich nur um absolute Gleichheit, nicht um proportionale Ähnlichkeit ging, konnten wir auf ein Verfahren mit sog. unscharfen Hashes zurückgreifen. Für jedes Dokument wurde ein solcher unscharfer Hash[57] errechnet (eine Operation mit dem Aufwand $O(n)$). Diese kurzen Repräsentationen der Dokumente sollten folgende Eigenschaften erfüllen:

- Gleiche Dokumente führen zu gleichen Hashes.
- Ähnliche Dokumente führen zu ähnlichen Hashes.
- Längere Dokumente führen zu längeren Hashes als kürzere Dokumente.

Ein Verfahren, das Hashes mit diesen Eigenschaften erzeugt, ist folgendes:

1. Trenne das Dokument an Whitespaces (Leerzeichen, Tabulatoren, Zeilenumbrüche) in Tokens.
2. Lösche alle Sonderzeichen (nicht-Buchstaben) aus den entstandenen Tokens.
3. Zähle die Häufigkeiten der verschiedenen Tokens (lokales Dictionary).
4. Lösche alle Häufigkeiten die kleiner als zwei sind.
5. Teile die verbliebenen Häufigkeiten Modulo mit 256, so dass sie durch ein Byte darstellbar sind.
6. Konkateniere die verbliebenen Häufigkeiten anhand der alphabetischen Sortierung der zugehörigen Tokens.

Das Ergebnis dieses Algorithmus ist eine Sequenz von Bytes, die durch die üblichen String-Operationen behandelt werden kann. Ein besonders schneller Vergleich auf Gleichheit zweier Dokumente läuft nun mit Eingabe eines Grenzwertes k wie folgt ab:

1. Vergleiche die Längen der beiden Byte-Folgen. Ist die Längendifferenz größer als k, so sind die Dokumente unterschiedlich.
2. Ist die Längendifferenz kleiner als k, berechne die Edit-Distanz (Levenshtein-Distanz) für einen kleinen Teil (die ersten $3 \cdot k + 1$ Zeichen) der Byte-Sequenz[58]. Ist die Edit-Distanz in diesem Teil bereits größer als k, sind die Dokumente unterschiedlich.

57 Formal gesehen wird sich zeigen, dass wir keine echten Hashes im Sinne von Hash-Funktionen erzeugen, denn die gleiche Länge der Hashwerte ist keine Anforderung für unseren Vergleich von Dokumenten.

58 Die Edit-Distanz zu berechnen hat den Aufwand $O(n \cdot m)$, dabei seien n, m die Längen der Zeichenfolgen. Da in den meisten Fällen der Vergleiche die Dokumente tatsächlich unterschiedlich sind, ist auch bereits in einem Teil des Hashes die Edit-Distanz größer als das gewählte k. Mit diesem Zwischenschritt wird also eine Beschleunigung des Vergleichs erreicht.

3. Ist die Edit-Distanz im ersten Teil der Byte-Folge kleiner als k, berechne die Edit-Distanz für die gesamte Byte-Folge. Ist diese Edit-Distanz größer als k, so sind die Dokumente unterschiedlich, sonst sind sie gleich.

Mit dem beschriebenen Verfahren (und dem Parameter $k = 5$) wurden die Dokumente der Computers & Security und der IEEE verglichen. Die IEEE Daten mit knapp 40.000 Dokumenten ließen sich in ca. 45min auf einer herkömmlichen 1,6 GHz CPU vollständig paarweise vergleichen[59]. Dabei wurden in den Computers & Security Daten ungefähr 1000 Duplikate gefunden, bei der IEEE ca. 8000. Ein Großteil davon waren leere Dokumente, die offensichtlich keinen OCR Text-Layer enthielten. Aus den gefundenen Gruppen gleicher Dokumente wurde jeweils eines behalten und die anderen Dokumente verworfen. Da die E-Mails und Newsgroup-Beiträge bereits eine eindeutige, von Servern vergebene Nachrichten-IDs trugen, konnten (bis auf vom Absender versehentlich doppelt verschickte Nachrichten) keine Dopplungen vorkommen. Wir haben deshalb auf das Entfernen von Duplikaten bei E-Mails und Newsgroups verzichtet. Ebenso war ein Entfernen von Duplikaten durch die geringe Zahl von Dokumenten in den ACM TISSEC Daten nicht erforderlich.

3.3.3.3 *Datenaufbereitung*

Durch die oben beschriebene Entfernung der Zitate in E-Mail- und Newsgroup-Beiträgen sind viele Texte recht kurz, enthalten also nur noch wenige Worte. Jedoch sind E-Mails und Newsgroup-Beiträge gleichermaßen in Diskussionsträngen gegliedert. Beim Beantworten von Nachrichten schicken Mail-Clients üblicherweise eine Referenz auf die beantwortete Nachricht mit, so dass ein Nachrichtenbaum entsteht. Unter der Annahme, dass in einem so entstehenden Thread über ein spezifisches Thema geredet wird, haben wir uns entschlossen alle Nachrichten eines Threads zusammen mit ihren Titeln (damit die Information in den Titeln auch zum Clustern herangezogen wird) zu einem Dokument zu konkatenieren. Der resultierende Titel des Dokuments wurde dabei als der kürzeste der Titel aller Nachrichten des Threads festgelegt. Das resultierende Entstehungsdatum ist das arithmetische Mittel aller Absendedaten des Threads. Eine Aggregation von Inhalten in Journal-basierten Quellen ist nicht sinnvoll und wurde dementsprechend nicht durchgeführt.

3.3.4 *Vorverarbeitung*

Nach der Datenaufbereitung stehen konsistente Corpora zur Verfügung, die nun weiter verarbeitet werden können. Der DBSCAN-Algorithmus clustert numerische Vektoren, die Dokumente müssen also in einem Vorverarbeitungsschritt dementsprechend umgewandelt werden. Wir wählen dabei den herkömmlichen Weg, also eine wortweise Tokenization von Dokumenten, mit TFIDF-Darstellung. Die einzelnen Schritte der Vorverarbeitung sowie die gewählten Parameter werden in den nächsten Abschnitten vorgestellt.

Dabei ist darauf zu achten, die Anzahl der Dimensionen eines TFIDF-Vektors möglichst gering zu halten sind ohne Informationen zu ver-

59 Ca. 800.000 Vergleiche in 45 min entspricht 290.000 Vergleichen pro Sekunde.

lieren. Eine hohe Anzahl von Dimensionen verringert nicht nur die Laufzeiteffizienz, sondern auch die Qualität des Clusterings erheblich.

3.3.4.1 *Tokenization*

Der erste Schritt besteht darin den Text in einzelne Komponenten zu zerlegen. Der Standardweg dabei ist die wortweise Zerlegung von Text. Alternativen wie N-Gramme und Multi-Word Term Extraction wurden ausgeschlossen. Da lediglich Dokumente, die in der englischen Sprache geschrieben sind, verwendet werden und die Darstellung der Dokumente für das Lernverfahren möglichst für Menschen lesbar gehalten werden soll um die Auswertung zu vereinfachen, wurde gegen N-Gramme entschieden. Multi-Word Term Extraction ist zwar ein viel versprechender Ansatz, aber es ist sehr aufwendig umzusetzen, die Berechnungszeit für viele Dokumente wird durch die benötigte Textanalyse dramatisch erhöht.

Daher wurde für die Tokenization in einzelne Wörter entschieden, wobei die Groß-/Kleinschreibung nicht beachtet wurde. Dies lässt sich mit dem Parsen des Textes mit Hilfe einfacher regulärer Ausdrücke bewerkstelligen. Bei der Tokenization wurden Wörter, die mit Bindestrich getrennt werden, auseinander gezogen, d.h. Wörter wie „e-mail“ oder „anti-virus“ werden zu „e“ und „mail“ bzw. „anti“ und „virus“. Auf diese Weise lässt sich das gesamte verwendete Vokabular reduzieren ohne viele Informationen zu verlieren.

Viele Wörter die mit Bindestrich geschrieben werden, tauchen nur sehr selten in den Dokumenten auf und können leicht durch spätere Dimensionsreduktionsverfahren ganz verloren gehen, obwohl sie sinnvolle Informationen beinhalten. So finden sich z.B. im „Computers & Security“-Corpus Wörter wie „security-class“, „protocol-based“, „spam-advertisement“ und „computer-product“. Trennt man diese Wörter beim Bindestrich, gehen keine wesentlichen Informationen verloren, aber es werden Dimensionen reduziert. Im „Computers & Security“-Corpus befinden sich insgesamt 90.011 Wörter[60], davon beinhalten 13.922 einen Bindestrich, also rund 15%, trennt man diese Wörter, so gibt es nur noch eine Gesamtanzahl von 75.248 Wörtern und damit eine Reduzierung von ca. 16%.

Desweiteren werden Wörter bei allen weiteren Sonderzeichen getrennt, also auch bei Hochkommas wie im Falle von „Peter's“ oder „aren't“. Zahlen werden komplett ignoriert, da sie in unserem Fall keinen semantischen Wert haben. Zwei Dokumente, in denen die Zahl 2000 häufig auftritt, müssen thematisch nicht zwangsläufig zusammengehören. Viel wichtiger hingegen ist die Einheit auf die sich die Zahl bezieht; sind also 2000 Jahre gemeint oder 2000 Codezeilen? Die Einheit, auf die sich eine Zahl bezieht, kann aber mit dem oben beschriebenen Verfahren als Wort extrahiert werden.

3.3.4.2 *Rechtschreibkorrektur*

Nach der Tokenization an Wortgrenzen wäre die Möglichkeit gegeben die Dokumente einer Rechtschreibkorrektur zu unterziehen. Wir haben in Abschnitt 3.2.1.4 zwei alternative Ansätze zur Korrektur von Rechtschreib- und insbesondere OCR-Fehlern beschrieben. Der komplizierte Ansatz von Tong u. Evans (1996) kam wegen fehlender Trainings-

60 Diese Zahl bezieht sich auf bereits gestemmte Wörter, also auf die Größe der endgültigen Feature Vektoren.

daten und Wörterbücher sowie dem hohen Implementierungsaufwand nicht in Frage. Das von uns vorgeschlagene Verfahren OCR-MAPPING wurde implementiert. Wir konnten für alle Corpora eine Korrekturtabelle erzeugen und unser Gesamtverfahren mit den so erzeugten korrigierten Dokumenten testen. Es hat sich leider gezeigt, dass der schon oben beschriebene Verwischungseffekt deutlich zu Tage tritt und sich negativ auf die Qualität der Cluster auswirkt. Cluster sind schneller verschmolzen, wobei gleichzeitig mehr Punkte als Rauschen identifiziert wurden. Wir haben deshalb auf die Rechtschreibkorrektur verzichtet und lediglich Worte mit bestimmten niedrigen Häufigkeiten entfernt. Diese Stufe findet jedoch erst nach dem Stemming statt und wird weiter unten in Abschnitt 3.3.4.6 genauer beschrieben.

3.3.4.3 *Entfernen der Stop-Words*

Zur Entfernung der Stop-Words wird die Stop-Words-Liste der NLTK-Bibliothek mit 571 englischen Wörtern verwendet, in dieser Stop-Word-Liste sind auch alle Buchstaben des englischen Alphabets enthalten, so dass „Wörter“, die nur aus einem Zeichen bestehen, wegfallen. Das betrifft vor allem Begriffe wie „e-mail“, die in „e“ und „mail“ zerlegt werden. Nach Anwendung der Stop-Word-Liste bleibt dabei also nur noch „mail“ übrig. Doppeldeutige Begriffe wie z.B. „can“ und „will“, die nicht nur als störende Hilfsverben, sondern auch als bedeutungsvolle Nomen aufgefasst werden können, sind dabei in der Stop-Word-Liste enthalten.

3.3.4.4 *Stemming*

Nach der Tokenization, werden die verschiedenen Wörter eines Dokuments gestemmt. Durch das NLTK-Framework besteht schon die Möglichkeit den Porter-Stemmer zu verwenden ohne ihn neu implementieren zu müssen, sowie eine Lemmatisation durchzuführen, also die Bestimmung von Wortarten mittels Part-of-Speech-Tagging, um danach mit dem NLTK Wordnet-Stemmer eine morphologische Substitution vorzunehmen.

Das testweise Anwenden von Lemmatisation auf den „Computers & Security“-Corpus hat einen starken Anstieg der Verarbeitungszeit ergeben. Desweiteren assoziierte der Part-of-Speach-Tagger von NLTK die Wörter mit dem Penntreebank-Tagset, während der mitgelieferte Wordnet-Stemmer lediglich zwischen Nomen, Verben, Adjektiven und Adverbien unterschied, weshalb die Tags im Tagset darauf gemappt werden mussten. Da es Wörter gibt, deren Wortarten nicht korrekt erkannt werden, kann es vorkommen, dass einmal ein Wort erfolgreich auf seinen Wortstamm zurückgeführt werden kann und in einem anderen Satz nicht richtig erkannt wird und daher nicht geändert wird. Dies tritt vor allem bei OCR- und Tippfehlern auf, so dass eine Lemmatisation für unsere Datensätze ungeeignet sind. Aus diesem Grund haben wir uns für den Porter-Stemmer entschieden. Der Porter-Stemmer arbeitet auch zuverlässig auf falsch geschrieben Wörtern, solange die Wortendung nicht betroffen ist und performt sehr gut bei großen Wortmengen.

3.3.4.5 *Erstellen der Feature-Matrix*

Der von uns ausgewählte Cluster-Algorithmus benötigt numerische Vektoren als Eingabedaten. Um die Dokumente geeignet darstellen zu können, haben wir uns für eine TFIDF-Gewichtung entschieden. Die Berechnung des TFIDF-Wertes eines Wortes erfolgt in konstanter Zeit, sofern beim Zusammenstellen des globalen Dictionaries auch mitgezählt wird wie häufig ein Wort in verschiedenen Dokumenten auftritt. Besitzt man diese Information, stellt die Erstellung der Feature-Matrix auch für große Datenmenge kein Problem da. Die Berechnungszeit spielt also bei der Beurteilung keine Rolle. Der TFIDF-Gewichtung schlägt nun vor allen die anderen Verfahren durch eine geschickte, statistische Bewertung der einzelnen Wörter. So werden Begriffe, die in sehr vielen Dokumenten auftauchen, einen sehr niedrigen Wert erhalten und daher an Einfluss verlieren; ein Verhalten das besonders bei den Begriffen „security“ und „computer“ zu erwarten ist. Und auf der anderen Seite werden Wörter, die lediglich in einem Dokument vorkommen, für dieses als sehr ausschlaggebendes Merkmal gewählt, ohne dass längere Textdokumente mit höheren Worthäufigkeiten den Corpus dominieren können. Damit stellt die Gewichtung mittels des TFIDF-Verfahrens die günstigste Wahl da.

3.3.4.6 *Weitere Dimensionsreduktion*

Nach dem Stemming hat z.B. der „Computers & Security“-Corpus nach dem oben beschriebenen Verfahren ca. 75.000 Wörter. Darunter sind leider viele OCR-Fehler und, im Falle der internetbasierten Quellen, Tippfehler. Da nur sehr selten die gleiche Tipp- oder OCR-Fehler mehrmals auftreten, wurden zusätzlich alle Wörter entfernt die nur einmal in einem Dokument auftreten. Diese Maßnahme wird auch Word-Threshold genannt, stellt also einen Schwellenwert für die Worthäufigkeit da bevor ein Wort als gültiges Feature eines Dokuments anerkannt wird. Selbst wenn so ein aussortiertes Wort kein Fehler ist, würde sein TFIDF-Wert nahezu Null sein und kann daher vernachlässigt werden ohne viel Informationen zu verlieren. Dies reduziert die Anzahl der Wörter und damit die Dimensionen dramatisch. Für Computers & Security brachte dieses Verfahren eine Reduktion von 75.248 auf nur 20.267 Wörter, also eine Reduktion um 73% ein.

Darunter sind aber noch immer einige OCR-Fehler zu finden. Denn häufig werden zwei oder mehr Wörter nicht richtig als separate Wörter durch die Schrifterkennung erkannt und damit in einem Wort zusammengefasst. So finden sich im „Computers & Security“-Corpus Wörter wie z.B. „whichisincompetitionwithorgan“ oder „othnationalinformationsystemsecurityconfer“. Zusätzlich finden sich aber auch sehr kurze, zweistellige Begriffe, die kaum semantische Information tragen und die gleichklingende Abkürzung unterschiedlicher Begriffe darstellen können.

Um diese Wörter auszuschließen wurden alle Begriffe entfernt deren Länge, also Anzahl von Zeichen, größer als 23 oder kleiner als 3 ist. Diese Maßnahme reduzierte den „Computers & Security“-Corpus von 20.267 Wörter auf nur noch 19.769.

Die hier beschriebenen Maßnahmen zur Dimensionsreduktion wurden auf alle Quellen angewendet. Die einzige Ausnahme ist der IEEE-Corpus, welcher beim ersten Verarbeitungsschritt eine Dimensionsgröße von 248.479 besaß, danach wurde die minimale Länge von Wörter

auf 4 angehoben und ebenfalls alle Wörter entfernt, die weniger als zweimal in einem einzelnen Dokument vorkamen. Damit reduzierte sich die Dimensionsgröße auf 122.009, also um knapp die Hälfte. Da dieser Wert für ein gutes Clustering noch immer zu hoch ist, wurde in einem weiteren Verarbeitungsschritt alle Wörter entfernt, welche nicht mindestens zehn mal im gesamten Corpus vorkommen. Diese Maßnahme führt zu einer Dimensionsgröße von nur noch 45.439 Wörtern, damit ergibt sich eine Reduktion von insgesamt 63%.

Von der weiteren Datentransformation und Dimensionsreduktion mittels LSA oder PCA wurde abgesehen. Sowohl LSA als auch PCA haben (wie in Abschnitt 3.2 beschrieben) so hohe Rechenzeit- und Speicheranforderungen, dass die Verfahren auf großen Corpora wie den Mailinglisten, der Newsgroup Comp.Sec.Misc und dem IEEE-Corpus nicht anwendbar sind. Die verbleibenden Corpora sind wiederum schon recht gut mit den bestehenden Mitteln zu verarbeiten. Ebenso ist unklar, welche präzisen Auswirkungen die LSA oder PCA auf das restliche Cluster-Verfahren hätten. Das Verwenden einer Dimensionsreduktion mittels LSA hätte zu dem den Nachteil, dass nur schwer nachvollzogen werden kann, welche Wörter zu den Konzepten gehören, die für die Definition eines Clusters ausschlaggebend sind.

3.3.5 *Auswahl der Clusterparameter*

DBSCAN erwartet zwei Parameter zum Clustern der Daten, zum einen *MinPts* die Anzahl der Punkte in der direkten Nachbarschaft eines Vektors, die ihn zu einem Core-Point, also einem zentralen Cluster-Mitglied machen. Desweiteren muss *Eps* der Radius festgelegt werden, der die Größe der Nachbarschaft definiert.

Ester u. a. (1996) schlagen vor *MinPts* auf 4 zu setzen, da eine Vergrößerung des Wertes in Experimenten nur wenig Vorteile gebracht hatte, aber die Rechenleistung erhöhte und den Radius mittels eines K-Dist-Graphen zu ermitteln (siehe Abschnitt 3.2.3.2). Dieses Vorgehen empfiehlt sich zumindest für die meisten zweidimensionalen Probleme im euklidischen Raum. In unserem Fall aber haben wir es mit hochdimensionalen Daten zu tun, deren Ähnlichkeit über die Cosinusdistanz ausgedrückt wird. Das hier vorgeschlagene Verfahren ist also nur bedingt anwendbar.

Für den TISSEC- und „Computers & Security“-Corpus wurden die K-Dist-Graphen für *MinPts* von 4, 5 und 6 berechnet. Die Graphen sind in der Abbildung 16 für den TISSEC-Corpus und in der Abbildung 17 für den „Computers & Security“-Corpus zu finden. Wie dort zu sehen, fallen alle Graphen am Anfang wie erwartet stark ab, dieser Abfall endet aber in allen Fällen bereits im Bereich von ungefähr $0,85$ für Computers & Security und für TISSEC sogar bei $0,95$, wobei diese Werte für zunehmende Anzahl von *MinPts* weiter nach oben steigen. Das hieße nach der klassischen Vorstellung zur Selektion der DBSCAN-Parameter, dass man mit einem Radius *Eps* von $0,85$ das günstigste Gleichgewicht zwischen Noise- und Core-Points erreicht, also ab dieser Distanz fallen viele Punkte zu Clustern zusammen. Das Durchführen des Cluster-Vorgangs mit einem so hohen Radius führt aber bei den hier verwendeten Datenquellen lediglich zu einem einzelnen großen Cluster, welcher den Großteil aller Punkte enthält, sowie wenige, sehr

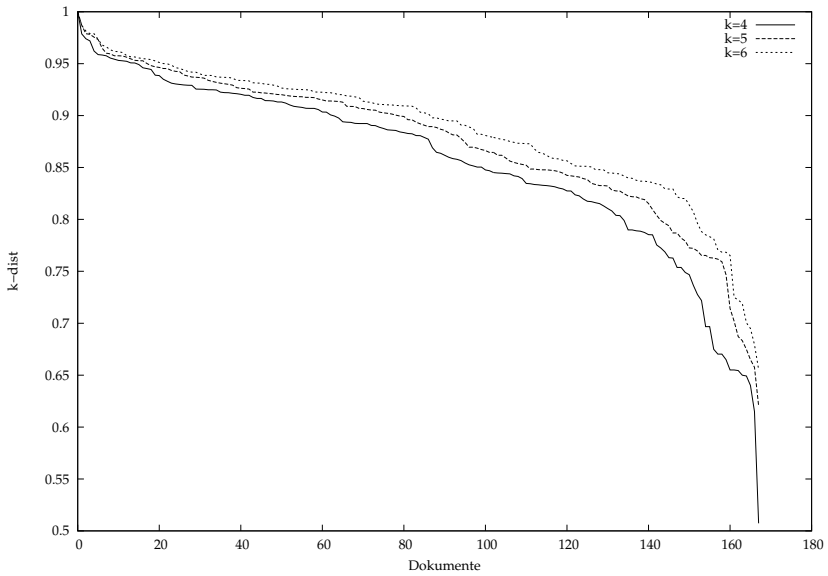


Abbildung 16: K-Dist-Graphen für TISSEC

kleine Cluster, die Ausreißer zusammenfassen. Die Details der Cluster-Ergebnisse werden im Kapitel 4.1 vorgestellt und diskutiert, dort wird auch näher auf die Eigenschaften des Feature-Raums eingegangen, welche sich aus den K-Dist-Graphen ablesen lassen.

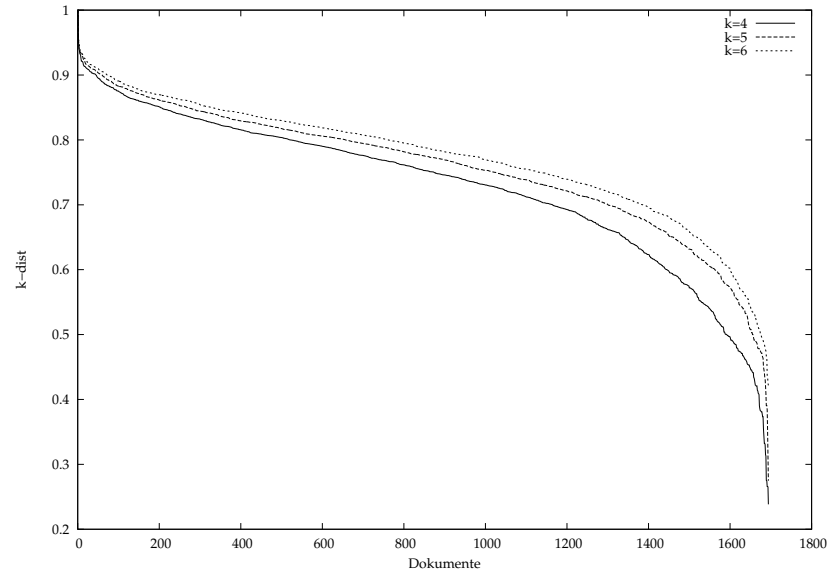


Abbildung 17: K-Dist-Graphen für Computers & Security

Aus den Ergebnissen der K-Dist-Graphen lassen sich also nur schwer günstige Parameter ableiten. Um die Wahl der Parameter zu vereinfachen, wurden daher für jede Datenbank die Cluster für die *MinPts* von 2 bis 6 berechnet mit jeweils einem *Eps* von $0,2$ bis $0,8$ in $0,1$er Schritten, sowie mit dem *Eps* von $0,85$. Da diese Unterteilungen teilweise nicht genau genug waren, wurde für alle Datenbanken noch einmal mit kleineren Schritten von $0,01$ in ausgewählten Bereichen auf der *Eps*-Skala

geclustert. So wurden beispielsweise für comp.sec.misc zusätzlich auch die Cluster-Ergebnisse für einen *Eps* von 0,3 bis 0,4 untersucht.

Die nun erzeugten Cluster wurden nach den Noise-Points über die Standardabweichung der Cluster-Längen angeordnet. In Abbildung 18 ist der entsprechende Parametergraph für Computers & Security abgebildet. Die Graphen für die anderen Datenquellen befinden sich im Anhang dieser Arbeit. Auf der X-Achse ist die Standardabweichung logarithmisch aufgetragen, die Y-Achse stellt den prozentualen Anteil von Noise-Points da, die aus einem spezifischen Clustering resultieren. Jeder Punkt stellt ein konkretes Clustering da und wird durch ein 3-Tupel bezeichnet, welches sich aus den Parametern *Eps* und *MinPts*, sowie der Anzahl resultierender Cluster zusammensetzt.

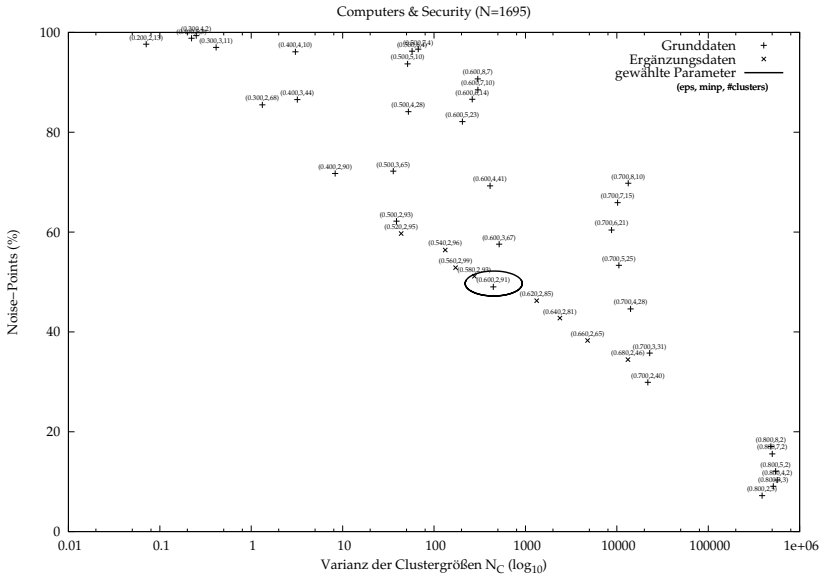


Abbildung 18: Parametergraph für den „Computers & Security"-Corpus

Bei allen Graphen lässt sich ein ähnliches Verhältnis zwischen dem Noise-Anteil und der Cluster-Größen beobachten. Mit steigendem Radius werden immer mehr Punkte zu einem zentralen Cluster zugeordnet, bis mit einem Radius von 1,0 alle Punkte in einem Cluster zusammengefasst werden, da dies der größtmögliche Radius ist und so jeder Punkt alle anderen Punkte in seiner Nachbarschaft hat. Bei Werten leicht unterhalb von 1,0 zerfällt dieser großer Cluster scheinbar an seinen Rändern langsam in kleinere Cluster, besitzt aber noch immer ein großes Zentrum.

Da ein Cluster, welcher einen Großteil aller Dokumente beinhaltet, keinerlei Aussagekraft für die hier betrachtete Fragestellung besitzt, müssen also Parameter gefunden werden, bei dem noch kein zentraler, großer Cluster existiert, was sich in der Varianz der Cluster-Größen bemerkbar macht. Als Richtlinie gilt also Parameter zu finden, die in einem Clustering resultieren, welches sowohl eine geringe Anzahl von Noise-Points hat, als auch nur eine geringe Standardabweichung der Cluster-Länge aufweist.

Für die einzelnen Datenquellen sind die ausgewählten Parameter, welche verwendet wurden um das endgültige Clustering für die Analyse zu erstellen, in der Tabelle 7 aufgelistet. Für den „Computers &

Corpus	Eps	MinPts
ACM TISSEC	0,815	2
Computers & Security	0,60	2
IEEE	0,32	2
InfosecNews	0,51	2
Security Basics	0,32	3
comp.sec.misc	0,32	3

Tabelle 7: Wahl der DBSCAN-Parameter für die verschiedenen Quellen

Security"-Corpus z.B. haben wir uns also für *MinPts* von 0.60 und *Eps* von 2 entschieden. Zusätzlich befinden sich in jedem Parametergraph Kreise, welche genau denjenigen Punkt einrahmen, welcher das ausgewählte Clustering repräsentiert.

3.3.6 *Auswertungsphase*

Um die gewonnenen Cluster auswerten zu können, benötigt man zwei Informationen: Zum einen welches Thema sie umfassen, und zum zweiten wie sich die Publikationshäufigkeiten über die Zeit verteilen.

In Abschnitt 3.2.4.1 haben wir verschiedene Zusammenfassungs-Systeme betrachtet und ein für unser auf TFIDF-Werten basierendes Verfahren optimiertes System genauer vorgestellt. Dieses ebenso auf den TFIDF-Werten der Dokumente aufbauendes System haben wir dann auch eingesetzt. Es präsentiert dem Betrachter eine kleine Menge charakteristischer Stichworte und Titel repräsentativer Dokumente, sowie bedeutende Sätze aus den Texten der Dokumente. Anhand dieser drei verschiedenen Informationen sollte sich eine grobe Vorstellung von den Inhalten des Clusters erreichen lassen. Der Hauptgrund auf dieses eigene System und nicht auf andere, bereits untersuchten oder vorgeschlagenen Systeme zurückzugreifen, stellt der zeitliche Aufwand einer Implementierung oder Einbindung in unsere Experimentierumgebung dar. Daneben ist auch die hohe Geschwindigkeit unserer Methode von enormen Vorteil. Andere Systeme stellten zudem noch jeweils individuell andere Anforderungen oder Annahmen an die Vorverarbeitung oder zusätzliche Eingaben in den Algorithmus, so dass letztlich nur die maßgeschneiderte Lösung realisierbar war.

Neben den Zusammenfassungen fehlt nun noch die Darstellung des zeitlichen Verlaufs der Publikationshäufigkeit. Wir haben uns entschieden eine über das gesamte Publikationsvolumen normalisierte Darstellung zu wählen und gleichzeitig diese Normierungsbasis zur Referenz mit darzustellen. Diese ist in diesem Fall das gesamte Publikationsvolumen, also die Summe aus Noise-Points und allen geclusterten Dokumenten. Die Entscheidung auch die Noise-Points mit einzubeziehen ist durch den hohen Anteil dieser begründet[61]. Es ist unwahrscheinlich, dass die Noise-Points tatsächlich unwichtig sind. Die Profile der Cluster werden als Messpunkte und mit einer durch Glättung erzeugten Kurve visualisiert. Die Aufsummierung hätte der detaillierten Analyse der einzelnen Cluster entgegen gestanden. Ebenso haben wir uns entschieden nur die größten acht Cluster abzubilden,

61 Dazu mehr in Abschnitt 4.2.

da die folgenden Cluster meist so klein sind, dass sie keine wirkliche Aussagekraft mehr besitzen.

Da nun mit der Auswertung auch der letzte Teil der Methode geklärt ist, können wir alles noch einmal zusammenfassen und optisch kompakt darstellen.

Implementierungsbeschreibung

Für das gesamte Projekt wurde auf die Programmiersprache Python zurückgegriffen, wobei für einige Teile aus Performancegründen mittels Cython Anbindungen an Module geschaffen wurden, die in C implementiert sind. Zwei Gründe haben die Wahl auf Python fallen lassen. Zum einen erlaubt das Sprachdesign mit nur wenigen Zeilen einen gut lesbaren, funktionierenden Code zu schreiben und desweiteren wird dank umfangreicher REPLs[a] wie IPython der Entwicklungszeitraum stark verkürzt. Das macht Python zu einer idealen Sprache für Rapid-Prototyping und erleichtert das Implementieren und Testen neuer Funktionen. Desweiteren kommt Python mit einer Vielzahl nützlicher Programmbibliotheken, die für die Entwicklung von Data-Mining-Anwendungen hilfreich sein können. Dazu zählt vor allem eine sehr umfangreiche Sammlung mathematischer Funktionen namens *SciPy* die in C implementiert ist und somit auch ausreichend performant ist. Auf der anderen Seite gibt es NLTK, das „Natural Language Toolkit“, eine Bibliothek bestehend aus Algorithmen, Werkzeugen und Text-Corpora für Sprachverarbeitungssysteme, die von der University of Pennsylvania unter einer freien Lizenz zur Verfügung gestellt wurden und seitdem aktiv weiter entwickelt werden[b].

Um maschinelle Lernverfahren auf allen Textquellen anwenden zu können, werden die Daten zuerst in ein einheitliches Format gebracht. Da für jedes Lernverfahren Texte in Form eines Feature-Vektors dargestellt werden, macht es Sinn die Beiträge und Veröffentlichungen der einzelnen Quellen erst vorzuverarbeiten und dann die Feature-Matrizen zu speichern. Aus diesem Grund wurden alle Textquellen in Datenbanken organisiert. Dabei werden alle Dokumente mit ihrem Veröffentlichungsdatum in einer Tabelle gespeichert, sowie eine Liste aller Wörter im gesamten Corpus und die Werte der Feature-Matrix. Nach dem Clustern der Daten folgen noch zwei weitere Tabellen um Textdokumente mit konkreten Clustern assoziieren zu können. Als Datenbanksoftware wurde SQLITE verwendet um das Datenbankmanagement so einfach wie möglich zu halten[c].

Um eine Textquelle also nun durch eine Feature-Matrix darstellen zu können und in eine Datenbank zu schreiben, wird das Python-Modul *Processor* bereit gestellt, welches erlaubt sämtliche Vorverarbeitungsschritte mit nur einem Methodenaufruf durchzuführen und ggf. auch anschließend ermöglicht das Ergebnis zu clustern.

a REPL steht für „read evaluate print loop“, eine Konsolenanwendung zur interaktiven Verwendung von Programmiersprachen, welche zum schnellen Testen und Ausprobieren von Code geeignet ist.

b Die Homepage von NLTK ist zu finden unter `http://www.ntlk.org`.

c SQLITE speichert jede Datenbank als einzelne Datei ab, weitere Informationen finden sich unter `http://www.sqlite.org`.

Dieses Modul hält allerdings viele Informationen im Arbeitsspeicher um einen schnellen Ablauf garantieren zu können, was bei größeren Datenbeständen Schwierigkeiten bereiten kann. Deshalb wurde für IEEE ein eigenes Skript geschrieben (*processIEEE.py*), das Zwischenergebnisse, die während der Vorverarbeitung anfallen, persistent auf der Festplatte oder der Datenbank speichert. Aus diesem Grund sind in der IEEE-Datenbank weitere Tabellen zu finden, u.a. um die Worthäufigkeiten einzelner Dokumente festzuhalten (lokales Dictionary).
Während der Vorverarbeitung wird an einigen Stellen von Klassen aus der NLTK-Bibliothek zurückgegriffen, so wird bei der Tokenization der *RegexpTokenizer* aus dem Modul *nltk.tokenize.regexp* verwendet. Um Texte also in Features aufzutrennen, wird dieser erst komplett in Kleinbuchstaben umgewandelt und dann mit dem regulären Ausdruck $[a-z]+$ aufgetrennt, so dass Zahlen und Sonderzeichen verloren gehen. Desweiteren wird die Porter-Stemmer-Implementierung von NLTK verwendet (Klasse *PorterStemmer* in Modul *nltk.stem.porter*), welche den Vorteil hat, dass bereits erfolgreich gestemmte Wörter zwischengespeichert werden. Wenn das gleiche Wort also nochmal gestemmt werden soll, kann auf den Speicher zurückgegriffen werden kann anstatt den Algorithmus ein weiteres mal anzuwenden. Dies bringt eine weitere Zeitersparnis bei großen Corpora.
Die Feature-Matrizen für alle Textquellen sind sehr groß und umfassen mehrere tausende Dimensionen. Dennoch sind die meisten Werte in der Feature-Matrix Nullen und nur wenige Felder sind tatsächlich mit größeren Werten besetzt. Um diese dünnbesetzten Matrizen möglichst speichersparend verwenden zu können, wurde auf die Implementierung der „Sparse"-Matrizen von *scipy* zurückgegriffen. Diese Datenstrukturen speichern lediglich die Werte ungleich Null mit ihrer Position in der Matrix und lassen sich wie gewöhnliche Matrizen verwenden, da sie alle wichtigen mathematischen Operationen dafür unterstützen[a].
Für den Cluster-Algorithmus DBSCAN wurde eine eigene Implementierung verwendet, da kein Code für den Algorithmus zur Verfügung stand und der Algorithmus so besser auf die gegebenen Datenformate abgestimmt werden kann. Um die Performance der Berechnung der Clusterings zu verbessern, werden sog. Distanzmatrizen dargestellt, also Matrizen in denen der Abstand zwischen zwei Punkten im Feature-Raum festgehalten wird. Während des Cluster-Vorgangs werden alle Punkte mitunter mehrmals miteinander verglichen, anstatt die Entfernung der Punkte also mehrmals berechnen zu müssen, lohnt es sich die Entfernung einmal zu speichern, so dass sie in konstanter Zeit wieder abgerufen werden kann.

a Im Modul *scipy.sparse* befinden sich mehrere „Sparse"-Matrizen, welche verschiedene Strategien zum Speichern der Werte verwenden, was sich in unterschiedlichen Zeitkomplexitäten für Zugriff und Manipulation zeigt.

Da der Abstand zweier Punkte ungerichtet ist, ist die Distanzmatrix an ihrer Diagonalen gespiegelt und lässt sich so speichereffizient darstellen. Dies gilt um so mehr, da für die Berechnung eines Clusterings lediglich Abstände niedriger als der angegebene Parameter *Eps* betrachtet werden und somit auch nur diese gespeichert werden müssen. Eine solche Distanzmatrix wird daher durch eine Datenstruktur repräsentiert, die auch *sparse distance matrix*, also dünnbesetzte Distanzmatrix, genannt wird. Das Python-Modul *DBScan* bietet Möglichkeiten an solche Distanzmatrizen aus Feature-Matrizen zu berechnen und mit dieser dann das Clustering durchzuführen.

4 ERGEBNISSE

Nachdem die Datenquellen vorliegen und die Methode gewählt wurde, konnte diese angewendet werden. Mit viel Geduld und mehreren Tagen Rechenzeit, wurden auch die größten Corpora auf Standard-Desktop-PCs verarbeitet und ihre Ergebnisse können auswertet werden. Diese Auswertung soll in drei Schritten erfolgen. Als erstes, werden wir detailliert betrachten, welche Inhalten in den größten Cluster gelangt sind und wie die zeitlichen Verläufe dieser Cluster sind. Nach dieser Betrachtung kann dann die Methode bewertet werden. Ist sie erfolgreich gewesen und wie weit reicht ihre Aussagekraft? Nachdem auch die Aussagekraft eingeschätzt werden kann, können dann Schlüsse aus den Verlaufen der Daten hergeleitet werden. Den Anfang bildet jedoch die detaillierte Betrachtung der Ergebnisse, sowohl in Form der Publikationsgraphen als auch der automatisch generierten Zusammenfassungen.

4.1 BESCHREIBUNG DER ERGEBNISSE

Dieser Abschnitt gibt eine Übersicht über die Eigenschaften der Datenbestände die durch die Vorverarbeitung erzeugt und anschließend geclustert wurden. In Tabelle 8 sind dazu als erstes die Anzahl der Wörter und Dokumente in jedem Text-Corpus aufgelistet, denn insbesondere die Anzahl der verarbeiteten Dokumente in der ersten Spalte ist eine Kennzahl für die Bedeutung des jeweiligen Corpus. In einigen Fällen wie dem IEEE-Corpus oder dem „Computers & Security“-Corpus sind deutlich weniger Dokumente verblieben als vor der Datenaufbereitung vorhanden waren, denn es wurden Dokumente aussortiert, wenn sie doppelt vorkamen oder nach der Vorverarbeitung keine Wörter mehr enthielten[1].

Die Anzahl der Dimensionen in der zweiten Spalte beschreibt die Anzahl der Wörter und damit die Größe des Feature-Raums der jeweiligen Datenquelle, welche nach allen Schritten der Vorverarbeitung verblieben sind. Wie im Falle von ACM TISSEC zu sehen ist, fallen schon bei sehr wenigen Dokumenten große Mengen von Wörter an. Die Anzahl der Dimensionen wächst allerdings bei steigender Dokumentenanzahl immer weniger stark an. Dies liegt daran, dass die Wahrscheinlichkeit ein neues Wort zu finden mit jedem weiteren Dokument sinkt.

Der IEEE-Corpus fügt sich zwar gut ins Bild, allerdings wurden hier auf Grund vieler OCR-Fehler auch eine etwas modifizierte Vorverarbeitung angesetzt. Es wurden zusätzlich alle Wörter entfernt, die nicht mindestens zehn mal im Gesamt-Corpus vorkommen. Generell weisen allerdings alle Cluster sehr viele Dimensionen auf. Für Text-Mining-Applikationen werden normalerweise 10.000 bis 20.000 Dimensionen erwartet. Je mehr Dimensionen zur Kategorisierung vorliegen, desto eher versagen die den Verfahren unterliegenden statistischen Annahmen. Empirische Experimente zeigen, dass Lernverfahren besser arbeiten, wenn sie mit einer geringeren Anzahl von Features

1 Siehe dazu auch Abschnitt 3.3.3.

Corpus	verarbeitete Dokumente	Dimensionen
ACM TISSEC	168	12350
Computers & Security	1695	19769
IEEE	37274	45439
Infosec-News	13090	24932
Security-Basics	15536	14244
comp.sec.misc	24228	37696

Tabelle 8: Anzahl von Dokumente und Dimensionen in den Corpora

konfrontiert werden[2]. Jedoch erreicht schont allein der „Computers & Security"-Corpus knapp 20.000 Dimensionen, während Infosec-News und comp.sec.misc und insbesondere IEEE diesen Wert weit überschreiten. Die Security-Basics-Mailingliste kommt allerdings auf erstaunlich wenig Dimensionen, ohne dass sie einer modifizierten Vorverarbeitung unterlaufen wäre. Im Abschnitt 4.2 werden wir uns diesen beobachteten Phänomenen genauer zuwenden.

Im Folgenden werden nun zunächst die erhaltenen Cluster beschrieben und analysiert, teilweise auch mit der historischen Entwicklung in Verbindung gesetzt. Für jeden Corpus und jeden Cluster im Corpus ist im Anhang A ein Publikationsgraph zu finden. In diesem Graph lässt sich jeweils das anteilige Publikationsvolumen des Clusters, das absolute Gesamtpublikationsvolumen des Corpus pro Monat und die Anzahl der Dokumente im Cluster (N_C) ablesen. Der Graph mit dem Gesamtpublikationsvolumen des betrachteten Zeitraums ist unterhalb der Clustergraphen abgebildet. Hier wird durch eine gestrichelte Linie auch der Anteil der nicht geclusterten Dokumente, also der Noise-Points in den jeweiligen Zeiträumen dargestellt. Außerdem wurde eine Wort-, Titel- und Satz-Zusammenfassung für die Cluster jeder Datenquelle vorgenommen. Aufgrund der teilweise sehr hohen Anzahl der Cluster wurden die Zusammenfassungen dabei nur für die jeweils acht größten Cluster zusammengestellt und betrachtet. Der IEEE-Corpus stellt eine Ausnahme dar. Hier wurden auf Grund der hohen Anzahl von Clustern und der Größe des Corpus die größten 16 Cluster betrachtet.

Jedem Cluster ist eine eindeutige Identifikationsnummer (ID) zugeordnet, die vom Cluster-Algorithmus festgelegt wurde. Anhand dieser Cluster-ID lassen sich diese eindeutig im Anhang wiedererkennen. Im Verlauf der Analyse werden Wörter aus der Cluster-Zusammenfassung zitiert oder Titel der Publikationen, bzw. Mails- und Newsgroupthreads genannt, die einem Cluster angehören. Die Wörter werden dabei in ihrer durch den Porter-Stemmer auf ihren Stamm zurückgeführten Form präsentiert. Alle Titel der Publikationen oder Beiträge in den internet-basierten Quellen werden wörtlich wiedergegeben und können daher Rechtschreib- oder OCR-Fehler beinhalten.

4.1.1 *ACM TISSEC*

Mit nur lediglich 168 Dokumente ist der ACM TISSEC Text-Corpus der kleinste von den hier untersuchten Datenquellen. Die Dokumente wurden mit *Eps* 0.815 und *MinPts* von 2 geclustert und erstrecken sich

2 Vgl. dazu Weiss u. a. (1999, S.3) und Verleysen u. François (2005).

über einen Zeitraum von knapp zehn Jahren beginnend 1999 bis 2009. Das Resultat sind neun Cluster welche insgesamt mehr als 60% aller Dokumente beinhalten. Der TISSEC-Corpus besitzt sehr deutliche Cluster-Zusammenfassungen, die Thematik der einzelnen Cluster lässt sich leicht bestimmen und die Wort-, Titel- und Satz-Zusammenfassungen stimmen in den meisten Fällen miteinander überein. Es folgt eine Beschreibung der vier größten Cluster, die weiteren Cluster besitzen lediglich sieben oder weniger Dokumente, so dass die Aussagekraft ihrer Eigenschaften zu gering ist um Aussagen ableiten zu können.

4.1.1.1 *Cluster 0 - Access Control*

Der größte Cluster mit der ID 0 beinhaltet 40 Dokumente und erstreckt sich über den gesamten betrachteten Zeitraum, wobei kein bemerkenswerter Zuwachs oder Abnahme zu beobachten ist. Der prozentuale Anteil der Dokumente dieses Clusters im Vergleich zum gesamten Publikationsvolumen bleibt daher konstant bei ca. 10% mit gelegentlichen Ausreißern nach oben. Besonders markante Wörter dieses Clusters sind u.a. *role*, *rbac* (Role Based Access Control[3]), *password*, *hierarchi* (hierarchy, hierarchical), *contraint*, *admin*, sowie *polici* (policy). Anhand der Wort-Zusammenfassung lässt sich bereits ein thematischer Bezug auf „Access Control" herauslesen. Die Zusammenfassung der Titel im Cluster bestätigen dies, wie die drei zuerst erwähnten Titel zeigen: *Administrative Scope: A Foundation for Role-Based Administrative Models*, *Formal Model and Policy Specification of Usage Control* und *The ARBAC97 Model for Role-Based Administration of Roles*.

4.1.1.2 *Cluster 1 - Sensor & Wireless Security*

Der zweite Cluster umfasst 22 Dokumente und kann einen Anstieg von Publikationen ab dem Jahr 2003 verzeichnen, dieser Anstieg knickt im Sommer 2006 ein und setzt sich in den darauf folgenden Jahren wieder fort. Die Wörter, welche den Cluster gut beschreiben, sind u.a. *sensor*, *node*, *piv* (Program Integrity Verification[4]), *cell*, *wireless*, *multicast* und *packet*. Dies sind alles Begriffe, die sich im Bereich von Wireless-Netzwerken, sowie in besonderen Sensor-Netzwerken einordnen lassen. In der Auflistung der Titel der Veröffentlichungen dominiert sehr stark der Begriff *Sensor Network*, dem häufig das Attribut *wireless* oder *distributed* zugewiesen wird: *Distributed Authentication of Program Integrity Verification in Wireless Sensor Networks*, *A Framework for Indentifying Compromised Nodes in Wireless Sensor Networks*, *Establishing Pairwise Keys in Distributed Sensor Networks*. Der Anstieg des Publikationvolumens über dieses Thema um das Jahr 2003 herum ließe sich mit der zunehmenden Verbreitung der Wireless-Technologie erklären[5].

4.1.1.3 *Cluster 2 - Cryptographic Protocols*

Der drittgrößte Cluster umfasst nur noch neun verschiedene Dokumente, was die Aussagekraft über die Cluster-Eigenschaften stark ein-

3 Ein Verfahren zur Koordinierung von Zugriffsrechten auf Ressourcen von Mehrbenutzer-Betriebssystemen.

4 Ein Verfahren mit dem die Integrität von Programmen in Sensoren sichergestellt und Manipulationen erkannt werden können.

5 Der IEEE 802.11a Standard (IEEE 1999) wurde zwar schon früher veröffentlicht, jedoch fand die Technik erst mit höheren Bandbreiten durch IEEE 802.11g aus dem Jahre 2003 hohe Verbreitung.

schränkt. Prinzipiell ist weder eine signifikante Zunahme noch Abnahme von Publikationen zu beobachten. Der prozentuale Anteil am gesamten Publikationsvolumen ist sehr gering. Die Begriffe in diesem Cluster konzentrieren sich um das Thema „Keying", „Encryption" und „Key Exchange", markante Wörter sind z.b. *ttp* (trusted third party), *prime*, *handshak* (handshake), *protocol*, *ipsec*, *hash*, *exchang* (exchange). Typische Titel in diesem Cluster sind *A Framework for Password-Based Authenticated Key Exchange* und *Just Fast Keying: Key Agreement in a Hostile Internet*. Es geht also um die Anwendung von Kryptographie zur Absicherung von Netzwerkprotokollen.

4.1.1.4 *Cluster 3 - Intrusion Detection*

Wie bei Cluster 2 besteht auch der Cluster 3 aus lediglich neun Dokumenten, wobei allerdings eine Konzentration von Veröffentlichungen im Jahr 2000 stattfindet. Auf Grund der Größe des Clusters kann auch hier keine valide Aussage getroffen werden, allerdings ist ein leichter Aufwärtstrend des Themas innerhalb der vergangenden Jahre durchaus erkennbar. Diese Beobachtung ist interessant, da sich die Dokumente dieses Clusters sehr eindeutig einem zentralen Thema zuordnen lassen. Zu den Wörtern, welche den Cluster gut beschreiben und auch in der Zusammenfassung hoch bewertet wurden, zählen u.a. *alarm*, *intrus* (instrusion), *detect*, *train*, *anomali* und *detector*. Dies sind alles Begriffe, die sich dem Themenbereich der „Intrusion Detection" befassen. Da innerhalb dieses Forschungsfeldes häufig Anomalieerkennungsverfahren eingesetzt werden, welche sich mit maschinellen Lernverfahren realisieren lassen, stellen Begriffe wie *train* (im Sinne von trainieren), *anomali* (anomaly), *learn* und *bayesian*[6] gute Clusterbeschreibungen dar. Dies lässt sich auch in den Titel-Zusammenfassungen erkennen, der Begriff „Intrusion Detection" dominiert dabei die Titel der Veröffentlichungen: *A Framework for Contructing Features and Models for Intrusion Detection Systems*, *Evaluation of Intrusion Detection Systems Under a Resource Constraint* und *Temporal Sequence Learning and Data Reduction for Anomaly Detection*.

4.1.2 *Computers & Security*

Die Dokumente im „Computers & Security"-Corpus wurden mit einem *Eps* von 0.6 und *MinPts* von 2 geclustert. Dabei wurden 91 Cluster gefunden, wobei ein Großteil der Cluster aus weniger als zehn Dokumenten besteht. Der gesamte Anteil von Noise-Points lag knapp über 40%, wobei dieser Anteil in den Zeitabschnitten 1987 bis 1990 und 1995 bis 2000 stark abnimmt, dafür ab dem Jahr 2000 wieder ansteigt und dann auch über die 40%-Grenze hinausgeht.

4.1.2.1 *Cluster 11 - Random Bits & Bytes*

Der größte Cluster in diesem Corpus umfasst 182 Dokumente und erstreckt sich über einen Zeitraum von ungefähr 1985 bis zur Jahrhundertwende im Jahr 2000. Ganz deutlich sind zwei starke Maxima zu erkennen bei denen die Anzahl der Publikationen zunimmt. Der erste Anstieg findet zwischen den Jahren 1986 und 1991 statt und findet seinen Höhepunkt im Jahre 1989. Danach fällt die Anzahl der Veröf-

6 Bezug auf Lernenverfahren mit Hilfe von Bayesscher Wahrscheinlichkeitbegriffe.

fentlichungen im Cluster stark ab und erreicht einen erneuten starken Anstieg 1995 mit einem Höhepunkt in den Jahren 1996, 1997 gefolgt von einem Abstieg bis zum Jahr 2000. Abbildung 19 zeigt diesen Verlauf.

Die Wörter aus der Wort-Zusammenfassung grenzen kein spezielles Thema ein, vielmehr finden sich Begriffe, die dem Thema „Virus“ und „Malware“ zugeordnet werden können, wie z.B. *virus*, *disk*, *infect*, *macro*, *worm*. Allerdings finden sich auch Wörter, die eher zum Gebiet der Netzwerkinfrastruktur gehören, wie *router*, *gateway*, *proxi* (proxy), *server*, sowie Begriffe zum Thema Kryptographie: *encrypt*, *ssl*, *pgp* (Pretty Good Privacy[7]). Die Titel-Zusammenfassung weist darauf hin, dass anscheinend viele Dokumente, die zu Computers & Security *Random Bits & Bytes* Reihe gehören, in diesem Cluster zusammengefasst worden sind. Da diese Beiträge verschiedene Themen ansprechen, ist die Wahrscheinlichkeit sehr hoch, dass auch andere Dokumente diesem Cluster zugeordnet werden, welche nicht zu der *Random Bits & Bytes* Reihe gehören[8].

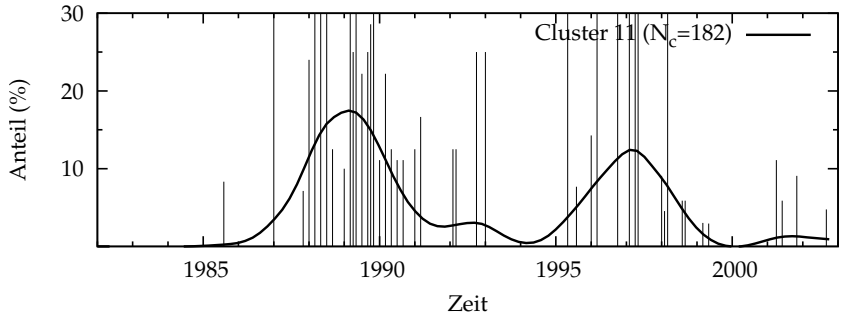


Abbildung 19: Computers & Security Cluster 11

4.1.2.2 *Cluster 1 - Network & Web Security*

Cluster 1 besteht aus 97 Dokumenten und konzentriert sich stark auf den Zeitraum zwischen den Jahren 1994 und 2000, wobei ein deutlicher Höhepunkt zwischen den Jahren 1996 und 1997 auszumachen ist. Der Cluster wird eindeutig von Begriffen dominiert, die in den Bereich Netzwerksicherheit fallen mit einem starken Fokus auf Websicherheit. So werden Wörter wie *firewal* (firewall), *java*, *applet*, *internet*, *intranet*, *activex*, *network*, *mail*, *hacker*, *authent* (authentication, authenticate) und *download* besonders gut bewertet. Die Titel in der Zusammenfassung umfassen verschiedene Themenbereiche innerhalb der Netzwerk- und Internetsicherheit und halten sich etwas allgemeiner als die Wortliste: *Anti-Virus belongs under the rug*, *Internet firewalls offer no protection against the enemy within*, *Privacy issues stir online passion*. Der Verlauf des Publikationsvolumens des Clusters über die Zeit ist in Abbildung 20 zu sehen, deutlich sieht man dort den starken Anstieg ab 1994 und die ebenfalls sehr deutliche Abnahme von Publikationen in diesem Cluster nach dem Jahr 1997. Dieser Anstieg stimmt zeitlich mit dem zweiten Maximum in Cluster 11 überein. Es ist daher vermutbar, dass sich Teile dieses Clusters mit Cluster 11 überlappt haben und in diesen eingegangen sind[9].

7 Programm zum Verschlüsseln und Signieren von Daten.

8 Das hieran beobachtbare Phänomen, das anhand struktureller/formaler Aspekte geclustert wurde, wird in Abschnitt 4.2.2.1 angesprochen.

9 Eeitere Überlegungen dazu in Abschnitt 4.2.1.2.

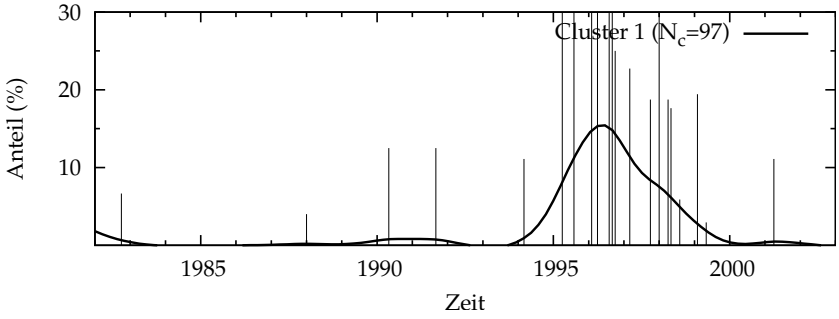


Abbildung 20: Computers & Security Cluster 1

4.1.2.3 *Cluster 43 - Audits & Wireless Security*

Der drittgrößte Cluster fasst 36 Dokumente zusammen und erstreckt sich in einem Zeitraum von 1994 bis zum Jahr 2003. Innerhalb dieses Zeitraums sind kleine Erhebungen während des Jahres 1995 und 1997 zu erkennen, sowie in der Zeit zwischen 2000 bis 2002. Dies bedeutet tendenziell eine leichte Zunahme der Cluster-Größe. Die Zusammenfassung der Wörter besitzt sowohl Begriffe aus dem Gebiet der Netzwerksicherheit, wie z. B. *wireless*, *privaci* (privacy), *network*, *cryptograph* (cryptography, cryptographic), *lan*, als auch Begriffe, die eher mit Projektführung und Unternehmensmanagement in Beziehung gebracht werden, so wie *busi* (business), *solut* (solution), *outsourc* (outsource), *staff*, *skill*, *polici* (policy), *control*. Auffällig ist auch, dass in den Titelangaben als auch in den Wörtern der Begriff *audit* sehr präsent ist. So findet man in den Titeln u.a. Veröffentlichungen wie *COBIT Audit guidance on effective implementation*, *Automated audit Tools techniques* oder auch *Internal vs. external IT audits or Mapping out a war zone?* und *Standards The need for a comon framework*. Der starke Fokus auf Audits lässt sich allerdings nicht in der Zusammenfassung der Sätze wiederfinden, wo es meist um Wirelesstechnologien geht.

4.1.2.4 *Cluster 33 - ITSEC & Security Evaluation*

Der Cluster 33 besitzt 21 Dokumente und konzentriert sich um den Zeitraum 1995, wobei in den vorangegangenen Jahren ein leichter Anstieg seit ca. 1991 zu verzeichnen ist. Eine Abbildung der Veröffentlichungen über die Zeit dieses Clusters findet sich in Abbildung 21.

Dominante Wörter in diesem Cluster sind *itsec* (ITSEC[10]), *evalu* (evaluation, evaluate), *criteria*, *certif* (certificate, certification), *icsa* (ICSA Labs[11]) und *assur* (assurance, assure). Allerdings finden sich auch Begriffe, die sich thematisch eher dem Themengebiet Computerkriminalität zuordnen lassen, wie z.B. *pirat* (pirate), *theft*, *piraci* (piracy), *caast* (CAAST[12]) und *polic* (police). Das Thema Computerkriminalität scheint also teilweise mit in den Cluster geraten zu sein. Betrachtet man allerdings die Titel in der Cluster-Zusammenfassung scheint der Großteil der Dokumente des Clusters tatsächlich ITSEC und Sicherheitsevaluationen zu umfassen: *An evaluation of HP-UX (UNIX) for database*

10 Information Technology Security Evaluation Criteria, ein technisch orientierter Standard um Computersysteme und Softwarekomponenten auf Sicherheitsaspekte zu prüfen.

11 ICSA Labs (International Computer Security Association) bietet Zertifizierungsprogramme für Sicherheitssoftware insbesondere Anti-Virus-Programme an.

12 Canadian Alliance Against Software Theft, Wirtschaftsverband zur Bekämpfung von Softwarepiraterie in Kanada.

protection using the European ITSEC, *Computer security evaluation: Developments in the European ITSEC programme*, *Security evaluation criteria* und *Information security management: The second generation*.

Der Verlauf des Clusters deckt sich in etwa mit der Erarbeitung des ITSEC Standards seit 1991 und der Verabschiedung im Jahr 1998[13].

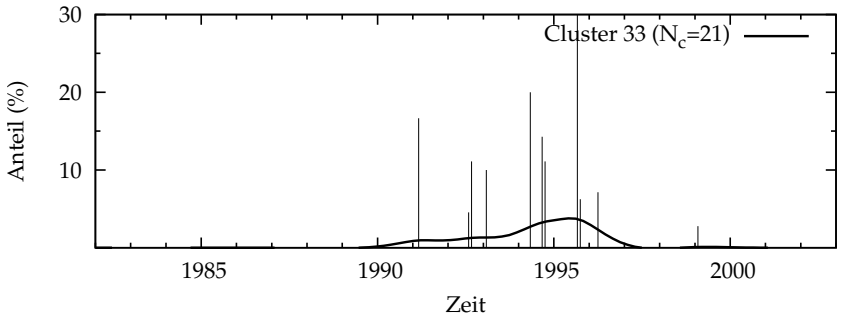


Abbildung 21: Computers & Security Cluster 33

4.1.2.5 *Weitere Besonderheiten*

Neben den Clustern, die aufgrund ihrer Größe eine gewisse Relevanz besitzen, haben sich auch noch andere charakteristische Cluster gebildet. So ist Cluster 14 eine Ansammlung von „Editorials“, die scheinbar ebenso wie die „Random Bits & Bytes“ anhand formaler Merkmale und nicht an Inhalten geclustert wurde. Ein inhaltlich dagegen sehr klarer Cluster ist Cluster 51, in dem es hauptsächlich um internationale Hacker Aktivitäten geht. So dominiert das Stichwort *mitnick* was für Kevin Mitnick, einem bekannten Hacker, steht. Mitnick wurde 1995 festgenommen und 1999 aufgrund mehrerer Computerverbrechen zu einer Gefängnisstrafe von knapp vier Jahren verurteilt[14]. Genau um die Jahre 1995 und 1999 konzentrieren sich dann auch die Publikationen des Clusters.

4.1.3 *IEEE*

Der IEEE-Corpus ist mit einer Gesamtanzahl von mehr als 37.000 Dokumenten der größte untersuchte Corpus. Die Veröffentlichungen beginnen bereits mit einigen wenigen frühen Beiträgen im Jahr 1957. Jedoch sind dies so wenige Publikationen, dass meist eine Auswertung nicht sinnvoll ist. Wir betrachten deshalb nur den Bereich ab 1980. Ab 1988 entsteht ein kontinuierlicher Publikations-Strom, der langsam bis bis zum Jahr 2002 ansteigt und zu diesem Zeitpunkt etwa 130 Veröffentlichungen pro Monat beträgt. Dann macht die Kurve einen Knick und die Anzahl der Publikationen wächst durchgehend sehr steil nach oben an. Der Höhepunkt wird 2008 mit etwa 560 Veröffentlichungen pro Monat erreicht und sinkt dann wieder ab. Der Corpus wurde mit einem *Eps* von 0.32 und *MinPts* von 2 geclustert, was in 883 Cluster resultierte. Dabei wurden etwas mehr als 20% aller Dokumente geclustert und die restlichen Dokumente als Noise-Points klassifiziert.

Die erhaltenen Cluster aus dem IEEE-Corpus unterscheiden sich in ihrem Kurvenverlauf als auch inhaltlich scheinbar nur geringfügig. Der

13 Vgl. ITSEC (1998) und Bundesamt für Sicherheit in der Informationstechnik (2008b).
14 Vgl. U.S. Departement Of Justice (1999).

zeitliche Verlauf ist bei allen Clustern sehr ähnlich. Die Anzahl der Publikationen ist in den Anfangsjahren, also von 1980 bis Ende der 90er Jahre nur sehr gering und bei den kleineren Cluster streckenweise sogar gleich Null. Vereinzelte Spitzen in diesem Zeitabschnitt ragen meist deutlich hinaus, was jedoch an der geringen Anzahl der Gesamtpublikationen liegt. Der Cluster 0 des IEEE-Corpus ist exemplarisch in Abbildung 22 dargestellt.

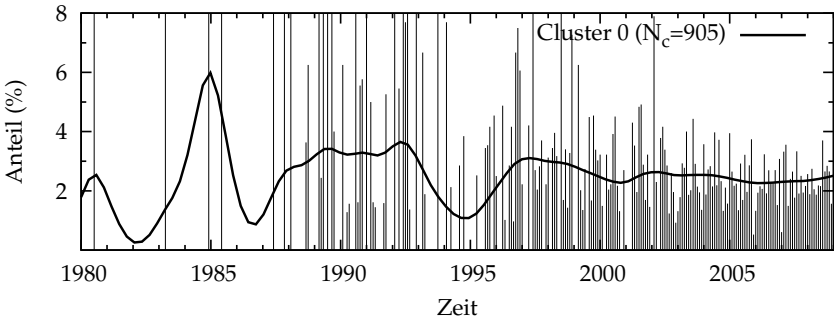


Abbildung 22: IEEE Cluster 0

Aus den Zusammenfassungen der verschiedenen Cluster lässt sich nur schwer ein durchgängiges Thema der einzelnen Cluster ablesen. Interessanterweise treten aber in fast allen Cluster die gleiche Menge thematisch sehr spezieller Begriffe auf. Diese würden für sich jeweils ein Thema umreißen und so sollte man sie eher in einem speraten Cluster vermuten. Besonders in den größeren Clustern treten Wörter wie *watermark*, *imag* (image), *node*, *sensor*, *packet* und *attack* auf. Die beiden letzteren Begriffe mögen ebenfalls in verschiedenen Themengebieten vorkommen, allerdings sind die beiden Begriffe „watermark“ und „image“ sehr spezifisch, das gleiche gilt für „sensor“ und „node“, wenn auch in etwas abgeschwächterem Maße. Der Umstand, dass diese Wörter keinen eigenen Cluster besitzen, sondern in den Zusammenfassungen fast aller Cluster vorkommen, wird in Abschnitt 4.2.1.3 genauer untersucht.

Prinzipiell lässt sich bei dem vorliegenden IEEE-Clustering beobachten, dass sowohl die Wort-Zusammenfassungen als auch die Satz-Zusammenfassungen um so allgemeiner werden, je größer die Cluster werden. So finden sich in Cluster 0 mit 905 Dokumenten Wörter, die sich thematisch sehr unterschiedlich einordnen lassen. So gibt es Begriffe, die mit Netzwerksicherheit assoziiert werden, wie z.B. *attack*, *packet*, *servic* (service), *intrus* (intrusion) und *proxi* (proxy), aber auch Begriffe, die eher eine elektrotechnische, sehr Hardware-nahe Bedeutung haben, wie *power*, *voltag* (voltage), *devic* (device), *embed* (embedded), *energi* (energy) und *frequenc* (ferquency). Ähnliche Beobachtungen lassen sich auch bei den Titel-Zusammenfassungen machen. So finden sich dort Titel von Arbeiten aus dem Bereich „Wireless Security“ wie im Falle von *Energy efficient watermarking on mobile devices using proxy-based partitioning* und *Analysis of payment transaction security in mobile commerce*, als auch Themen zur allgemeinen Netzwerksicherheit: *Mitigating Denial-of-Service File Attacks on the Chord Overlay Network*, *Design issues in mobile agent programming systems* und *TVA: A DoS-Limiting Network Architecture*. In der Zusammenfassung von Cluster 0 befinden sich viele Themen, die sehr neue Technologien ansprechen, das betrifft vor allem

der Bezug auf Wireless-Netzwerke und mobile Geräte. Da aber der Cluster viele Dokumente aus dem Zeitraum 1988 bis 1992 aufweist, können dies nicht die alleinigen Themen sein, da die angesprochene Technologie damals noch nicht vorhanden war. Zu vermuten ist also, dass verschiedene Themen in diesem Cluster zusammengefasst wurden und die Bereiche „Wireless Security", sowie Netzwerksicherheit lediglich dominante Themen sind, die in der Zusammenfassung als erstes genannt werden.

Alle Cluster innerhalb des IEEE-Corpus haben mit ähnlichen Problemen zu kämpfen wie der eben betrachtete Cluster 0. Besonders aber die großen Cluster sind schwer in eine Kategorie einzuordnen. Cluster 13, mit 680 Dokumenten der zweitgrößte Cluster, beinhaltet zum Beispiel ebenfalls Hinweise auf das Verschmelzen verschiedener Themen. So befinden sich dort Arbeiten über VoIP-Technologien wie z.B. *Securing VoIP and PSTN from integrated signaling network vulnerabilty* und *VPN Analysis and New Perspective for Securing Voice over VPN Networks*, als auch zum Thema „Grid Computing" , wie zu sehen in *Security issues in on-demand grid and cluster computing* und in *Security impleciations of typical Grid Computing usage scenarios* sowie „Wireless Security", wie z.B. *Toward Secure Low Rate Wireless Pesonal Area Networks* und *JANUS: A Framework for Scalable and Secure Routing in Hybrid Wireless Networks*. Auch über die Themen der kleineren Cluster des Corpus lässt sich nur schwer eine Aussage treffen. So finden sich im Cluster 77, dem sechstgrößten Cluster im Corpus mit 93 Dokumenten, Begriffe wie *seismic*, *power*, *gravit* (gravitation), *univers* (universe, university), *voltag* (voltage) und *warp*. Dies sind alles sehr physikalische Begriffe, die sich nur schwer mit den anderen, sehr technischen Wörtern aus dem Cluster vereinbaren lassen. Aufgrund der schlechten Qualität der Cluster kann daher keine Aussage über den Diskurs über Informationssicherheit anhand des IEEE-Corpus gemacht werden. Genauere Gründe für das teilweise Versagen der Methode werden in Abschnitt 4.2.1.2 diskutiert.

4.1.4 *Security-Basics*

Aus der Mailingliste Security-Basics wurden knapp 15.500 Dokumente in Form von Diskussionsthreads untersucht. Die Daten wurden mit einem *Eps* von 0.32 und *MinPts* von 3 geclustert. Diese Parameterwahl führt allerdings dazu, dass lediglich nur knapp 25% aller verfügbaren Dokumente durch den Algorithmus kategorisiert werden, während der Großteil als Noise-Points markiert wurde. Jedoch verhindert diese Parameterwahl das sehr frühe Zusammenfallen der Cluster im Security-Basics-Corpus. Die Mailingliste beinhaltet Beiträge vom Zeitraum 2002 bis zum Jahr 2009, wobei die Anzahl der Beiträge stetig abnimmt. Beginnend bei mehr als 300 Beiträgen pro Monat ist die Mailingliste heute nur noch bei weniger als 80 E-Mails pro Monat angekommen.

4.1.4.1 *Cluster 1 - Low Level Network Security*

Der erste Cluster umfasst 550 Dokumente und weist ein kontinuierliches Abfallen über den gesamten Zeitraum auf. Von einem prozentualen Anteil von 6% ist die Beitragsrate auf 1% gesunken. Die Beiträge in den Clustern beinhalten viele Wörter, die sich thematisch auf Netzwerksicherheit beziehen, wobei ein starker Fokus auf tiefere Protokollschichten und Sicherheitsproblemen in der Transportschicht des

Internets auszumachen ist und nur wenige anwendungsbezogene Problemfelder angesprochen werden. Einige sehr kennzeichnende Begriffe sind u.a. *port*, *tcp*, *firewal*, *scan*, *udp*, *packet*, *filter*, *nmap* und *address*. Dieser starke Fokus auf infrastrukturelle Aspekte der Netzwerksicherheit finden sich auch in den Titel-Zusammenfassungen wieder; Fachbegriffe wie *TCP*, *port* oder *UDP* sowie Anwendungen zur Untersuchung von Netzwerkverkehr werden häufig erwähnt, wie zu sehen in *nc help needed*, wobei *nc* das Netzwerk-Werkzeug „netcat“[15] bezeichnet. Weitere markante Titel sind *Nmap Under the hood*, *Windows 2000 server ports, services to close*, *What is this port? Is it a trojan?*, *TCP DNS requests* und *Port TCP/8000*. Das Publikationsvolumen der Beiträge in diesem Cluster ist in der Abbildung 23 dargestellt. Der kontinuierliche Abfall der Beiträge zu dem Thema dieses Clusters ist dort deutlich sichtbar.

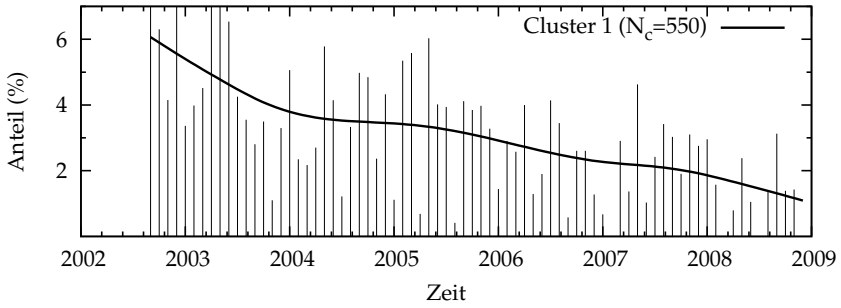


Abbildung 23: Security-Basics Cluster 1

4.1.4.2 *Cluster 11 - Passwords & Access Control in Networks*

Als zweitgrößter Cluster umfasst Cluster 11 insgesamt 341 Dokumente. Die Anzahl der Beiträge steigen in diesem Cluster bis 2004 leicht an und beginnen ab dann sehr leicht, aber kontinuierlich abzunehmen. Aus der Cluster-Zusammenfassung lassen sich zwei Themen heraus lesen. Die Wort-Zusammenfassung hat einen sehr starken Fokus auf Zugriffskontrollmechanismen in Netzwerken. Wörter wie *vpn*, *polici* (policy), *admin*, *account*, *ssl*, *trust* und *access* führen die Liste an. Am stärksten hingegen wird der Begriff *password* bewertet. Die Titel-Zusammenfassung hat demnach viele Beiträge ausgewählt, die sich mit Passwörtern befassen, während die in der Wortliste dominanten Themen der Zugriffskontrollmechanismen lediglich unterrepräsentiert sind. Dies zeigt deutlich, dass DBSCAN dazu neigt, überlappende Cluster zusammenzuführen, diese Problematik wird anschließend in Abschnitt 4.2.1.2 genauer diskutiert. Einige typische Titel sind: *Minimum password requirements*, *A doable frequent password change policy?* und zum Thema „Access Control“ *Deploying SSL-based VPNs* und *VPN Client and Local Service*.

4.1.4.3 *Cluster 6 - Audits, Compliance & Standards*

Der drittgrößte Cluster besitzt 77 Beiträge, die im Frühjahr 2007 beginnen und im Sommer 2008 enden. Thematisch lässt sich der Cluster

15 Ein Programm, mit dem man automatisch oder interaktiv Daten an einem beliebigen Port senden kann, sowie selbst an einem Port lauschen kann um Daten zu empfangen. Netcat wird häufig zum Testen von Netzwerksoftware oder Finden von Fehlern im Netzwerk benutzt und bezeichnet sich in seiner Manpage selbst als das Schweizer Taschenmesser für TCP/IP.

anhand der Wortliste in das Gebiet „Audits, Compliance & Standards“ einordnen, so sind Wörter wie *risk*, *cost*, *standard*, *complianc* (compliance, compliant), *econom* (economy, economical), *partnership*, *audit*, *certif* (certificate, certify) sehr präsent. Die Titel-Zusammenfassung unterstützt diese Beobachtung, zeigt aber auch, dass Beiträge zu Netzwerk-Scannern in dem Cluster vertreten sind: *How does a customer get PCI audited?*[16], *Preparing for GSEC*[17], *Concepts: Security and Obscurity*, besonders auffällige Titel zu Beiträgen zu Netzwerkscannern sind *Nessus Scan*[18], sowie *Web Application Vulnerability Scanner*.

Ein weiteres bedeutendes Merkmal dieses Clusters findet sich in der Zusammenfassung der Sätze. Besonders häufig ist dort die E-Mailsignatur von Craig S. Wright zu finden, ein zertifizierter Fachmann für GSEC mit Goldstatus, der sehr aktiv auf der Mailingliste geschrieben hat[19].

Da Herr Wright zumindest in der Zeitspanne über die sich der Cluster erstreckt seine E-Mailsignatur nicht gemäß RFC3676 kennzeichnete, so dass sie während des Vorverarbeitungsschrittes weggeschnitten werden konnte, sind seine Beiträge in diesem Cluster stark vertreten. Dies könnte eine gute Erklärung sein warum auch andere Themen, welche nicht direkt mit Sicherheitsmanagement zu tun haben diesem Cluster zugeordnet wurden. Der Verlauf des Clusters über die Zeit ist in der Abbildung 24 zu sehen.

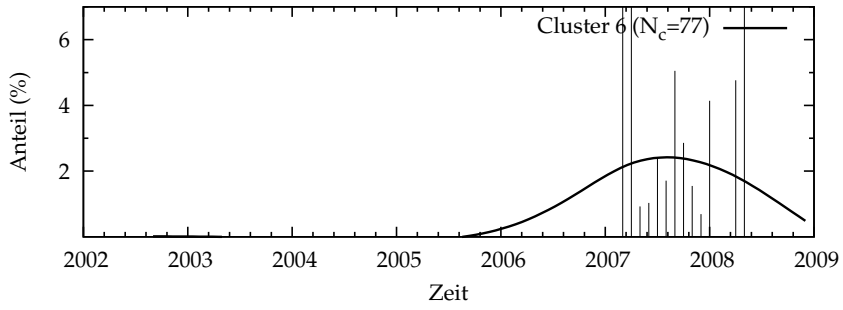


Abbildung 24: Security-Basics Cluster 6

4.1.4.4 *Cluster 59 - Personal Certification*

Cluster 59 kommt mit nur 41 Dokumente von der Größenordnung nach an siebter Stelle im Security-Basics-Corpus. Zu erkennen ist ein leichter Anstieg an Beiträgen. Ähnlich wie Cluster 6 kommen auch hier häufig Wörter vor, die im direkten Zusammenhang zum Sicherheitsmanagement stehen. Besonders prägnante Wörter sind: *certif* (certificate, certify), *expir* (expire), *cissp* (CISSP[20]), *job*, *liabil* (liability, liable), *company*, *verifi* (verfification, verfiy), *credibl* (credibility). Ein zweites Thema, das sich in der Wortliste wiederfindet ist Kryptographie, so sind ebenfalls Wörter wie *rsa*, *encrypt*, *pgp* und *sign* vorhanden. Schaut man hingegen

16 PCI steht für Payment Card Industry, eine Vereinigung, welche den PCI Data Security Standard (PCI-DSS) etabliert hat(vgl. PCI Security Standards Council, LLC 2008).

17 GSEC steht für „GIAC Security Essentials Certification“.

18 Nessus ist ein sog. „Vulnerability Scanner“ zum Finden von Sicherheitslücken in Netzwerken.

19 Mehr Informationen zur Person auf dem Blog Wright (2009) und der Liste der GSEC-zertifizierten Security Professionals GIAC (2009).

20 Certified Information Systems Security Professional, eine vom ISC angebotene Zertifizierung, die auf Wissen im Bereich Computersicherheit prüft.

auf die Titel-Zusammenfassung ist wieder bis auf einige Ausnahmen ein starker Fokus auf Zertifizierung, insbesondere CISSP zu finden: *Value of certifications*, *Expired certificates*, *CISSP Prep books?*, *University Degree or CISSP*.

In dem Cluster geht es also im großen Maß um die Zertifizierung von Fähigkeiten einzelner Personen. Jedoch vermischt sich das Thema aufgrund der Gleichheit der Wörter auch mit der Zertifizierung und Zertifikaten im Sinne Kryptographie. Es handelt sich also wieder um einen Fall in dem Cluster verschmolzen sind. Die Zertifizierung persönlicher Fähigkeiten scheint jedoch dominanter zu sein.

4.1.5 *Infosec-News*

Infosec-News ist mit knapp 13.000 untersuchten Threads der kleinere E-Mail-Corpus und beginnt Mitte des Jahres 1999. Während der ersten Jahre steigt die Anzahl der Beiträge rapide an und erreicht ihren Höhepunkt um 2002. Danach bleibt die Aktivität der Mailingliste relativ konstant und schwankt leicht zwischen 100 bis 120 Beiträgen pro Monat. Die Cluster wurden mit einem *Eps* von 0.51 und *MinPts* von 2 erstellt. Insgesamt ergibt dies 434 Cluster, von denen lediglich die ersten 16 Cluster mehr als 30 Dokumente vorzuweisen haben. Insgesamt wurden 40% der Dokumente geclustert, so dass über die Hälfte aller Beiträge als Noise-Points kategorisiert wurden.

4.1.5.1 *Cluster 11, 22, 33, 26 und 0 - Newsletter*

Das besondere an den erhaltenen Cluster des Infosec-News-Corpus ist, dass sie sich sehr stark an Kolumnen und regelmäßig erscheinenden Newslettern orientieren. So wurden Cluster für den „Linux Advisory Watch“, „Secunia Weekly Summary Issue“, dem „ITL Bulletin“, sowie die Ausgaben von „Windows IT Pro - Security Update“ gefunden.

Cluster 11 stellt mit 694 Dokumenten den größten Cluster dar. Die Titel-Zusammenfassung liefert einen deutlichen Hinweis darauf, dass es sich um eine Aggregation der „Linux Advisory Watch“-Meldungen handelt. Auch die Wortliste bestätigt dies, so erhalten wir als meist bewertete Begriffe Wörter aus dem einleitenden Text der „Linux Advisory Watch“ Meldungen: *linuxsecur* (linuxsecurity[21]), *dave* (Dave Wreski, einer der beiden Autoren), *newslett* (newsletter), *linux*, *advisori* (advisory) und *weekli* (weekly). Desweiteren sind die Namen einiger Firmen aus dem Linuxumfeld, sowie bekannte Namen von Linuxdistributionen in der Wortliste zu finden, welche den Fokus auf Linux in diesen Meldungen betonen, z.B. *debian* , *caldera*, *mandrak* (Mandrake), *debian* und *freebsd*[22]. Der Cluster 11 folgt mit seinem Anstieg um das Jahr 2004 herum der insgesamten Abnahme von Dokumenten zur gleichen Zeit im gesamten Corpus. Zu erklären ist dies mit der konstanten Anzahl von Dokumenten pro Monat, die durch den Newsletter verursacht werden. Da die „Linux Advisory Watch“-Meldungen wöchentlich erscheinen, muss also die absolute Anzahl der Beiträge immer gleich vier sein. Der Cluster besteht allerdings nicht aus allen Newsletter-Meldungen, besonders am Anfang um das Jahr 2000 herum, sowie gegen Ende ist der prozentuale Anteil der Beiträge in Corpus 11 zu gering um

21 Der „Linux Advisory Watch“ ist der wöchentliche Newsletter von `linuxsecurity.com`.
22 FreeBSD ist zwar streng genommen kein Linux-System, aber ein Unix-ähnliches freies Betriebsystem wie Linux.

wöchentlichen Meldungen zu entsprechen. Es dürften sich also auch leicht andere Beiträge mit ähnlichen Inhalten zu einem speziellen Newseintrag innerhalb der „Linux Advisory Watch"-Meldung in diesem Cluster befinden.

Die Beobachtungen, die hier ausführlich für Cluster 11 zusammengetragen wurden, lassen sich in ähnlicher Weise auch für andere Cluster machen. So fasst Cluster 22 mit 409 Dokumenten viele „Secunia Weekly Summary Issue"-Artikel zusammen und Cluster 33 umfasst mit 180 Beiträgen monatliche „ITC Bulletin"-Nachrichten. Welche Newsletter genau zusammengefasst werden, sind eindeutig aus der Zusammenfassung der Titel abzuleiten, in denen direkt die Betreffzeile der jeweiligen Newsletter-Meldungen stehen.

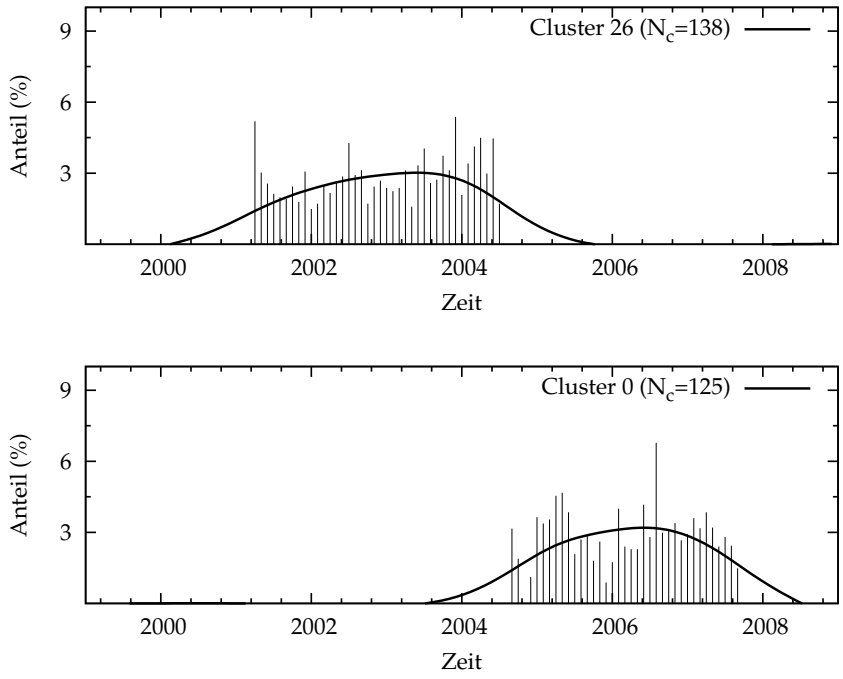


Abbildung 25: Infosec-News Cluster 26 und 0

Ein interessantes Phänomen lässt sich noch bezüglich der Clustern 26 und 0 beobachten, deren Verläufe in Abbildung 25 dargestellt sind. Beide Cluster beinhalten in ihrer Titel-Zusammenfassung „Security Update"-Meldungen, verlaufen aber zeitlich sehr anders. Während Cluster 26 beginnend im Jahr 2001 bis 2005 reicht und eine konstante Beitragsanzahl pro Monat aufweist, beginnt Cluster 0 im Jahre 2005 und läuft mit ebenso konstanter Beitragszahl bis Mitte 2007 weiter. Schaut man in die Wort-Zusammenfassung, so führt *winnetmag* die Liste des ersten Clusters an und *windowsitpro* die Wortliste des zweiten Clusters. Tatsächlich wurde der Name des Online Magazins, welches für die Veröffentlichung der „Security Update"-Meldungen verantwortlich ist von „Windows & .NET Magazine" auf „Windows IT Pro" geändert. Ein Erhöhen des Cluster-Radius würde diese beiden Cluster mit hoher Wahrscheinlichkeit zusammenfallen lassen. Ebenso dominieren bei dem Clustering jedoch leider wieder die strukturellen Aspekte gegenüber den inhaltlichen Merkmalen, wie bei allen Newsletter-Clustern.

4.1.5.2 *Cluster 3 - Microsoft Security*

Cluster 3 ist mit einem Umfang von 349 Dokumenten der drittgrößte Cluster in diesem Corpus. Er erstreckt sich über den gesamten Zeitraum. Er beginnt um 2000 mit wenigen Beiträgen und steigt stetig bis zur Mitte des Jahres 2006 an und sinkt dann wieder leicht ab.

Thematisch sind viele Begriffe enthalten, die allgemein mit Windows und Microsoft assoziiert werden. Die beiden am höchsten bewerteten Begriffe sind *microsoft* und *windows*. Weitere Wörter, die sich mit Microsoft oder deren Betriebssystemen beschäftigen, sind z.B. *zotob* (Zotob)[23], *wmf* (Windows Metafile[24]), *rutkowska*[25], sowie *eey* (eEye) und *maiffret*[26]. Auch die Titel in der Zusammenfassung beinhalten zum großen Teil diese beiden Wörter, so dass der Bezug auf Windowssicherheit sehr deutlich wird: *Wait for Windows patch opens attack window*, *Major graphics flaw threatens Windows PCs*, und *Microsoft Patches 20 Security Vulnerabilities*.

Zeitlich gesehen hat Microsoft mit dem „Trustworthy Computing" Memo von Bill Gates[27] im Jahr 2002 Sicherheit zur einer hohen Priorität gemacht. In der Tat lässt sich direkt in 2002 ein Peak im Cluster feststellen. Der weitere Verlauf der Kurve ließ sich jedoch bisher nicht durch historische Veränderungen erklären.

4.1.5.3 *Cluster 32 - Cybersecurity & Cyberwar*

Cluster 32 ist mit 92 Dokumenten der kleinste unter den analysierten Clustern und besitzt Beiträge über die gesamte betrachtete Zeitperiode hinweg. Es gibt einen erkennbaren Anstieg ab dem Jahr 2001 der bis 2005 anhält. Nach 2005 nimmt die Häufigkeit des Themas dann langsam wieder ab. Der zeitliche Verlauf der Beiträge in diesem Custer ist in der Abbildung 26 dargestellt. Markante Wörter dieses Clusters sind vor allem *cybersecur*, *homeland*, *govern* (government), *nation*, *depart* (department), *bush* (George W. Bush), *presid* (president), *cia*, *cyberattack* und *cyber*. Dies sind also alles Begriffe aus dem Bereich „Cyberwar" und „Homeland Security" mit sehr starkem Bezug auf die Vereinigten Staaten. Die Titel-Zusammenfassung deutet ebenfalls sehr klar auf dieses Thema: *U.S. cybersecurity due for FEMA-like calamity?*, *Anti-Terror Pioneer Turn In the Badge*, *Bush Approves Cybersecurity Strategy*, *Bush Needs To Ramp Up In New Year*, *White House Officials Debating Rules for Cyberwarfare*. Der Anstieg dieses Clusters fällt dabei genau auf die Amtsperiode des damalige U.S. Präsidenten George W. Bush und den von seiner Regierung geführten „Kampf gegen Terrorismus", in dessen Zuge im Jahr 2003 der Irakkrieg begonnen wurde. Außerdem wurde 2002 das „Departement of Homeland Security" eingerichtet, was sicherlich ein wichtiger Grund für die deutliche Zunahme dieses Clusters ist.

23 Ein Wurm aus dem Jahr 2005, der Windows 2000 Rechner angriff.

24 Ein proprietäres Grafikformat von Microsoft, welches durch Windows-eigene Routinen verarbeitet wird und dadurch zum Einfallstor für Schadcode und Angriffe wurde.

25 Johanna Rutkowska, eine bekannte, polnische Sicherheitsexpertin, die sich mit dem Verstecken von Malware unter Windows Vista beschäftigte (Blue Pill).

26 Marc Maiffret ist der Mitgründer von eEye Digital Security, eine Computersicherheitsfirma, die in Zusammenarbeit mit Microsoft Sicherheitslücken im IIS geschlossen hat.

27 Siehe Gates (2001).

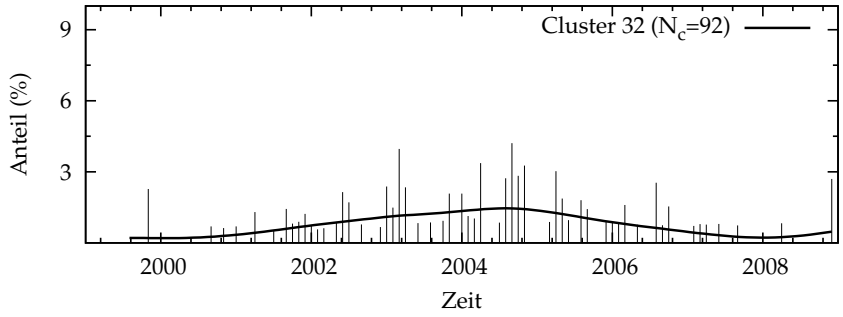


Abbildung 26: Infosec-News Cluster 32

4.1.6 *comp.sec.misc*

Comp.sec.misc ist die einzige untersuchte Newsgroup und ist mit mehr als 24.000 Dokumenten die größte internet-basierte Quelle. Der Corpus wurde mit einem *Eps* von 0.32 und *MinPts* von 3 geclustert. Die resultierenden Cluster umfassen lediglich 20% aller Dokumente und verwerfen damit knapp 80% aller Threads als Noise-Points. Allerdings verhindert die Wahl der Parameter ähnlich wie bei den Corpora der Mailinglisten das frühe Zusammenfallen der Cluster. Es wurden 389 Cluster gefunden, wobei lediglich 16 von ihnen mehr als 30 Dokumente beinhalten. Die Beiträge der Newsgroup beginnen im Jahr 1992 und reichen bis etwa 2009. Zu Beginn ist ein starker Anstieg der Beiträge zu verzeichnen, der bis etwa Mitte 1995 anhält und dann zu einem Stillstand kommt. Bis zur Jahrhundertwende im Jahr 2000 schwankt die Menge der Beiträge im Bereich von 150 bis 180 pro Monat und sinkt dann kontinuierlich bis 2009 ab.

4.1.6.1 *Cluster 4 - Passwords & Cryptography*

Der größte Cluster erstreckt sich über den gesamten Zeitraum der Newsgroup und umfasst 888 Dokumente. Bis zum Jahr 1997 umfasst der Cluster 4% aller Beiträge der Newsgroup, sinkt dann aber kontinuierlich ab und endet im Jahr 2004 bei ca. 2% aller Veröffentlichungen. Er bleibt ab dann konstant bei diesem Beitragsanteil.

Zwei Themen lassen sich aus den Zusammenfassungen herauslesen. Zum einen ist dies der Bereich „Passwörter und Cracking“, welcher eher durch die Titel-Zusammenfassung bemerkbar wird und auf der anderen Seite gibt es viele Begriffe in der Wortliste, die eher mit Verschlüsselung im Allgemeinen zu tun haben. Markante Wörter sind so z.B. *password*, *encrypt*, *pgp*, *key*, *hash*, *crack*, *algorithm* und *rsa*. Diese Wörter weisen auf einen starken Fokus im Bereich Kryptographie hin und auch in den Satz-Zusammenfassungen lassen sich Anfragen zum Thema PGP und Verschlüsselung finden, beispielsweise *Hi, I want to secure my email and I don't know whether I can use PGP [...]*. Die Titel-Zusammenfassung hat hingegen einen sehr starken Fokus auf „Passwörter und Cracking“, einige Beispiele dazu wären *Unreadable password file (was can people really have trouble memorizing long passwords?*, *Strong Passwords Revisited*, *Password Crackers* und *Effectiveness of Forced Password Changing*.

4.1.6.2 *Cluster 0 - Web Security*

Cluster 0 ist der zweitgrößte Cluster und umfasst 184 Dokumente. Der Cluster beinhaltet Beiträge aus dem gesamten Zeitraum der Newsgroup. Eine auffällige Häufung von Beiträgen lässt sich in den Jahren 2000 und 2001 beobachten, bei der der Anteil der Beiträge am gesamten Veröffentlichungsvolumen von unter 1% auf über 2% steigt.

In der Zusammenfassung der Wörter tauchen verstärkt Begriffe auf, die mit dem Internet, speziell dem World Wide Web in Verbindung gebracht werden, so finden sich dort u.a. Begriffe wie *mail*, *java*, *www*, *rpc*, *mime*, *post*, *http*, *html*, *site*, *url* und *download*. Allerdings sind auch Wörter aus dem Bereich der Kryptographie vorhanden wie *encrypt*, *crypto* (cryptography, cryptografic), *trust*, *ssl* oder *rsa*. Das Vorhandensein dieser Wörter bietet eine gutes Beispiel dafür, warum Cluster bei anderer Parameterwahl, wie zum Beispiel einen größeren Eps-Radius dazu tendieren zusammenzufallen, denn anscheinend war der Cluster-Algorithmus nicht in der Lage genauer zwischen „Web Security“ und „Cryptography“ zu unterscheiden. Mehr Überlegungen dazu finden sich in der Problemanalyse in Abschnitt 4.2.

Obwohl aber Wörter aus dem Themengebiet der Verschlüsselung in diesem Cluster präsent sind, weist die Zusammenfassung der Sätze einen starken Fokus auf „Web Security“ auf, so sind dort z.B. folgende Titel zu finden: *Netscape Security Error? Help!!!*, *http to https posted data lost ! why ?* , *Network security training*, *Internet security*, *security suggestion for online documents system*. An Anstieg dieses Clusters ließe sich mit der steigenden Zahl der Internetanschlüsse weltweit begründen, doch erklärt dies nicht, warum nach 2001 die durch den Cluster präsentierten Themen einen so plötzlichen Einbruch erlitten haben. Außerdem muss in Betracht gezogen werden, dass das Clustering lediglich um die 40% aller Gesamtdokumente betrachtet und daher nicht ausgeschlossen werden kann dass ein thematisch ähnlicher Cluster mit geringerer Dichte in den Noise-Points liegt.

4.1.6.3 *Cluster 18 - Public Key Infrastructures*

Cluster 18 ist mit 179 Dokumenten der drittgrößte Cluster und erstreckt sich ebenfalls über den gesamten Zeitraum der Newsgroup, wobei außer einem Anstieg des Themas im ersten Jahr des betrachteten Zeitraums keine signifikante Veränderung der Beitragshäufigkeit zu beobachten ist. Der Inhalt des Clusters wird sowohl durch die Wortliste als auch die Zusammenfassung der Titel sehr deutlich bestimmt, so gibt es einen sehr starken Fokus auf Kryptographie und dabei speziell auf „Public Key Infrastructures“, also die Möglichkeiten öffentliche Schlüssel für asymmetrischer Verschlüsselungsverfahren auszutauschen. Typische Wörter in dieser Kategorie sind *certif* (certificate, certify), *rsa cert*, *ssl*, *key*, *crypto* (cryptography), *sign*, *trust*, *openssl*, *encrypt*, *pgp*, *verifi* (verify, verification) und *signatur*. In der Titel-Zusammenfassung finden sich diese Begriffe ebenfalls wieder, sowie beispielsweise in *What is a Certificate?*, *Invalid certificate on 'security' site.*, *Microsoft's Certificate Server Problem Solving S/Mime setup* und *private key in PKI*.

4.1.6.4 *Cluster 16 - Web-Proxy Security*

Der Cluster 16 ist mit 107 Dokumenten der sechstgrößte Cluster des Corpus und ist deshalb interessant, weil er trotz seiner Größe ein

äußerst spezifisches Thema umreißt. Die Cluster 31 und 34 beinhalten zwar mehr Dokumente, beziehen sich aber beide auf spezielle FAQs, so befinden sich in Cluster 31 hauptsächlich Beiträge zur „Firewall FAQ“ und in Cluster 34 Beiträge zur „SSL-Talk List FAQ“. Die Anzahl der Beiträge des Clusters 16 beginnen Mitte der 90er Jahre leicht zu steigen, scheinen um die Jahrtausendwende ihren Höhepunkt erreicht zu haben und sinken dann wieder leicht ab. Der Cluster betont sehr stark das Thema „Web-Proxy Security“, also die Sicherheit von HTTP-Traffic im Zusammenhang mit Application-Layer Gateways (Proxys), einer sehr beliebten Sicherheitstechnik in lokalen Netzen.

In der Wort-Zusammenfassung sind die Begriffe *proxi* (proxy), *ssl*, *server*, *tomcat* und *apache* sehr dominant. Desweiteren finden sich dort auch Begriffe die man mit typischerweise mit „Browsen im Web“ assoziieren würde, dazu zählen *site*, *page*, *browser*, *http*, *web* oder *netscap* (Netscape). Die Titel-Zusammenfassungen haben ebenfalls einen sehr starken Fokus auf den Themenbereich Sicherheit im Web und enthalten sehr oft den Begriff *proxy* und *SSL*, was den starken Bezug dieses Clusters zu dem genannten Thema weiter unterstreicht. Beispiele für typische Titel sind u.a. *Is HTML Get Method secure?*, *ISP DNS, proxies and security*, *SSL traffic via proxy server being 'intercepted'?*, *Multiple proxy servers* und *Help with SSL More info*.

4.2 ANALYSE DER ANGEWANDTEN METHODE

Man kann nicht immer alles erreichen was man sich vornimmt und so müssen auch wir in Anbetracht der eben beschriebenen Cluster-Ergebnisse feststellen, dass wir unser Ziel nicht ganz erreicht haben. Es ist deutlich zu sehen, dass nur einige Cluster präzise Themen beinhalten. In einigen Fällen wurde klar nach strukturellen und nicht nach inhaltlichen Merkmalen geclustert. Jedoch ist dies kein Grund aufzugeben, denn die Richtung scheint durchaus vielversprechend zu sein. Für die Probleme der Methode gibt es Ursachen, die ermittelt werden müssen. Eine genauere Betrachtung der Warum's und Wieso's darf also nicht fehlen und ist für das bessere Verständnis der Methode, ihren Stärken und Schwächen essentiell. Dies soll in diesem Teil der Arbeit versucht werden. Vielleicht bleibt am Ende ja doch noch etwas, was die Ergebnisse sagen können.

Wenden wir uns jedoch vorerst in aller Bescheidenheit den allzu offensichtlichen Problemen zu. Dazu werden wir hier nacheinander auf die verschiedenen auffälligen Aspekte, die sich in den Ergebnissen feststellen lassen, eingehen (Abschnitt 4.2.1) um dann anschließend eine Urteil über einzelnen Stufen der Methode zu fällen (Abschnitt 4.2.2).

4.2.1 *Besondere Beobachtungen*

Aus den Cluster-Zusammenfassung gehen einige sehr spezielle Eigenschaften hervor, die die untersuchten Text-Corpora aufweisen. Dazu zählt zum einen die ungewöhnlich hohe Anzahl von Dimensionen, die auch einen Einfluss auf die Performance des Clusterings hat. Eine Erklärung für die hohen Dimensionen muss in den Daten liegen. Um sie zu finden, müssen die Wörter des globalen Dictionaries genauer untersucht werden, dies geschieht in Abschnitt 4.2.1.1. Eine weitere Beobachtung ist die schlechte Qualität der Cluster, so ist in der Betrachtung der Cluster aufgefallen, dass dort verschiedene Themen in

einem Cluster zusammengefasst wurden. In Abschnitt 4.2.1.2 werden die Ursachen dafür in der Wortverteilung und den Eigenschaften des Feature-Raums der Datenquellen untersucht. Desweiteren ist speziell beim IEEE-Corpus aufgefallen, dass dort in fast allen Clustern sehr spezifische Begriffe auftauchen, von denen man erwartet hätte, dass sie eher in einem separaten Cluster aufgeführt würden. Ein möglicher Erklärungsansatz dazu findet sich in Abschnitt 4.2.1.3. Eine weitere besondere Beobachtung betrifft die Diskrepanz zwischen den Wort-, Titel- und Satzzusammenfassungen die im letzten Abschnitt angesprochen werden.

4.2.1.1 *Hohe Anzahl der Dimensionen*

Zu Beginn von Abschnitt 4.1 wurde bereits kurz auf die ungewöhnlich große Anzahl von Dimensionen der Feature-Vektoren in den verschiedenen Datenquellen eingegangen. Normalerweise sollten sich Wörter in einem Text-Corpus entsprechend Zipf's Law verhalten. Zipf's Law besagt, dass es sehr wenige Wörter gibt, die sehr häufig verwendet werden und sehr viele Wörter, die sehr selten verwendet werden. Die Gesamtzahl verschiedener Wörter, würde mit der Länge des Corpus nur logarithmisch wachsen[28]. Diese Aussage lässt sich an verschiedenen Sprachen und Corpora überprüfen. Der naheliegendste Grund für unsere hohen Zahlen sind OCR-Fehler bei den Journals und Rechtschreibfehler bei den Mail- und Newsgroup-Corpora. Diese kann man als in etwa gleichverteilt abschätzen, was bedeutet, dass die Anzahl der Worte mit Rechtschreib- oder besonders OCR-Fehlern viel schneller (vermutlich annähernd linear) mit der Größe des Corpus wächst. Ein weiterer Grund könnten Eigennamen und Abkürzungen sein, die in ebenso größerer Häufigkeit vorkommen. Die verschiedenen Möglichkeiten sollen hier nochmal detaillierter betrachtet werden.

OCR- UND RECHTSCHREIBFEHLER Fehler, die durch die Texterkennung beim Einscannen von Dokumenten entstehen und Rechtschreibfehler führen dazu, dass zwei Terms nicht als gleich erkannt werden können. Tippfehler führen zu dem gleichen Problem und werden der Einfachheit halber hier zu den Rechtschreibfehlern gezählt, da das Resultat sehr stark „gewöhnlichen" Rechtschreibfehlern gleicht. Die Anzahl von falsch geschriebenen Wörter in den einzelnen Datenquellen anzugeben ist ohne eine aufwendige statistische Analyse kaum möglich, weshalb hier nur grobe Abschätzungen gegeben werden können. Der „Computers & Security"-Corpus weist mit einer Dimension von knapp 20.000 unterschiedlichen, gestemmten Wörtern eine große Zahl von Wörtern auf. Auch wenn 20.000 Wörter für Text-Corpora nicht ungewöhnlich sind, so kann diese Anzahl dennoch stark verringert werden, da sich noch immer viele Rechtschreibfehler im Corpus befinden. Normalerweise würde man erwarten, dass die Begriffe *computer*, *computes*, *computation* und *computing* alle auf den Stamm *comput* reduziert werden. Im globalen Dictionary befinden sich allerdings auch viele ähnliche Begriffe, die sehr wahrscheinlich durch OCR-Fehler entstanden sind. So zum Beispiel *compuf*, *compul*, *computa* und *computer*. Hier wurden Buchstaben in den Wörtern falsch erkannt, so dass der Stemming-Algorithmus zu einem anderen Ergebnis als vorgesehen kam. Diese Art von Fehler lässt sich für viele Wörter entdecken. Solche

28 Vgl. z. B. Gelbukh u. Sidorov (2001).

anderen Beispiele wären *portmap* und *portmapp*, *technolog*, *technolob* und *technolobv*, sowie *authent*, *authenti* und *authen*.

Solche Fehler befinden sich in allen Datenquellen, wobei diese in den Internetquellen seltener vorkommen. Ein Grund hierfür könnte sein, dass Rechtschreibfehler statistisch weniger häufig vorkommen als OCR-Fehler. Diese Annahme und der Fakt, dass die Beiträge dieser Quellen im allgemeinen kürzer sind, als die Veröffentlichungen der Journals erhöht die Wahrscheinlichkeit, dass ein falsch geschriebenes Wort in einem Beitrag so selten vorkommt, dass es am Word-Threshold in der Vorverarbeitung verworfen wird[29].

Eine andere Art der OCR-Fehler ist das Zusammenfügen von Wörtern durch das fehlende Erkennen von Leer- oder Trennzeichen, dieses Phänomen tritt besonders bei IEEE sehr stark auf. So findet man dort den Begriff *accesspermiss* in 42 verschiedenen Dokumenten und *issuecommand* in 22 Dokumenten. Durch die Datenreduktion bei der Vorverarbeitung konnte zwar ein Großteil gefunden werden, doch befindet sich noch immer ein erstaunlich großer Teil an verketteten Wörtern im Corpus.

Trotz Vorverarbeitungsmaßnahmen sind also noch immer falsch geschriebene Wörter im Corpus enthalten, welche die Anzahl der Dimensionen unnötig erhöhen.

EIGENNAMEN UND ABKÜRZUNGEN Städte-, Länder- oder Firmennamen geben gewöhnlich sehr gut wieder, worüber in einer Textquelle gesprochen wurden. Man möchte diese Art von Namen also möglichst in einem Text erhalten. Aber sowohl die internetbasierten Quellen, als auch die Journals beinhalten enorm viele Eigennamen. Bei den journalbasierten Daten spielen vor allem Quellenangaben eine große Rolle. Je nach Zitierstil treten die Namen der Autoren häufiger oder seltener in den Dokumente auf und können deshalb durch den TFIDF hoch bewertet werden. Auf jeden Fall aber erzeugen sie viele neue Wörter, welche die Dimensionen erhöhen. Ob Quellenangaben von der Vorverarbeitung verworfen werden sollten oder nicht, ist eine schwierige Frage. Auf der einen Seite erhöhen die Titelangaben der zitierten Veröffentlichung fachspezifische Wörter, die mit dem Inhalt des betrachteten Dokuments assoziiert werden können, auf der anderen Seite werden durch die Namen der Autoren viele neue Wörter erzeugt, die nichts über den originären Inhalt der Arbeit aussagen. Ein weiteres Problem ist, dass ein gleicher Name nicht bedeutet, dass dieselbe Person gemeint ist und selbst wenn dies der Fall sein sollte, ist so nicht sichergestellt, dass ein Autor auch nur über ein Thema schreibt. Besonders wenn es darum geht Unterthemen einer Forschungsrichtung zu identifizieren, kann damit gerechnet werden, dass sich gleiche Personen über unterschiedliche Themen äußern.

Es ergibt sich das Problem, dass das Ähnlichkeitsmaß zweier Feature-Vektoren durch gemeinsame Namen von Autoren kleiner ausfällt als es eigentlich der Fall sein müsste. Im Vorverarbeitungsschritt dieser Arbeit wurden Quellenangaben nicht aus den Dokumenten entfernt, der teilweise hohe Anteil von Eigennamen scheint aber dafür zu sprechen diese tatsächlich zu entfernen. Letztendlich kann nur ein Versuch zur Klärung des Effekts auf die Cluster-Ergebnisse Gewissheit bringen.

Eigennamen von Autoren sind allerdings nicht die einzigen Bezeichnungen die problematisch sind. Das gleiche Problem ergibt sich auch

29 Details zur Datenreduktion bei der Vorverarbeitung finden sich in 3.3.4.6.

in einigen anderen Fällen. Besonders zwiespältig sind auch die Benennungen von Funktionen, Variablen und Datentypen in Pseudocode, der in Veröffentlichungen oder noch mehr in internetbasierten Quellen auftreten kann, zu beachten. Häufig werden diese Namen sehr kurz und prägnant gewählt oder bestehen nur aus wenigen Buchstaben. In allen Fällen aber spiegelt die Variablen- und Funktionsbezeichnung nur sehr unzuverlässig wieder, worum in einem Text gesprochen wird[30]. Diese Wörter können außerdem auch in thematisch sehr unterschiedlichen Themen auftreten, was gerade für Bezeichner für Datentypen problematisch ist. Das gleiche gilt für unbekannte oder nur selten benutzte Abkürzungen. Begriffe wie PGP (Pretty Good Privacy) sind in der Computerwelt weit verbreitet, Abkürzungen, die aber unbekannt sind und nur in sehr wenigen Veröffentlichungen gebraucht werden, können dazu beitragen die Qualität des Dictionaries zu verschlechtern. Da Abkürzungen gewöhnlich nur aus wenigen Buchstaben bestehen, besteht die Gefahr, dass zwei verschiedene Autoren mit der gleichen Abkürzung Verschiedenes meinen. Desweiteren erschweren unbekannte oder sehr spezifisch auf eine kleine Gruppe von Problemen bezogenen Abkürzungen die Auswertung durch Text-Zusammenfassung, da ihre ursprüngliche Bedeutung in vielen Fällen nicht mehr hergeleitet werden kann. So finden sich in den Wörtern der verschiedenen Corpora alle möglichen nicht lexikalischen Wörter wieder, denen nur schwer eine Bedeutung zugeordnet werden kann, welche aber die Anzahl der Dimensionen erhöhen. Beispiele dafür aus dem „Computers & Security"-Corpus sind *aaf*, *def*[31], *rec*, *zsi* und *fju*.

4.2.1.2 *Schlechte Cluster-Qualität*

Die verschiedenen Themen, über denen in den Datenquellen gesprochen wurde, konnten nur schwer mit der vorgestellten Methode extrahiert werden. So fielen die unterschiedlichen Cluster bei steigendem Radius *Eps* schnell zu einem großen Cluster zusammen[32]. Möchte man dies verhindern, entsteht eine Vielzahl von kleinen Clustern und ein großer Anteil von Noise-Points, was wiederum die Aussagekraft des Clusterings stark beeinträchtigt. Gerade das Clustering für den IEEE-Corpus macht deutlich, dass verschiedene Themengebiete häufig fälschlicherweise in einem Cluster gruppiert werden. Aber auch bei anderen Datenquellen tritt dieses Verhalten auf, wie zum Beispiel in Cluster 11 des Security-Basics-Corpus, in dem die Themen „Passsword" und „Access Controll in Networks" verschmelzen.

Es gibt mehrere mögliche Gründe, die für dieses Verhalten sprechen. Zum einen scheinen die Abstände zwischen den verschiedenen Dokumenten im Feature-Raum keine große Varianz aufzuweisen und auf der anderen Seite scheinen sich Cluster zu überlappen, was zu ihrer Verschmelzung führt. Beide Gründe werden im Folgenden genauer betrachtet und analysiert.

UNIFORME DICHTE Um einen Hinweis zu bekommen, warum dieses Verhalten bei allen Datenquellen auftritt, muss ein Blick auf die Verteilung der Feature-Vektoren im Feature-Raum geworfen werden. Die

30 Ein berühmtes Beispiel ist das Wort *foobar*, welches schon häufig als Bezeichner für Codebeispiele herhalten musste, oder die üblichen Schlüsselwörter von Sprachen wie *while*, *def*, *for*.

31 Das Schlüsselwort *def* wird in Programmiersprachen wie Python oder Ruby, sowie in Pseudocode verwendet um eine Definition von Variablen oder Funktionen einzuleiten.

32 Wie in Abschnitt 3.3.5 geschildert.

k-Dist-Graphen können dazu einen hilfreichen Hinweis liefern. Jeder k-Dist-Graph liefert eine sortierte Liste der Entfernungen zum k-nächsten Nachbarn und damit einen Hinweis auf die Anzahl von Abständen einer bestimmten Länge zwischen den Punkten[33]. Wenn die Abstände zwischen den Punkten im Feature-Raum sehr uniform sind, dann ist zu erwarten, dass auch viele Punkte dieselbe Entfernung zu ihrem k-nächsten Nachbarn haben. Treten hingegen starke Dichteunterschiede auf, dann ist zu erwarten, dass sich dies auch im k-Dist-Graphen zeigt, da es nun Punkte mit entweder sehr großem Abstand zum k-nächsten Nachbarn haben (Noise-Points) oder Punkte mit sehr geringem Abstand zu ihren k-nächsten Nachbarn (Core-/Border-Points).

Die k-Dist-Graphen für Computers & Security und TISSEC zeigen deutlich einen relativ linearen Abfall, was es sehr schwer machte den genauen Wert für die *Eps*-Radien der Clusterings zu bestimmen. Dieser lineare Abfall zeigt, dass es keine genau entscheidbare Grenze zwischen Punkten mit entweder großem oder kleinem Abstand zu ihren k-nächsten Nachbarn gibt. Vielmehr gibt es für jede mögliche Entfernung etwa gleichviele Punkte, so dass der Verlauf des k-Dist-Graphen nahezu linear ist. Dies zeigt, dass die Punkte im Feature-Raum keine klar abgegrenzten Cluster mit gleicher Dichte bilden, was wiederum dazu führt, dass einzelne Themen nicht klar von einem Cluster zusammengefasst werden konnten, sondern auch Teile von anderen Clustern beinhalteten, wie es insbesondere beim IEEE-Corpus sehr deutlich zu beobachten war.

Die gleichmäßige Verteilung unterschiedlicher Dichteverhältnisse innerhalb des Feature-Raums lassen darauf schließen, dass es möglicherweise Cluster unterschiedlicher Dichte gibt. DBSCAN betrachtet aber die Dichte zwischen Punkten eines Clusters immer mit einem konstanten Radius. Eine Lösung wäre die Verwendung von SNN-Density als Maß für die Dichte[34].

ÜBERLAPPUNG VON CLUSTERN Die Wortzusammenfassungen der Cluster zeigen, dass es viele Begriffe gibt, die sehr häufig in Clustern vorkommen, eigentlich jedoch unterschiedliche Themengebiete umfassen. Ein Beispiel dafür wären die Wörter *admin*, *password*, *account*, *access* und *user*. Diese Wörter würden sowohl in einer Textquelle stark bewertet werden, die sich um das Knacken von Passwörtern und Systemsicherheit dreht, als auch in einer Quelle über die beste Wahl einer firmeneigenen Password Policy. Zwar würden andere Begriffe wie *management*, *policy* oder *company* die zweite Quelle durchaus von der ersten unterscheidbar machen, allerdings sinkt der Abstand der Dokumente auf Grund gleicher Wörter erheblich. Ziel sollte es aber sein, beide Themengebiete möglichst strikt voneinander zu trennen, da es sich im ersten Text um ein sehr technisches Thema handelt und die zweite Quelle stark Management-orientiert ist.

Dokumente, welche Wörter beinhalten, die auch in einem anderen Kontext auftreten können, befinden sich also im Feature-Raum in der Nähe zweier unterschiedlicher Cluster-Zentren. Übersteigt die Anzahl solcher Dokumente den gewählten MinPts-Wert, so ist es sehr wahrscheinlich, dass die beiden Cluster über gerade solche Dokumente verbunden werden und damit zusammenfallen.

33 In Abschnitt 3.2.3.2 wurde genauer beschrieben, wie ein k-Dist-Graph definiert ist.
34 In Abschnitt 5.3 gehen wir auf diesen Weg noch etwas ein.

Eine weitere Gefahr für die Cluster-Qualität können Cluster darstellen, die Newsletter oder Kolumnen in den Datenquellen beinhalten. Diese Art von Clustern wurden verstärkt in den Mailinglisten gefunden, so wurden zum Beispiel im Infosec-News-Corpus bei einem Eps-Radius von 0.51 vier verschiedene Cluster gefunden, welche spezielle Newsletter beinhalten[35]. Da Newsletter über verschiedene Themen berichten, sind in diesen Clustern auch Dokumente enthalten, die wiederum mit den Dokumenten anderer Cluster über ihren eigentlichen Inhalt nahestehen können. Enthält beispielsweise ein Newsletter Berichte über die Netzwerksicherheit von Linuxsystemen, so könnten diese Berichte Dokumente zum Themengebiet über Netzwerksicherheit für Unixsysteme sehr nahestehen und DBSCAN würde bei entsprechender Parameterwahl die Beiträge des Newsletters als auch die Berichte über Netzwerksicherheit in einem Cluster zusammenfügen. Die Folge wäre, dass unnötig große Cluster entstehen.

DIMENSIONSANZAHL Offensichtlich führt die oben schon erörterte hohe Anzahl von Dimensionen im Feature-Raum dazu, dass die Methode keine guten Ergebnisse erzielen kann. Zu beobachten ist, dass die Ergebnisse am kleine TISSEC-Corpus mit weit unter 20.000 Dimensionen noch thematisch sehr klare Cluster umreißt, während das Clustering immer weiter verwischt, je mehr Dimensionen der Corpus besitzt und am IEEE-Cluster völlig versagt. Ebenso muss man mit steigender Zahl an Dimensionen auch immer mehr Noise-Points in Kauf nehmen. Das Verwischen der Cluster kann zum einen am DBSCAN Algorithmus liegen, der mehrere Cluster verschiedener Dichte in einem größeren Cluster subsumiert. Zum anderen kann es aber auch an den TFIDF-Zusammenfassungen liegen, die bei größeren Clustern evtl. nicht mehr in der Lage sind passende Zusammenfassungen zu liefern. Nach Aussagen der Literatur bzgl. des Problems der hohen Dimensionalität kann auch beides zutreffen.

4.2.1.3 *Das Watermark-Phänomen*

Im IEEE-Cluster treten in unterschiedlichen Clustern immer wieder die Begriffe *watermark*, *image*, sowie *sensor* und *node* als hochbewertete Begriffe in den Wortzusammenfassungen auf. Es wäre aber zu erwarten gewesen, dass diese Wörter jeweils einen separaten Cluster bilden. Es gibt verschiedene Erklärungen, wie dieses Phänomen entstanden sein könnte.

Eine Möglichkeit wäre, dass die Dokumente tatsächlich mit einem elektronischen Wasserzeichen versehen sind. Schaut man sich aber die Veröffentlichungen an, die das Wort *watermark* beinhalten, so bezieht sich ein Großteil auch in ihrem Titel auf diesen Begriff. Desweiteren führen bei weitem nicht alle Dokumente diesen Begriff in ihrem Text-Corpus auf. Lediglich 1417 Dokumente beinhalten das Wort *watermark* in ihrem lokalen Dictionary. Es scheint also keinen derartigen Wasserzeichen-Text in den Dokumenten zu geben.

Es könnte auch ein einfacher Softwarefehler dafür verantwortlich sein. Die Verarbeitung der lokalen Dictionaries zur Feature-Matrizen wurde aus Geschwindigkeitsgründen nur für IEEE in eine gesonderte Programmbibliothek ausgelagert. Dennoch wurde bei der Vorverarbeitung der Textdaten aus dem IEEE-Corpus, sowie zum Clustern und

35 Siehe Abschnitt 4.1.5 für mehr Details.

Zusammenfassen der Daten der gleiche Programmcode verwendet, wie bei den anderen Datenquellen. Wenn also tatsächlich ein Softwarefehler für dieses Verhalten verantwortlich sein sollte, dann kann er sich nur in dem für IEEE neu geschriebenen Bereich befinden, da ein ähnliches Fehlverhalten nicht in den anderen Cluster-Ergebnissen zu beobachten ist. Allerdings ist uns nicht ersichtlich, wie ein Softwarefehler für diese Art von scheinbarer Fehlkategorisierung verantwortlich sein kann.

Wie im vorherigen Kapitel erwähnt es ist sehr schwer Aussagen über den thematischen Schwerpunkt eines Clusters innerhalb des IEEE-Corpus zu machen. Die Ursache ist eine Vermischung von Dokumenten verschiedener Themen in den Clustern. Viele Cluster beinhalten damit also auch Dokumente zum dem Thema „Watermarks“ und „Sensor Networks“[36].

Wenn in Dokumente zum Thema „Watermarks“ die Begriffe *watermark* und *image* häufig vorkommen und in anderen Veröffentlichungen nicht genannt oder zumindest nur sehr selten angesprochen werden, dann werden diese Begriffe mit einem sehr hohen TFIDF-Wert belegt. Dies wiederum erhöht die Wahrscheinlichkeit in der Zusammenfassung der Wörter für die Cluster sehr weit oben zu erscheinen. Der Cluster 0 mit einer Größe von 905 Dokumenten beinhaltet allerdings nur fünf Veröffentlichungen, die den Begriff *watermark* enthalten. In allen Veröffentlichungen dieser Art kommt der Begriff allerdings deutlich häufiger als hundert mal vor, was für die gesamte Cluster-Bewertung den Begriff sehr stark gewichtet. Trotzdem deutet dies darauf hin, dass die gewählte TFIDF-Zusammenfassungs-Methode für spezielle Situationen ungeeignete Zusammenfassungen erzeugt. Von allen Erklärungen scheint diese die vielversprechendste zu sein, auch wenn zur endgültigen Beantwortung der Frage nach der Ursache des Watermark-Phänomens eine umfassende statistische Analyse notwendig sein wird.

4.2.1.4 *Diskrepanz der Zusammenfassungen*

In einigen Ergebnissen der automatischen Zusammenfassung ist eine Diskrepanz zwischen den Wort-, Titel- und Satzzusammenfassungen zu erkennen. Ein typisches Beispiel dafür ist der Cluster 4 des comp.sec.misc-Corpus dessen Beschreibung im Abschnitt 4.1.6 zu finden ist. Dort wird der Begriff *password* zwar sehr hoch bewertet, es folgen dann aber viele Begriffe aus der Kryptographie, wie etwa *encrypt*, *pgp* oder *secur*. Das Thema des Clusters wurde dementsprechend „Passwords & Cryptography“ bezeichnet.

In der Titelzusammenfassung tauchen vermehrt Titel auf, die sich auf Beiträge zum Thema Passwörter beziehen, während die Satzzusammenfassung zusätzlich das Thema Kryptographie hervorhebt. Solche Diskrepanzen lassen sich auch in weiteren Clustern anderer Corpora beobachten. So z. B. in Cluster 43 des „Computers & Security“-Corpus. Hier steht der Begriff *wireless* an zweiter Stelle der Wort-Zusammenfassung, während an dritter Stellen dann das charakteristische Wort *audit* folgt. In den Titel-Zusammenfassungen kommt danach hauptsächlich das Thema Security-Audits zum Vorschein, während in der Satz-Zusammenfassung eher Wireless-Netzwerke thematisiert werden.

Die Titel-Zusammenfassungen richten sich lediglich nach der Häufigkeit des Vorkommen der Wörter aus der Wort-Zusammenfassung

36 Die Begriffe *sensor* und *node* könnten in Dokumente zu diesem Thema häufig vorkommen.

in dem betrachteten Dokument, nicht aber nach seiner konkreten Gewichtung durch seinen TFIDF-Wert. Bei der Satz-Zusammenfassung hingegen gilt das Wort mit dem höchsten TFIDF als ausschlaggebend für seine Auswahl. Es ist also nicht verwunderlich, dass Wort- und Titel-Zusammenfassungen eher konsistent sind, während sich Satz-Zusammenfassungen etwas davon abheben können.

4.2.2 *Bewertung der Methode*

Ein Ziel der Arbeit ist es eine Methode zu entwickeln mit der wissenschaftlicher Diskurs quantitativ untersucht werden kann. Dazu wurden eine Methode entwickelt, welche Dokumente einer beliebigen Textquelle in einzelne Wörter zerlegt und diese als einen numerischen Vektor innerhalb eines mehrdimensionales Raums darstellt. Dessen Dimensionen setzten sich aus den Wörtern der Textquelle zusammen. Vektoren mit ähnlichen Werten werden durch einen Cluster-Algorithmus zu Kategorien zusammengefasst, um so Mengen von thematisch ähnlichen Dokumenten zu erhalten. Die Analyse der Cluster erfolgt mit Hilfe eines einfachen Verfahrens zur automatischen Zusammenfassung.

In diesem Abschnitt wird untersucht wie gut die vorgestellte Methode zur Untersuchung von wissenschaftlichen Diskurs in Textdaten geeignet ist. Dabei werden auf die Stärken und Schwächen der einzelnen Verarbeitungsschritte eingegangen und abschließend ein gesamtes Fazit gezogen. Eine exakte Evaluierung der Methode mit statistischen Massen konnte im Rahmen dieser Arbeit noch nicht geleistet werden und steht noch aus[37].

4.2.2.1 *Datenaufbereitung*

In der Datenaufbereitung werden die Datenquellen in eine konsistente Form gebracht. Dazu wurden zuerst Störmerkmale entfernt (nur E-Mails und Newsgroups), dann Duplikate gelöscht (nur IEEE und Computers & Security) und schließlich die Inhalte zusammengefasst (nur E-Mails und Newsgroups).

Die Zusammenfassung der Inhalte kann als sinnvoll bewertet werden, da nur so überhaupt eine gewisse Menge von Inhalten in den internetbasierten Beiträgen zu finden ist. Ebenso ist die Entfernung der perfekten Duplikate erfolgreich verlaufen. Stichprobenhafte Kontrollen haben gezeigt, dass keine tatsächlich unterschiedlichen Dokumente entfernt wurden. Einzig die Entfernung der Störmerkmal ist noch mangelhaft. So wurde Cluster 6 des „Computers & Security“-Corpus zu einem großen Teil anhand der Signatur eines Autors geclustert. Dessen Signatur war nicht wie üblich durch „- - “ vom Textkörper getrennt und hat durch die ständig wiederkehrenden gleichen und gleichzeitig markanten Wörter zu hoher Ähnlichkeit geführt. Ebenso wurden im InfosecNews und comp.sec.misc-Corpus viele Newsletter geclustert. Dies zeigt, dass noch mehr strukturelle Merkmale entfernt werden müssen. Ähnliche Effekte zeigen sich im „Computers & Security“-Corpus für Kolumnen („From The Editor“ und „Random Bits & Bytes“).

37 Kapitel 5 beschäftigt sich mit möglichen Erweiterungen um die Performance und Qualität der Methode zu verbessern.

4.2.2.2 *Vorverarbeitung*

Während der Vorverarbeitung werden die Textdaten aus den gegebenen Quellen eingelesen und in numerische, maschinenlesbare Daten umgewandelt. Bei diesem Schritt ist es wichtig, dass die numerischen Werte, also die Feature-Vektoren, möglichst gut die Originaldaten abbilden und dabei möglichst wenig Dimensionen besitzen. Schaut man sich die am höchsten bewerteten Wörter innerhalb der generierten Feature-Vektoren an, lässt sich durchaus auf den thematischen Inhalt der Originaldaten schließen. So sind zum Beispiel die fünf Begriffe welche die Veröffentlichung *Design and evaluation of a high-performance ATM firewall switch and its applications* aus dem IEEE-Corpus am besten beschreiben *cell*, *firewal* (firewall), *packet*, *switch* und *cach* (cache). Diese Wörter geben sehr gut die Zugehörigkeit der Arbeit zu dem Themenbereich „Low Level Network Technology“ wieder. Zwei dieser Wörter werden sogar im Titel der Arbeit genannt.

Ein anderes Beispiel aus dem selben Text-Corpus sind die Begriffe *compani* (company), *stateg* (strategy, strategic), *internet*, *plan* und *questionair*, welche eine Veröffentlichung unter dem Titel *Management problems of Internet systems development* zusammenfassen. Diese Arbeit wertet eine Onlineumfrage aus, um spezifische Managementprobleme von Internetsystemen ausfindig zu machen. Auch hier bieten die extrahierten Wörter eine gute Vorstellung von dem eigentlichen Inhalt. Die Tokenization und Gewichtung der Wörter durch TFIDF scheinen also aussagekräftige Feature-Vektoren zu genieren.

Die verwendete Art der Vorverarbeitung ist weiterhin in der Lage Textdaten aus verschiedenen Quellen wie wissenschaftlichen Journals und Beiträgen aus Mailinglisten bzw. Newsgroups sinnvoll zu verarbeiten und nicht auf eine spezifische Textform beschränkt.

Problematisch allerdings ist die bereits beschriebene hohe Anzahl von Dimensionen der Wörter[38]. Diesem Problem wurde durch das Entfernen von Wörtern begegnet, welche nur sehr selten in den betrachteten Text-Corpus vorkommen oder eine ungewöhnlich Länge aufweisen[39]. Diese Vorgehensweise reduziert die Anzahl von Dimensionen je nach Quelle um bis zu 70-80% und ist dabei sehr laufzeiteffizient. Dennoch liegt die Anzahl der Dimensionen für viele Datenquellen noch immer zu hoch und muss weiter reduziert werden, möchte man die Qualität der Feature-Vektoren erhöhen.

FAZIT Die Datenvorverarbeitung hat sich insgesamt als sehr effizient erwiesen und liefert gute Ergebnisse. Die Feature-Vektoren stellen die originalen Textdaten erkenntlich da. Es wurden außerdem sehr laufzeiteffiziente Wege gefunden um einen Großteil von störenden Wörtern zu entfernen, dennoch ist die Anzahl der Dimensionen für einige Datenquellen noch zu groß, was das anschließende Verwenden von maschinellen Lernverfahren auf diesen Daten erschwert und die Qualität der Ergebnisse negativ beeinflusst.

4.2.2.3 *Clustering*

Das Clustering fasst die als numerische Vektoren dargestellten Textdokumente in verschiedene vorher nicht spezifizierte Kategorien zusammen. Dabei wurde der DBSCAN-Algorithmus verwendet, wobei die

38 Siehe dazu Abschnitt 4.2.1.1.
39 Wie in Abschnitt 3.3.4.6 beschrieben.

Ähnlichkeit der Feature-Vektoren durch deren Cosinusdistanz bestimmt wurde.

Die Verwendung von DBSCAN hat den Vorteil besonders laufzeiteffizient zu sein. Das Clustering war so auch für große Datenquellen mit vielen Dokumenten wie dem IEEE-Corpus möglich, was sich bei der Parameterwahl auszahlte. Die Parameter wurden zuerst versucht durch k-Dist-Graphen zu finden, aber auf Grund der uniformen Dichte hat sich dies als nicht praktikabel herausgestellt.

Der alternative Ansatz war den Cluster-Vorgang mit verschiedenen Parametern und deren anschließenden Bewertung anhand der Noise-Points, Gesamtanzahl der Cluster und Standardabweichung der Cluster-Größen mehrfach durchzuführen. Aus den Ergebnissen wurde dann ein Parametersatz gewählt. Dieses Verfahren zum Bestimmen der Parameter erleichtert das Finden der Parameter erheblich und liefert zusätzliche Erkenntnisse über die Eigenschaften des Feature-Raums[40]. Auf der anderen Seite erfordert diese Lösung eine sehr effiziente Implementierung von DBSCAN in Bezug auf die Laufzeit, damit der Cluster-Vorgang für geeignet viele Parameter wiederholt werden kann.

Der Cluster-Algorithmus findet Cluster zusammengehöriger Dokumente für die untersuchten Datenquellen, hat aber Probleme verschiedene Themengebiete gut voneinander zu separieren. So ist häufig zu beobachten, dass thematisch verschiedene Dokumente, die man in einem jeweils eigenen Cluster erwarten würde, miteinander verschmelzen. Dieses Problem konnte nur durch die Auswahl besonders strikter Parameter begegnet werden, was dazu führte, dass lediglich Zentren hoher Dichte im Feature-Raum zusammengefasst wurden. Das Ergebnis sind eine Vielzahl kleiner Cluster, sowie eine große Anzahl von Noise-Points. Dennoch wurden einige ausdrucksstarke Cluster von Dokumenten gefunden.

So fiel die Zuordnung von Themen für die Cluster des TISSEC-Corpus sehr einfach. Hier konnten auch dank der geringen Anzahl von Dokumenten pro Cluster die Titel der jeweiligen Veröffentlichungen komplett in die Cluster-Zusammenfassung aufgenommen und evaluiert werden. Ein Grund für die guten Ergebnisse bei TISSEC könnte an der geringen Größe der Feature-Vektoren liegen. Für die Cluster aus dem Journal Computers & Security und der Infosec-News Mailingliste konnten ebenfalls relativ eindeutige Themen zugewiesen werden. Bei den anderen Datenquellen hingegen war immer wieder zu beobachten, dass verschiedene Themen in einem Cluster verschmolzen.

FAZIT Der Einsatz von DBSCAN mit Verwendung der Cosinusdistanz erzielt ausreichend genaue Ergebnisse für Corpora mit niedriger Anzahl von Dimensionen, sofern man bereit ist viele Noise-Points in kauf zu nehmen. So werden viele Cluster gefunden deren thematischer Bezug sich eindeutig bestimmen lässt. Die Qualität des Clusterings sinkt allerdings mit steigender Anzahl von Dimensionen der Datenquellen. Dennoch kann eine Aussage über den Verlauf eines wissenschaftlichen Diskurses in Form der Feststellung von Trends gemacht werden. Unter Betrachtung der hohen Anzahl von Noise-Points ist es allerdings nicht möglich genauere Entwicklungen aus den Cluster-Ergebnissen abzulesen ohne deren Aussagekraft stark einschränken zu müssen.

40 z. B. wurde so das frühe Zusammenfallen der Cluster sehr deutlich gezeigt.

4.2.2.4 *Cluster-Zusammenfassung*

Bei der Zusammenfassung der Cluster wurden durch den TFIDF hochbewertete Wörter aus den Dokumenten der Cluster extrahiert und auf deren Basis die Titel bedeutender Dokumente ausgewählt. Zusätzlich wurden anhand der TFIDF-Werte jedes Dokuments die charakteristischsten Sätze der Cluster ermittelt.

Aus der Wortliste lässt sich in den meisten Fällen auf eine generelle thematische Tendenz des Clusters schließen. Die Titel-Zusammenfassung konkretisiert das Thema gut und stellt ein hilfreiches Mittel dar um eine Themenzuweisung vorzunehmen. Die Satz-Zusammenfassung hingegen bietet nur in wenigen Fällen zusätzliche Informationen und betont in einigen Fällen sogar andere thematische Schwerpunkte als die Satz-Zusammenfassung. Es kann ebenso keine Aussage gemacht werden, ob das Verfahren immer die ganze thematische Breite eines Clusters wiedergeben kann. Dies erschwert es bei großen Clustern einzuschätzen, ob noch andere, eher schwächer repräsentierte Themen im Cluster vorhanden sind.

FAZIT Die Wort- und Titel-Zusammenfassungen sind bei der Benennung der Cluster hilfreich und liefern vor allem bei kleinen Clustern eine gute Darstellung des thematischen Schwerpunkts. Problematisch hingegen ist die fehlende Evaluierung des Verfahrens, was wiederum die Validität der Benennung der Cluster beeinträchtigt.

4.2.2.5 *Gesamtbewertung*

Die hier vorgestellte Methode ist in der Lage Textdaten, die aus unterschiedlichen Quellen wie Journals, Mailinglisten und Newsgroups kommen und daher verschiedene Eigenschaften aufweisen, automatisiert zu analysieren und in verschiedene Themenbereiche zu gruppieren. Aus dem Ergebnis des Cluster-Vorgangs lassen sich vor allem für Datensätze mit niedrigen Dimensionen leicht thematische Schwerpunkte für die Cluster benennen. Die einzelnen Verarbeitungschritte sind dabei sehr lauffzeiteffizient. Das gilt sowohl für die Vorverarbeitung als auch für den Cluster-Vorgang. Das schnelle Berechnen von Ergebnissen erlaubt daher auch neben k-Dist-Graphen das Erstellen von Parametergraphen um Parameter für den Algorithmus zu wählen. Desweiteren bietet das Verfahren die Möglichkeit die Anzahl der Dimensionen mit speziellen Reduktionsverfahren auf sehr einfache Weise stark zu reduzieren.

Dennoch scheinen die Maßnahmen zur Dimensionsreduktion nicht auszureichen. Die Schwäche des Verfahrens ist vor allem die schlechte Qualität der Cluster bei hochdimensionalen Daten. Häufig werden Dokumente, die unterschiedliche Themen behandeln, fälschlicherweise in einem Cluster zusammengefasst. Um brauchbare Ergebnisse zu erhalten, muss daher bei der Wahl der Parameter für das Clustern häufig eine hohe Zahl von Noise-Points in Kauf genommen werden. Dies verringert allerdings die Aussagekraft der Schlussfolgerungen, die aus der Analyse der Cluster-Ergebnisse gezogen wird.

4.3 AUSSAGEN

Nachdem wir nun die Schwächen des Verfahrens kennen und abschätzen können, welche Aussagen noch gemacht werden können, kommen wir nun zu genau diesen. Wir werden drei Thesen aufstellen, sie an-

hand der Kurven der Cluster unterstützen und kurze Anmerkungen zur Aussagekraft unserer Argumente machen. Anschließend werden diese Thesen in Zusammenhang mit anderen, bereits formulierten Strukturierungen des Diskurses über Informationssicherheit gesetzt.

4.3.1 *Thesen*

Obwohl wir wissen, dass unsere Beweislage wegen der Schwächen der Methoden und eines fehlenden sinnvollen Clustering der IEEE Daten sehr dünn ist, sind uns beim betrachten der Diagramme (die in Anhang A vollständig wiedergegeben sind) drei mögliche Aussagen aufgefallen, die sich zugleich mit der subjektiven Wahrnehmung als auch den statistischen Ergebnissen decken[41]. Diese Aussagen werden wir mit Hilfe der Zuordnung von Clustern und der Bewertung ihrer Aussagekraft unterstützen. Die Bewertung der Aussagekraft erfolgt dabei anhand dreier Faktoren:

1. Anzahl der Dokumente im Cluster
   Ist ein Cluster, im Verhältnis zur Gesamtzahl der Publikationen im Corpus nur sehr klein, so kann schwerlich eine Aussage getroffen werden. Das argumentative Gewicht ist also direkt proportional zum Anteil des Clusters am gesamten Publikationsvolumen.

2. Deutlichkeit des Themas
   Ein Cluster aus dessen Zusammenfassung hervorgeht, dass in ihm offensichtlich mehrere Themen zusammenfallen, kann nicht voll zur Unterstützung einer Aussage herangezogen werden. Im Gegensatz dazu, ist eine klar abgrenzbare Zusammenfassung ein gutes Zeichen für die interne Konsistenz eines Clusters.

3. Deutlichkeit des Trends
   Eine minimale Schwankung der Kurven ist weniger ausdrucksstark und eindeutig als eine deutliche Auf- bzw. Abbewegung. Je stärker also ein Trend sichtbar ist, umso stärker auch die Aussagekraft der Kurve.

Aus diesen Faktoren schätzen wir auf einer Skala (hoch, mittel, gering, sehr gering) die Aussagekraft einer Cluster-Kurve ab. Mit diesem Bewertungsschema können nun die drei Thesen formuliert und unterstützt werden.

41 Eine mögliche vierte Feststellung, dass nämlich die Verbreitung des Internets und der exponentielle Anstieg der Internetanschlüsse in den späten 90ern einen Themenwandel hin zu Netzwerk- und Web-Themen induziert hat, ist relativ trivial und hat keine direkte Bedeutung für Wandel im Diskurs über Informationssicherheit. Wir überlassen deshalb die Beobachtung und Interpretation dieses Phänomens dem geneigten Leser.

4.3.1.1 *These I*

Diskussionen über *Basistechnologien* nehmen ab.

Wir haben viel überlegt, wie wir denn das, was wir an den Clustern beobachten, in möglichst eindeutigen Worten formulieren können, sind aber zu dem Schluss gekommen, dass wir hier auf ein philosophisches Problem gestoßen sind. Versuchen werden wir es trotzdem:

In fast allen Corpora lässt sich beobachten, dass die Diskussionen zu Technologien niedriger Abstraktionsebene abnehmen. Wir nennen diese Technologien nun einfach *Basistechnologien*. Beispiele solcher Technologien sollen sein:

- die unteren Schichten der Kommunikationsnetze (Vermittlungsschicht, Transportschicht), dazugehörige Technologien wie Firewalls, Router und Tools zur Diagnose von Problemen in diesen Schichten (Nmap, Netcat, ...),
- niedrige Abstraktionsebenen von Betriebsmitteln in Rechnern (Dateien, Prozesse, Sockets, ...),
- elementare Elemente der IT-Sicherheit wie Passwörter, Zugriffsberechtigungen oder teilweise auch Kryptographie, insbesondere in Verbindung mit Netzwerkprotokollen.

Im Gegensatz dazu werden wir in der nächsten These *höhere Technologien* betrachten. Wo nun genau die Grenze zwischen *Basistechnologien* und *höheren Technologien* liegt, können wir nicht genau definieren, werden uns im Folgenden jedoch an einer Erklärung versuchen.

Es handelt sich bei der Entwicklung von *Basistechnologien* zu *höheren Technologien* in jedem Fall um eine Bewegung hin zu immer abstrakteren Prinzipien und Betrachtungsweisen von ähnlichen Vorgängen oder Problemen. Abstraktion ist bekanntlich eines der Grundprinzipien der Informatik, und so verwundert es nicht, dass Abstraktion auch passiert. Im Allgemeinen versteht man darunter das Weglassen von bestimmten Aspekten. Eine nicht-abstrakte Basistechnologie soll also etwas sein, was von wenigen Aspekten abstrahiert. Besonders schön lässt sich dies an der End-to-End Architektur von TCP/IP-Netzwerken beschreiben. Jede Schicht stellt der nächst-höheren Schicht bestimmte Funktionen zur Verfügung. Was genau alles notwendig ist um diese Funktion zu realisieren wird versteckt. So werden in der untersten, physischen Schicht noch Bits und Bytes auf Wellen aufmoduliert, während in der obersten Schicht, der Anwendungsschicht nur noch ein zuverlässiger Kommunikationskanal zwischen zwei Endpunkten betrachtet werden muss. Diese Abstraktion lässt sich weiterführen, so prophezeien einige Stimmen[42], dass es zukünftig nicht mehr die Kanäle sein werden, die uns interessieren, sondern die Inhalte, die von A nach B gelangen.

Ein weiteres Beispiel, an welchem sich das Prinzip der Abstraktion gut beschreiben lässt, sind Programmiersprachen. Wurden frühere Computer noch mit Lochkarten und direkt in Maschinencode programmiert, entwickelten sich die Programmiersprachen weiter, erst zu Assembler-Dialekten, die immer noch wenig von der Maschine abstrahierten, dann zu den ersten sog. höheren Sprachen (C, FORTRAN, COBOL), die zumindest vom Befehlssatz, nicht jedoch vom Aufbau der Maschine, abstrahierten. Diese Entwicklung hält an. So werden heute

42 Siehe dazu z. B. Clark u. a. (2005) oder FCN/FIA (2009).

Sprachen entwickelt, welche sehr stark von der Maschine und deren Aufbau losgelöst arbeiten. Es gelingt jedoch nicht immer auf dieser Entwicklungslinie scharfe Schnitte zu setzen.

Im Bezug auf Informationssicherheit scheint es so, dass sich seit den späten 1990ern, auf jeden Fall jedoch seit dem Beginn des 21. Jahrhundert das Interesse für die oben genannten *Basistechnologien* in einigen Bereichen des Diskurses über Informationssicherheit deutlich verringert. Wir haben dazu in Tabelle 9 alle Cluster aufgelistet, welche wir dem Bereich der *Basistechnologien* zuordnen wollen.

Cluster (Corpus, #ID)	Praxis/ Theorie	Thema	Verlaufsbeschreibung	Aussagekraft	Bemerkungen
comp.sec. misc #4	P	Passwords & Cryptography	abfallend	hoch	
Security Basics #1	P	Low Level Network Security	abfallend	hoch	
Security Basics #11	P	Passwords & Access Control in Networks	abfallend	hoch	
Security Basics #32	P	Ethernet Security	leicht abfallend	mittel	
TISSEC #0	T	Access Control	konstant	mittel	
TISSEC #1	T	Sensor & Wireless Security	leicht ansteigend	mittel	*a*
TISSEC #2	T	Cryptographic Protocols	konstant	sehr gering	*b*
TISSEC #6	T	Linux Security	Peak um 2002-2004	sehr gering	*b*
TISSEC #7	T	SSH-Security	Peak um 2004	sehr gering	*b*
Computers & Security #25	T	Cryptography	Peak um 1989	sehr gering	*b*

a Erklärbar durch Novität.
b Vernachlässigbar wegen geringer Größe.

Tabelle 9: Cluster über Basistechnologien und deren Verläufe, sortiert nach Aussagekraft

Es ist ablesbar, dass sich, bis auf zwei Ausnahmen im Bereich der Forschung, diese These in den Daten bestätigen lässt[43]. Die eine Ausnahme im Bereich der Forschung betrifft den Bereich der Sicherheit von Sensor- und Funknetzen. In dem entsprechenden Cluster ist ein Anstieg der Forschung zu verzeichnen. Dies kann mit der generellen bedeutenden Zunahme und Verfügbarkeit von Funktechnologien auch

43 Mit der gebotenen generellen Vorsicht der Methode gegenüber.

im Consumer-Bereich begründet werden. Die zweite Ausnahme betrifft wiederum die Forschung. In der ACM TISSEC scheint „Access Control" ein ständiges Thema zu sein. Dies kann mit der in Abschnitt 2.2.1.1 beschrieben Ausrichtung des Journals begründet werden[44].

Bemerkenswert ist dabei, dass diejenigen Cluster, welche die These ganz klar unterstützen, alle aus dem Anwenderbereich stammen. In Mailinglisten und Newsgroups kommunizieren hauptsächlich, wie in den Abschnitten 2.2.3 und 2.2.2 beschrieben, private oder berufliche IT-Anwender über ihre Fragen und Probleme. Dass selbst dort das Interesse für die *Basistechnologien* abnimmt, deutet entweder darauf hin, dass ihre Komplexität beherrschbar geworden ist und keine Aufmerksamkeit mehr nötig ist, oder die Aufmerksamkeit zwangsläufig anderen Bereichen gewidmet wird, da sie dort notwendiger geworden ist.

44 Ein Urteil, ob die TISSEC damit ihrer Zeit hinterher hängt, sei dem Leser überlassen.

4.3.1.2 *These II*

Diskussionen über *höhere Technologien* nehmen zu.

Im Gegensatz zu den *Basistechnologien* sollen, wie oben angedeutet, *höhere Technologien* auf einem höheren Abstraktionsniveau angesiedelt sein. Als Paradebeispiel können wiederum die End-to-End Netzwerke herangezogen werden. Dort ist zu beobachten, dass sich immer mehr Funktionalität in die oberen Schichten verschieben. Wurde früher für einen neuen Dienst noch ein neues Anwendungsprotokoll entworfen (z. B. IMAP als Fortschritt gegenüber POP3 für die komfortable E-Mail-Benutzung) und auf einem eigenen TCP/IP Port als Dienst angeboten, so wird heute viel in HTTP als Web-Service oder Web-Anwendung angeboten (z. B. Webmail Dienste). Die Abstraktion besteht dabei durch eine Kapselung der Verbindung in Konzepten wie Transaktionen. Um den Begriff der *höheren Technologien* weiter mit Leben zu füllen, seien hier noch weitere Beispiele angegeben:

- im Bereich der Netzwerke: Web-Services und Web-Anwendungen, Intrusion Detection, Layer-7 Firewalls;
- im Bereich der Kryptographie: Public-Key-Infrastrukturen und Zertifikat-Ketten, komplexere Krypto-Verfahren;
- im Bereich der Systeme: Virtualisierung, Anwendungen und Update-Mechanismen.

Diese *höheren Technologien* unterscheiden sich alle von *Basistechnologien* dadurch, dass sie immer mehr bestimmte Aspekten versteckend. So sind Public-Key-Infrastrukturen z. B. mathematische Abstraktionen der Vertrauensbeziehungen zwischen Akteuren, die dann dazu verwendet werden kryptographische Probleme zu bewältigen. Web-Services stellen, wie schon gesagt, im gleichen Maße eine höhere Abstraktion der Kommunikationskanäle dar; Intrusion-Detection-Systeme analysieren Netzwerkverkehr nicht mehr nach einzelnen Angriffen oder nur ganz bestimmten Paketen, sondern erkennen eine Vielzahl von Mustern und Unregelmäßigkeiten, teilweise sogar in der Anwendungsschicht; wird über Anwendungen und deren Verhalten als Ganzes sowie die Einflüsse von Updates diskutiert, so wird von der konkreten ausführbaren Datei und dem veränderten Code abstrahiert.

In Tabelle 10 sind alle Cluster nach Aussagekraft sortiert aufgelistet, welche wir dem Bereich der *höheren Technologien* zuweisen wollen. Obwohl nicht ganz so deutlich wie für *These I*, lässt sich auch hier in Summe ein Aufwärtstrend erkennen. Klar zu erkennen ist, dass sich in der Zeit um 1995-2000 durch die rasche Verbreitung des Internets eine rege Diskussion um Web-Security Themen etabliert hat. Dieser Trend hat sich zwar nach 2000 wieder deutlich abgekühlt, erfährt aber dann wieder einen leichten Anstieg. Ebenso hat das Thema Microsoft-Security bis ca. 2006 einen starken Zuwachs erfahren. Danach fällt es wieder ab, was möglicherweise mit den verstärkten Sicherheitsbestrebungen von Microsoft zusammenhängen könnte. Das Thema der Public-Key-Infrastrukturen hat in den betrachteten Zeiträumen entweder konstante oder leicht zunehmende Aufmerksamkeit erfahren. Zusammengenommen kann dies alles als Zeichen für die Zunahme der Wichtigkeit von *höheren Technologien* gewertet werden.

Cluster (Corpus, #ID)	Praxis/ Theorie	Thema	Verlaufsbeschreibung	Aussagekraft	Bemerkungen
Computer & Security #1	T	Network & Web Security	Peak um 1995-2000	hoch	*a*
Infosec News #3	P	Microsoft Security	ansteigend bis 2006, dann stark abfallend	mittel	
comp.sec. misc #0	P	Web Security	Peak um 2000, danach leicht ansteigend	mittel	
comp.sec. misc #18	P	Public Key Infrastructures	konstant	mittel	
comp.sec. misc #16	P	Web-Proxy Security	leicht ansteigend bis ca. 2000, danach leicht abfallend	gering	
Infosec News #9	P	Oracle Security	leicht ansteigend	gering	
Security Basics #38	P	Microsoft Security	leicht ansteigend	gering	
TISSEC #3	T	Intrusion Detection	konstant	sehr gering	*b*
TISSEC #5	T	Trust Negotiation	ansteigend	sehr gering	*b*
TISSEC #4	T	XML-Security	konstant	sehr gering	*b*

a Fällt mit der Verbreitung des Internets zusammen.
b Vernachlässigbar wegen geringer Größe

Tabelle 10: Cluster über höhere Technologien und deren Verläufe, sortiert nach Aussagekraft

4.3.1.3 *These III*

Diskussionen über Management-Aspekte der Informationssicherheit nehmen zu.

Wenn wir bei dieser Aussage von *Management-Aspekten* reden, dann wollen wir damit alle Sub-Themen erfassen, die anerkennen, dass zur Lösung des Sicherheitsproblems auch ökonomische Ansätze herangezogen werden können oder müssen. Dazu gehören typischerweise Standards (auf verschiedensten Ebenen) sowie zugehörige Audits und Zertifizierungen, Methoden zur Erfassung (Measurement) der Informationssicherheit, aber auch schon viel einfachere Abwägungen über Delegation, Berechtigungen und Rollenvergabe abseits der reinen technischen Umsetzung. Obwohl eine genauere Unterteilung der *Management-Aspekte* prinzipiell möglich wäre, lässt die Präzision unserer Methode noch keine solche zu. In Tabelle 11 sind alle Cluster zusammengestellt, welche wir den *Management-Aspekten* zuordnen können.

Cluster (Corpus, #ID)	Praxis/ Theorie	Thema	Verlaufsbeschreibung	Aussagekraft	Bemerkungen
Computers & Security #33	T	ITSEC & Security Evaluation	Peak um 1990-1997	mittel	
Computers & Security #43	T	Audits & Wireless Security	leicht ansteigend	mittel	*a*
Security Basics #59	P	Personal Certification	leicht ansteigend	mittel	
Security Basics #6	P	Audits, Compliance & Standards	Peak um 2007-2008	niedrig	*b*

a Ungenaue Themenabgrenzung.
b Craig-Wright-Cluster, siehe Abschnitt 4.1.

Tabelle 11: Cluster über *Management-Aspekte* der Informationssicherheit und deren Verläufe, sortiert nach Aussagekraft

Aus den Daten lässt sich nun ableiten, dass *Management-Aspekte* seit den 1990ern immer mehr an Gewicht gewinnen. Eine Besonderheit bildet dabei der Verlauf der Kurve des Clusters 33 des „Computers & Security"-Corpus. In diesem Cluster geht es um die Evaluierung von Sicherheit, vorrangig nach ITSEC-Standard. Diesen eher technisch orientierten Standard ordnen wir hier trotzdem den *Management-Aspekten* zu, da ein Standard generell ein Mittel ist um möglichen Kommunikationspartnern Sicherheit zu signalisieren. Die Publikationsspitze ab 1990 fällt mit der Verabschiedung des Standards zusammen. Nach 1997 fällt das Publikationsvolumen deutlich ab, was darauf hindeutet, dass das Sub-Thema der Zertifizierung nach technikzentrierten Sicherheitsstandards zumindest im Bereich der Forschung wieder unwichtiger wird. Jedoch finden sich bereits vor und weit nach dieser Spitze Publikationen in

diesem Cluster, was darauf hindeutet, dass auch andere Standards und Zertifizierungen hier mit hinein gezählt haben[45].

Zusammengenommen lässt die unterstützende Aussagekraft in Anbetracht der schwachen und nicht ganz eindeutigen Kurvenverläufe jedoch zu wünschen übrig.

4.3.1.4 *Zusammenfassung*

Die drei von uns aufgestellten Thesen können, wenn zwar auch nur schwach, aber dennoch empirisch unterstützt werden. *These I* (*Basistechnologien* werden unwichtiger) und *These II* (*höhere Technologien* werden wichtiger) hängen dabei offensichtlich zusammen. Sie zeigen, dass die Informationstechnik einer ständigen Entwicklung unterliegt und dem natürlich auch der Diskurs über Informationssicherheit folgt. *These III* (*Management-Aspekte* werden wichtiger) steht dagegen für sich allein und bestätigt die subjektive Wahrnehmung der Veränderungen im Diskurs über Informationssicherheit. Detailliertere Aussagen lassen sich anhand der vorliegenden Ergebnisse noch nicht treffen.

4.3.2 *Relevanz der Thesen*

Nun muss sich der Bogen noch schließen: Welche Bedeutung haben unsere drei Thesen für andere Arbeiten im Bereich der Informationssicherheits-Forschung? Wir wollen uns hier auf die eingangs erwähnten Arbeiten von von Solms, sowie Pallas beziehen und die Bedeutung unserer Ergebnisse für deren Modelle herausarbeiten.

Von Solms (2000) beschreibt drei „Security Waves“, Wellen der Informationssicherheitsforschung, die sich historisch gegenseitig ergänzen. Diese Wellen sind:

1. Technik-Welle
   Die Technik-Welle ist dadurch gekennzeichnet, dass Sicherheitsprobleme nur als technische Probleme aufgefasst wurden. Dementsprechend wurde ausschließlich versucht das Sicherheitsproblem durch die Erforschung und Anwendung von Technologie zu lösen. Die Welle ging einher mit Mainframe-Computern und deren Sicherheitstechniken wie z. B. Passwörtern, Access Control Lists oder Prozess-Separation.

2. Management-Welle
   Die weitere Entwicklung der Informationstechnik führte dazu, dass zunehmend auch die Führungsebene von Organisationen an Informationssicherheit interessiert war. Dies führt zur Einführung von Richtlinien und Verantwortungen im Management für Informationssicherheit. Es kam allerdings nicht zu einer Ablösung, sondern zu einer Ergänzung der technisch realisierten Informationssicherheit durch das Management.

3. Institutionalisierungs-Welle
   In einem weiteren Schritt wurden zu den bereits existierenden Management-Ansätzen explizite und in offiziellen, institutionalisiertem Rahmen verfasste Ansätze hinzugefügt. Solche Ansätze sind z. B. Standards und Zertifizierungen, bewusstes verbessern der Sicherheitskultur und Sicherheitsmetriken zur Bewertung des

45 Siehe dazu auch genauere Ausführungen weiter oben in Abschnitt 4.1.

Stands der Informationssicherheit in einer Organisation. Wiederum findet keine Ablösung sondern Ergänzung statt.

Diese Vorstellung der drei *Security Waves*, insbesondere ihre gegenseitige Ergänzung und nicht Ablösung, lässt sich wie in Abbildung 27 visualisieren.

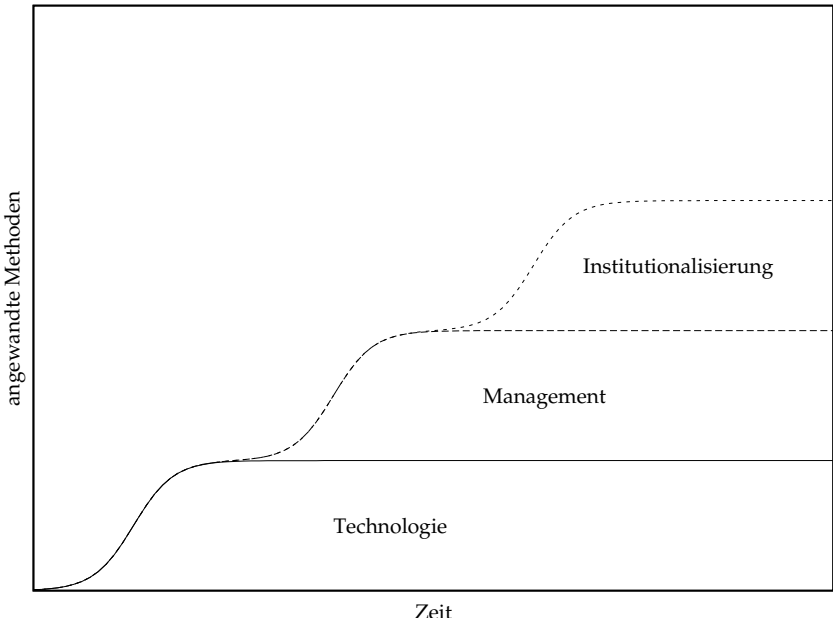


Abbildung 27: *Security Waves* nach von Solms (2000), übernommen von Pallas (2009, S.31)

Von Solms bezieht sich bei seinen Ausführungen dabei scheinbar sowohl auf den wissenschaftlichen Diskurs als auch die praktische Umsetzung[46], nimmt jedoch keine explizite oder scharfe Abgrenzung vor. In einer Weiterentwicklung des Modells[47], fügt er den drei Wellen noch eine vierte Welle hinzu. Diese Welle bezeichnet er als „Governance Wave" und bezieht sich dabei auf die Erkenntnis, dass Informationssicherheits-Management auch als Teil des Business-Continuity Managements betrachtet werden kann. Demnach liegen die Lösungsmittel auch in der wohlgeordneten Führung des Unternehmens und der Einhaltung von relevanten Regulierungen wie z. B. dem Sarbanes-Oxley Act[48][49].

Zwei Aspekte fallen an diesem Modell auf, die nicht ohne weiteres hingenommen werden können und als Fragen formuliert lauten:

1. Bleibt die Wichtigkeit der bereits etablierten *Security Waves* konstant?

2. Ist die Einteilung der *Security Waves* adäquat?

Auf Frage Zwei haben unsere Thesen offensichtlich direktes (wenngleich beschränktes) Antwortpotential: Der Umstand, dass *Management-Aspekte* an Wichtigkeit zunehmen, deutet darauf hin, dass in der Tat

46 Zumindest lesen wir das in der durchgängig allgemein gehaltenen Formulierung „development of information security" (siehe von Solms 2000, S.615).
47 In von Solms (2006).
48 Vgl. von Solms (2006, S.166).
49 Oder damit ebenso dem europäischen Pendant, Basel II.

eine inhaltliche Abgrenzung zwischen einem technischen und einem Management-zentrierten Sicherheitdiskurs gemacht werden kann. Darüber, ob sich dieser Management-zentrierte Sicherheitsdiskurs jedoch in die aufgeführten zwei oder drei zusätzlichen Wellen unterteilen lässt, können wir noch keine Aussage treffen. Ebenso kann über unsere erste Fragen, also den absoluten quantitativen Anteil des technischen Sicherheitsdiskurses am Gesamtvolumen, anhand unserer Ergebnisse vorerst keine Aussage getroffen werden. Die gleichzeitige Unterstützung von *These I* und *These II* durch die Daten lässt die Vermutung zu, dass sich auch dieser Aspekt unterstützen ließe.

Auch Pallas geht auf die Frage ein, ob die Einteilung der *Security Waves* adäquat ist[50] (also vollständig beschreibend und passend abgrenzend ist). Er kommt zu dem Schluss, dass in der Tat noch ein weiterer Abschnitt in der Informationssicherheits-Forschung notwendig ist um das Thema umfassend zu beschreiben. Vor der Verwendung von Mainframes waren Computer üblicherweise isolierte Rechenmaschinen. Der Zugang zu und Zugriff auf diese Rechner konnte gleichzeitig durch ausschließlich physische Schutzvorrichtungen (Türen, Schranken, Zugangskontrollen, ...) garantiert werden. Erst die Einführung des Sharings als neuem Operationsmodus machte Algorithmen und informatische Verfahren (Code) zur Zugriffskontrolle notwendig. Über diese Welle der physischen Sicherheit im Zeitalter der isolierten Computer hätte der IEEE-Corpus Erkenntnisse bringen können. Jedoch kann aufgrund der äußerst geringen Anzahl an geclusterten Dokumenten von vor 1990 und der schlechten Cluster-Qualität noch keine Aussage dazu zustande kommen.

Ebenso stellt Pallas fest, dass sich die Inhalte der Management-Welle und der Institutionalisierungs-Welle anhand ihrer zeitlichen Abfolge nicht klar trennen lassen[51]. Er führt an, dass Standards (insbesondere technische Standards) schon früh, gleichzeitig zur technischen Welle, eingerichtet wurden und Anwendung fanden. Die Vorstellung, dass Institutionalisierung also erst später verwendet wurde, lässt sich so nicht mehr halten. Jedoch zeigt Pallas, dass anstelle dessen die Veränderung der Rechnertechnologie (weg vom Mainframe, hin zum Personal Computer) Veränderungen in der grundlegenden Kostenstruktur des Sicherheitsproblems induzierte und damit auch der Blick auf andere mögliche Gegenmaßnahmen gerichtet wurde. Anhand der Veränderung der Rechner-Paradigmen lässt sich dann eine konsistente Unterteilung der Entwicklung der Informationssicherheit vornehmen. Die Institutionalisierung einer Maßnahme ist dabei immer vorrangig Mittel ihre Kosten zu reduzieren und sie messbar und demonstrierbar zu machen. Institutionalisierung kann sowohl auf der Ebene von technischen Maßnahmen als auch Management-Maßnahmen vorgefunden werden. Sehrwohl hat aber mit der Verbreitung der PCs auch eine Zunahme der Wichtigkeit von Sicherheitsmaßnahmen in Form von formellen Regeln anstelle von technischen Implementierungen stattgefunden. Diese letzte Aussage von Pallas deckt sich mit unserer *These III*.

Genauere Ergebnisse, etwa eine präzise und statistisch überprüfbare Unterteilung des Diskurses über Informationssicherheit, sind mit dem aktuellen Stand unserer Methode noch nicht möglich.

50 Siehe dazu Pallas (2009, Teil 1, Kap. 2).
51 Vgl. Pallas (2009, S.42).

5 IN WEITER FERNE

Nach dem nun der Pflicht-Teil absolviert ist, die Methode gewählt, angewendet und ausgewertet wurde, können wir ein wenig Luft holen, reflektieren und in die Glaskugel schauen, welche Verbesserungen weiterhin möglich und nötig sind.

Wie in Abschnitt 4.2 beschrieben, hat unser Verfahren einige Schwächen. Die größte davon ist der große Anteil an ungeclusterten Dokumenten (Noise-Points). Die Ursachen scheinen in der Struktur der Daten und der Wahl der Feature-Vektoren zu liegen. Zum Teil wurden thematisch zu breite oder nicht thematisch, sondern formal-strukturellen Inhalten folgende Klassen gebildet. In allen Bereichen der Methode, der Datenaufbereitung, der Vorverarbeitung und dem Clustering liegen Verbesserungsmöglichkeiten.

5.1 DATENAUFBEREITUNG

Wie man in den Summaries der Cluster des Infosec-News-Corpus gesehen hat, wurde oftmals Dokumente nicht unbedingt nach ihrer inhaltlichen, sondern eher ihrer formalen Ähnlichkeit angeordnet. So wurden Newsletter des gleichen Absenders regelmäßig einem Cluster zugeordnet. Dies ist dem Umstand zu verschulden, dass formal strukturierende Textelemente wie Überschriften oder „sponsored-by“ Notizen auf genau die gleiche Weise in die weitere Verarbeitung eingehen wie die eigentlichen Inhalte des Dokuments. Am Beispiel des Clusters Nr. 6 im Security-Basics-Corpus lässt auch erkennen, das nicht-abgetrennte Signaturen, in diesem Fall von dem Autor „Craig Wright“, sehr schnell zur Agglomeration vieler Dokumente führen (wobei in diesem Fall sicherlich auch der Inhalt im Bereich der Security Compliance und Standards mitentscheidend war).

Da im späteren Verlauf der Verarbeitung völlig agnostisch gegenüber der Bedeutung und Position der Worte in einem Dokumente gearbeitet wird, lässt sich dort nicht mehr (oder nur über vorher erstelle Negativ- oder Positivlisten) über Bedeutungsgehalt eines Wortes entscheiden. Es bleibt also der Datenaufbereitung überlassen Dokumente ihrer formalen Ähnlichkeiten zu berauben (wie z. B. für Signaturen bereits geschehen). Für diese Anforderung müssen also Filter und kleine Grammatiken gebaut werden, die für häufige Spezialfälle die formalen Elemente entfernen. Dies setzt eine (auch Zeit-) intensive Beschäftigung mit und ständige Neuberechnung der Datenquellen vorraus um iterativ zu einem möglichst vollständigen Satz an Filtern zu gelangen. Ein solches Vorgehen kommt dann einem sehr gezielten Entfernen von irrelevanten Features gleich, ein Prozess, der auch in der Vorverarbeitung stattfinden sollte.

5.2 VORVERARBEITUNG

Die Dimensionalität unser Feature-Vektoren ist enorm hoch. In durchgängig allen Corpora ist die Anzahl der Terms, nach denen differenziert werden kann, höher als die Anzahl der Dokumente, die unterschieden

werden müssen. Dies deutet auf einen Überfluss an (eben teilweise nutzlosen) Informationen hin, mit denen die Algorithmen nicht gut umgehen können. Der „Curse of Dimensionality“ hat uns sicher zu einem Teil erwischt. Die größte Aufgabe bei Verbesserungen der Methoden wird deshalb sein, die wichtigsten Features auszuwählen. Dabei bieten sich einfache Methoden wie das stärkere Abschneiden von seltenen oder besonders häufigen Features ebenso an, wie komplexere Methoden und komplett andere Selektion von Features. Ob sich stärkeres Abschneiden von Features beim Beibehalten der Cosinus-Distanz jedoch tatsächlich gravierend auf die Qualität des Clusterings auswirken, bleibt Versuchen überlassen. Da sowohl TFIDF-Berechnung als auch Cosinus Distanz bereits dazu entworfen sind diese Probleme zu lösen, ist es schwer vorherzusagen, welche Effekte damit noch erreicht werden können. Große Erfolgschancen liegen vermutlich in der Verwendung von festen, manuell gewählten Wörterbüchern als Ausgangsbasis für Feature-Vektoren. In diesen wären dann nur noch nach menschlichen Maßstäben in hohem Maße relevante Features enthalten. Vielleicht könnte man dies dann jedoch wiederum auch als Schwäche, im Sinne des Verlusts der algorithmischen Objektivität, auffassen. Wie eingangs in Abschnitt 3.1 erwähnt, ist es auch möglich die Auswahl der Features als eine Lernaufgabe zu betrachten. Blum u. Langley (1997) bieten eine Übersicht über eine ganze Reihe von Algorithmen zur Feature-Selection. Diese und andere müssten genauer auf ihre Eignung überprüft werden und durch Experimente in die Methoden integriert werden.

Eine völlig andere Richtung ist es nicht einzelne Terms als Features zu verwenden, sondern Multi-Terms, also mehrere zusammengehörige Wörter. Im Abschnitt 3.2 über mögliche Alternativen zur Vorverarbeitung sind wir bereits kurz darauf eingegangen. Hier wollen wir jedoch noch einmal kurz auf vielversprechende konkrete Arbeiten in dem Bereich der Multi-Term Feature-Selection verweisen[1]. So lernt z. B. der KEA-Algorithmus[2] aus ein einem Traingsset ein Model wie aus Texten Keyphrases (also Multi-Terms) gewonnen werden können. In der Anwendung des Modells auf unbekannt Daten werden dann die gesuchten Multi-Terms erzeugt. Der nächste Schritt, die Ähnlichkeit der Multiterms festzustellen und sie in zu untersuchenden Texten zu zählen und zu bewerten, wurde von der Forschergruppe noch nicht gemacht. Ähnliche Arbeiten wie z. B. Zhang u. a. (2004) zeigen jedoch, das dies möglich ist und insbesondere auch zur Verbesserungen der Cluster-Ergebnisse führt. Es gibt eine Vielzahl an Algorithmen aus der Computerlinguistik die Multi-Terms, Sinneinheiten, aus Texten extrahieren. Wir haben diese Verfahren für unsere Arbeit ausgeschlossen, da ihr Rechen- und Implementierungsaufwand für einen ersten Vorstoß in die Richtung dieser Arbeit zu hoch ist. In weiteren Arbeiten sollten die Verfahren der Computerlinguistik jedoch wieder zur Diskussion stehen.

Neben Multi-Term oder Keyphrase Erkennung gibt es auch Versuche sog. Koreferenzen aufzulösen[3]. D.h. Sinneinheiten aus einem vorhergehenden Satz, welche z. B. durch Relativpronomen erneut angesprochen werden, auf ihren ursprünglichen Bezug zu transformieren. Aus den

1 Rechercheansätze haben wir dabei bei Castellví u. a. (2001) und Jacquemin u. Bourigault (2003) gefunden.

2 Keyphrase Extraction Algorithm aus Witten. u. a. (1999) und gleichzeitig eine für ihre Intelligenz berühmte Vogelart Neuseelands, der Heimat der an der Universität von Waikato entstandenen Arbeit.

3 Für einen Überblick, siehe dazu Mitkov (1999) oder Mitkov (2003).

Sätzen: „Computational Linguistics are fun. But they take a lot of computing power.“, würden dann eine Transformation werden: „Computational Linguistics are fun. But they[Computational Linguistics] take a lot of computing power.“. Würde eine solche Transformation auf alle Rohdaten angewendet werden, so würden eventuell die Spezifika jedes Textes besonders betont werden und wichtige Worte eine höhere Wortfrequenz erhalten. Versuche über den Einfluss der Koreferenzauflösung auf Cluster-Verfahren sind jedoch noch mehr oder weniger unbekannt[4]. Ein ähnlich spannendes Verfahren der Computerlinguistik ist die sog. Word-Sense-Disambiguation[5]. Damit werden Verfahren beschrieben, die zwischen verschiedenen Bedeutungen von Worten, anhand des Kontextes in dem sie auftauchen, unterscheiden können. So gibt es z. B. in der Domäne der Informationssicherheit mehrere Bedeutungen des Wortes „Certificate“. Zum einen kann damit ein X.509 Zertifikat aus der asymmetrischen Kryptographie und Netzwerkprotokollsicherheit bezeichnet werden, zum anderen kann mit einem „Certificate“ auch eine Bescheinigung der Befolgung eines Standards gemeint sein. Besonders wenn man, wie wir, nur noch Wortstämme (wie „Certif“, o.ä.) betrachtet, verschwimmen unterschiedliche Konzepte in gleichen Worten immer mehr. Eine Word-Sense-Disambiguation-Stufe vor der weiteren Verarbeitung könnte hier Abhilfe schaffen. Damit diese Systeme funktionieren, benötigen sie jedoch meist externe Informationsquellen wie Wörterbücher, semantische Netze oder ähnliches. Ob die Begriffe der Informationssicherheitsforschung darin ausreichend gut berücksichtigt werden, bleibt leider immernoch fraglich.

Verlässt man den Pfad der Computerlinguistik und wendet sich in die entgegengesetzte Richtung, landet man bei statistisch verwurzelte Verfahren. Diese betrachten üblicherweise in keinster Weise linguistische Informationen, sondern arbeiten rein auf den numerischen Feature-Vektoren. Sie sind damit der Art der Daten gegenüber völlig blind und sehr allgemein anwendbar. Die beiden großen Techniken dort sind die „Latent Semantic Analysis“ und die „Principal Component Analysis“ (dt. auch Hauptkomponentenanalyse)[6]. Sie betrachten die Menge aller Feature-Vektoren als eine Matrix und transformieren diese dann. Das Hauptproblem ist dabei der enorme Speicherbedarf für große Matrizen. Während die üblichen TFIDF-Feature-Vektoren sehr dünn besetzt sind (engl. „sparse“), sprich viele Nullen aufweisen, und deshalb sehr speicherplatzsparend abgelegt und verarbeitet werden können, sind die mathematisch transformierten Matrizen meist voll besetzt. Eine 40.000×40.000 Matrix (in etwa die Größte des IEEE-Corpus) aus 32bit Fließkommazahlen würde voll besetzt eine Größe von etwa 6GB einnehmen. Dies ist mit heutigen herkömmlichen Rechner nicht mehr ohne weiteres handhabbar sondern bedarf schon eines speziellen größeren Rechners, der uns nicht zur Verfügung stand. Die Techniken an sich scheinen jedoch vielversprechend und auch wenn der Implementierungs- und Durchführungsaufwand größer sein sollte, so wäre ein Versuch, in Abstimmung mit anderen Veränderung am Verfahren, vermutlich fruchtbar. Entsprechend notwendige Ressourcen ließen sich auftreiben.

Für andere Methoden der Vorverarbeitung stellen nicht die Rechenressourcen eine Grenze dar, sondern ganz klar der Implementierungs-

4 Zumindest konnten wir keine Literatur dazu ausfindig machen.
5 Für einen Überblick siehe dazu Stevenson u. Wilks (2003).
6 In Abschnitt 3.2 sind wir ausführlicher auf beide Datentransformationen eingegangen.

und Rechercheaufwand. Eine große Informationsquelle für Dokumente wurde nämlich bei all unseren Verfahren überhaupt nicht mit einbezogen: Die Zitationen der Dokumente untereinander. Während diese Informationen bei E-Mails und Newsgroupbeiträgen nicht vorhanden ist, sind wissenschaftliche Arbeit mit Verweisen zueinander ausgestattet. Unter der Annahme, dass ein Verweis ein Hinweis auf Ähnlichkeit ist, ließe sich der Zitationsgraph mit in das entstehende Clustering einbeziehen. Auf dem Weg zu dieser zusätzlichen enormen Informationsquelle, stehen allerdings diverse praktische Probleme: Eine eindeutige und zuverlässige Identifizierung der Zitationen zu erreichen, ist nicht einfach. Bei der Vielzahl an möglichen Zitationssystemen und zusätzlichen Fehlern durch OCR oder teilweise falsche Daten, wird ein absolut korrekter Zitationsgraph nicht zu erreichen sein. Indirekt fließt diese Information in das von uns gewählte Clustering zwar bereits mit ein, denn die Literaturangaben am Ende jedes Artikels wurden nicht entfernt, jedoch wäre ein Zitationsgraph viel präziser.

5.3 CLUSTERING

Die Vorverarbeitung der Daten hat einen enormen Einfluss auf die Ergebnisse des Clusterings, ebenso aber natürlich auch die Ergebnisse. Der von uns gewählte Ansatz des dichtebasierten Clusterns hat den Vorteil der Annahmefreiheit bezüglich der Anzahl der Cluster. Einzig eine Grenzdichte, als Schwelle unterhalb derer Punkte als nicht zu irgendeinem Cluster gehörend verworfen werden, muss experimentell bestimmt werden. Die Anzahl der Cluster folgt dann daraus. Problematisch dabei sind Cluster unterschiedlicher Dichte. Wählt man die Grenzdichte zu hoch, verschwinden sie völlig. Wählt man die Grenzdichte zu niedrig, verbinden sie sich zu schnell mit anderen Clustern.

Ein weiterer Effekte des dichtebasierten Clusterings ist die unregelmäßige Form der Cluster. So können leicht langgezogene Cluster entstehen, deren Inhalte am einen Ende nur noch wenig mit den Inhalten am anderen Ende zu tun haben können. Es ist daher grundsätzlich anzudenken, ob rein dichtebasierte Verfahren der richtige Weg für Text-Mining sind. Die Vorstellung, dass sich ein Thema als kugelförmige Wolke von Punkten manifestiert, wirkt sehr natürlich. Hybride Ansätze aus dichtebasierten Verfahren zum Lernen der gesuchten Anzahl der Cluster und ein anschließendes k-Means Verfahren auf den gefundenen Clustern könnte Abhilfe schaffen. Auch vor einem solchen hybriden Ansatz muss jedoch erst die Schwäche des reinen DBSCAN für Cluster unterschiedlicher Dichte beseitigt werden. Dazu bieten sich entweder Erweiterungen des DBSCAN-Algorithmus oder ebenfalls dichtebasierte, aber an sich völlig verschiedene Cluster-Verfahren an.

Eine Erweiterung des DBSCAN-Algorithmus ist das sog. SNN-Clustering[7] (Shared Nearest Neighbor Clustering). Bei diesem Verfahren geht dem DBSCAN-Algorithmus eine Phase der Distanzberechnung über die gemeinsamen Nachbarn vorraus. Anstelle der eigentlichen Ähnlichkeit zweier Dokumente wird nunmehr die Anzahl der gemeinsamen Nachbarn (die wiederum über herkömmliche Ähnlichkeit bestimmt werden) als Distanzmaß verwendet. Dies sorgt dafür, dass auch Cluster unterschiedlicher Dichte erkannt werden können, führt aber auch einen neuen Parameter ein und erhöht somit den Suchaufwand bei der Parametrisierung. Eine weitere DBSCAN-Variante ist

7 Beschrieben in Ertöz u. a. (2002).

OPTICS[8]. OPTICS berechnet durch einen Trick nicht nur ein dichtebasiertes Clustering sondern quasi alle möglichen Cluster-Konstellationen bei beliebigen Grenzdichten unterhalb einer festlegbaren Grenze. In einer anschließenden visuell unterstützten interaktiven aber auch automatisierbaren Parametrisierung werden dann bestimmte Grenzdichten gewählt und die dazugehörigen Cluster abrufbar.

Eine komplett andere Richtung wäre das sog. hierarchische Clustern. Dabei wird eine Datenmenge bis zu einer Grenze immer wieder hierarchisch in Untermengen zerlegt (oder in der anderen Richtung immer neue Obermengen zusammengesetzt). So entstehen Cluster-Hierarchien, die bei richtiger Dateneingabe verschiedenen Sub-Themen entsprechen würden. Es gibt hochgradig erfolgversprechende Kandidaten, die dichtebasiert arbeiten und die Erfahrungen aus dem DBSCAN- und SNN-Cluster-Verfahren vereinigen. Einer davon ist CHAMELEON[9], ein agglomerativ und dichtebasiert arbeitender Algorithmus. Seine grundsätzliche Struktur ist es aus den Feature-Vektoren, ähnlich wie beim SNN-Clustering, einen k-Nearest-Neighbor-Graph[10] zu erzeugen, diesen dann an seinen schwach verbundenen Kanten in kleine Stücke zu zerlegen und danach wieder neu zu verbinden. Die Verbindung erfolgt dabei anhand der Kriterien Nähe (im Sinne von genereller Näher aller Punkte des einen zu allen Punkten des anderen Clusters) und Verbundenheit (im Sinne von Punkten aus beiden Clustern, die direkt benachbart sind). Der Verbindungsvorgang kann dabei an beliebigen Fortschritten beobachtet werden. Dieses Ergebnis eines hierarchischen Cluster-Algorithmus hat dementsprechend allerdings auch enorm höhere Anforderungen an die Visualisierung, da die Dimension der thematischen Breite beliebig fein gefenstert werden kann und Themen ineinander übergehen. Mei u. Zhai (2005) haben z. B. sog. „Theme Evolutions Graphs“ vorgestellt; Graphen, welche die Abspaltung von Themen über die Zeit visualisieren. Evtl. wäre sogar eine interaktive Visualisierung, mit der man in die Daten hinein zoomen könnte, hilfreich zur Analyse der Daten.

Letztendlich kommt es jedoch auf einen gezielten systematischen, empirischen Vergleich der Verfahren für das Problem und viele Experimente mit den Daten an um einen robusteren Algorithmus zu finden.

5.4 AUSWERTUNG

Die von uns gewählte Auswertung besteht aus den automatisch generierten Extrakten der Cluster-Inhalte und deren normalisierter, quantitativer Verteilung über die Zeit. Während letzteres ein robustes Vorgehen ist[11], sind unsere Summaries in ihrer Qualität noch nicht evaluiert. Es steht also aus, entweder ein formal definiertes und nachgewiesenermaßen robustes Verfahren zur Generierung der Summaries zu verwenden, oder die Robustheit des von uns gewählten Verfahrens nachzuweisen. Letzteres ist ein nicht-trivialer Vorgang[12] und erfordert umfangreiche

8 Vorgestellt in Ankerst u. a. (1999).
9 Beschrieben in Karypis u. a. (1999).
10 In einem k-Nearest-Neighbor-Graph sind die Knoten die zu verarbeitenden Dokumente und eine Kante besteht, wenn ein Knoten unter den k ähnlichsten Knoten eines anderen ist.
11 So z. B. auch von Mei u. Zhai (2005) verwendet.
12 Eine detaillierte Beschreibung der Evaluierungsmöglichkeiten von Zusammenfassungssystemen findet sich z. B. bei Mani u. a. (2002).

Test-Corpora. Aufgrund der in Abschnitt 3.2.4.1 bereits beschriebenen Ähnlichkeit zu anderen Systemen sind wir jedoch überzeugt, dass die Summaries ihrem Zweck in dieser Arbeit gerecht werden.

Neben der Benennung der Cluster ist auch die Visualisierung der Struktur der Cluster für zu ziehende Schlüsse relevant. Dieser Anforderung konnten wir vorerst nur durch einfache Diagramme nachkommen. Gerade beim eventuellen Umstieg auf hierarchische Cluster-Verfahren entstehen umfangreiche, neue Möglichkeiten und eventuelle interaktive Visualisierungen könnten die Struktur der Themenwandel im Diskurs über Informationssicherheit deutlicher und detaillierter zum Ausdruck bringen.

FAZIT

Wir haben mit dieser Arbeit versucht, die intuitiv oder analytisch aufgestellten Erklärungen für den Verlauf der Entwicklung der Informationssicherheit mit einer empirisch Herangehensweise zu ergänzen. Das für diese Arbeit zusammengetragene Datenmaterial ist in seiner Reichweite zwar nicht grenzenlos aber bereits sehr umfangreich und kann als Grundlage für weitere Forschung dienen. Die Methoden des Data-Minings oder spezifisch des Text-Minings bieten dazu eine scheinbar unerschöpfliche Menge von Algorithmen und Verfahren um mit diesen unstrukturierten Informationen in Form von Texten umzugehen. Darin begründet sich nun aber die große Schwierigkeit unseres speziellen Unternehmens: Welches Verfahren liefert eine geeignete Antwort auf die Fragen die wir stellen wollen?

Das von uns gewählte Verfahren ist in der Lage auch große Text-Corpora performant zu verarbeiten und kann, da es vollständig Domänen-unspezifisch ausgelegt ist, auf beliebig speziellen oder allgemeinen Wissensgebieten angewendet werden. Es zeigen sich damit erste vielversprechende Ergebnisse, die jedoch in ihrer Qualität noch nicht zufriedenstellend sind. In der Interpretation dieser vorläufigen Resultate konnten drei Thesen aufgestellt werden. Sie deuten eine Entwicklung von einfachen technischen Anwendungen (Basistechnologien) hin zu komplexeren Technologien und Management-Ansätzen an.

Wenn wir auch unser anfänglich gesetztes Ziel, eine robuste Methode zu finden, mit der sich der Verlauf von Diskursen verfolgen lässt, noch nicht erreicht haben, so sind wir ihm doch etwas näher gekommen. Die Ergebnisse stellen Indizien dar, welche unserer Meinung nach in die richtige Richtung weisen. Sie zeigen, dass sich die Themengebiete der technischen Informationssicherheit wandeln und Management-Ansätze immer wichtiger werden. Vieles muss jedoch noch getan werden um aus diesen Indizien handfestes Material zu machen.

ACM 2001
ACM: *TISSEC - Topics of Interest*. Webseite. http://tissec.acm.org/Topics.html. Version: Oktober 2001. – Zuletzt abgerufen am 04.09.2009, Kopie beim Autor archiviert. (Zitiert auf Seite 6.)

Angermüller 2001
Angermüller, Johannes: Diskursanalyse: Strömungen, Tendenzen, Perspektiven. In: Angermüller, Johannes (Hrsg.) ; Bunzmann, Katharina (Hrsg.) ; Nonhoff, Martin (Hrsg.): *Diskursanalyse: Theorien, Methoden, Anwendungen.* Hamburg : Argument, 2001, S. 7–22 (Zitiert auf Seite 13.)

Ankerst u. a. 1999
Ankerst, Mihael ; Breunig, Markus M. ; Kriegel, Hans-Peter ; Sander, Jörg: OPTICS: ordering points to identify the clustering structure. In: *Proceedings of the 1999 ACM SIGMOD international conference on Management of data*, 1999, S. 49–60 (Zitiert auf Seite 119.)

Barzilay u. a. 1999
Barzilay, Regina ; McKeown, Kathleen R. ; Elhadad, Michael: Information fusion in the context of multi-document summarization. In: *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics* Association for Computational Linguistics Morristown, NJ, USA, 1999, S. 550–557 (Zitiert auf Seite 55.)

Beil u. a. 2002
Beil, F. ; Ester, M. ; Xu, X.: Frequent term-based text clustering. In: *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining* ACM New York, NY, USA, 2002, S. 436–442 (Zitiert auf den Seiten 51, 52 und 53.)

Blum u. Langley 1997
Blum, Avrim L. ; Langley, Pat: Selection of relevant features and examples in machine learning. In: *Artificial Intelligence* 97 (1997), Nr. 1-2, S. 245–271 (Zitiert auf Seite 116.)

Boros u. a. 2001
Boros, E. ; Kantor, P.B. ; Neu, D.J.: A clustering based approach to creating multi-document summaries. In: *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2001 (Zitiert auf Seite 57.)

Bundesamt für Sicherheit in der Informationstechnik 2008a
Bundesamt für Sicherheit in der Informationstechnik: *BSI-Standard 100-1: Managementsysteme für Informationssicherheit (ISMS).* https://www.bsi.bund.de/cae/servlet/contentblob/471450/publicationFile/30759/standard_1001_pdf.pdf. Version: Mai 2008 (Zitiert auf Seite 1.)

Bundesamt für Sicherheit in der Informationstechnik 2008b
Bundesamt für Sicherheit in der Informationstechnik: *IT-Sicherheitskriterien und Evaluierung nach*

ITSEC. Webseite. https://www.bsi.bund.de/cln_155/ContentBSI/Themen/ZertifizierungundAkkreditierung/ZertifierungnachCCundITSEC/ITSicherheitskriterien/ITSEC/itsec_eval.html. Version: Mai 2008. – zuletzt abgerufen 06.09.2009, Kopie beim Autor archiviert (Zitiert auf Seite 83.)

Cartwright 2009
CARTWRIGHT, John: *Full-Disclosure Mailing List Charter*. Webseite. http://lists.grok.org.uk/full-disclosure-charter.html. Version: 2009. – zuletzt abgerufen am 08.09.2009, Kopie beim Autor archiviert (Zitiert auf Seite 10.)

Castellví u. a. 2001
CASTELLVÍ, M. Teresa C. ; BAGOT, Rosa E. ; PALATRESI, Jordi V.: Automatic Term Detection: A Review Of Current Systems / Institut Universitari de Lingüística Aplicada, Universitat Pompeu Fabra La Rambla, Barcelona. 2001. – Forschungsbericht (Zitiert auf den Seiten 21 und 116.)

Cavnar u. Trenkle 1994
CAVNAR, W.B. ; TRENKLE, J.M.: N-gram-based Text Categorization. In: *Ann Arbor MI* 48113 (1994), S. 4001 (Zitiert auf Seite 20.)

Clark u. a. 2005
CLARK, David D. ; WROCLAWSKI, John ; SOLLINS, Karen R. ; BRADEN, Robert: Tussle in cyberspace: defining tomorrow's internet. In: *IEEE/ACM Transactions on Networking* 13 (2005), June, Nr. 3, S. 462–475 (Zitiert auf Seite 105.)

Ertöz u. a. 2002
ERTÖZ, Levent ; STEINBACH, Michael ; KUMAR, Vipin: A new shared nearest neighbor clustering algorithm and its applications. In: *Workshop on Clustering High Dimensional Data and its Applications, SIAM Data Mining*, 2002 (Zitiert auf Seite 118.)

Ester u. a. 1996
ESTER, M. ; KRIEGEL, H.P. ; SANDER, J. ; XU, X.: A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In: *Proc. 2nd Int. Conf. on Knowledge Discovery and Data Mining, Portland, OR, AAAI Press*, 1996, S. 226–231 (Zitiert auf den Seiten 47, 49, 51 und 69.)

FCN/FIA 2009
FCN/FIA: Why do we need a Content-Centric Internet? Proposals towards Content-Centric Internet Architectures / Future Content Networks Group, Future Internet Assembly, European Commission, Information Society and Media. Version: Mai 2009. http://www.future-internet.eu/fileadmin/documents/prague_documents/FIA-FCN_Internet_Architecture_20090507.pdf. 2009. – Forschungsbericht (Zitiert auf Seite 105.)

Frank u. a. 1999
FRANK, Eibe ; PAYNTER, Gordon W. ; WITTEN, Ian H. ; GUTWIN, Carl ; NEVILL-MANNING, Craig G.: Domain-Specific Keyphrase Extraction. In: *Internation Joint Conference on Artificial Intelligence* Bd. 16, 1999, S. 668–673 (Zitiert auf Seite 21.)

Frankfurter Allgemeine Zeitung 2009
FRANKFURTER ALLGEMEINE ZEITUNG: *Frankfurter Allgemeine Archiv*. Webseite. http://fazarchiv.faz.net/. Version: 2009. – zuletzt abgerufen am 01.06.2009, Kopie beim Autor archiviert (Zitiert auf Seite 4.)

Fuller u. Zobel 1998
FULLER, M. ; ZOBEL, J.: Conflation-based comparison of stemming algorithms. In: *In Proceedings of the Third Australian Document Computing Symposium*, 1998 (Zitiert auf Seite 22.)

Gates 2001
GATES, Bill: *Trustworthy computing*. Interne Memo E-Mail, abgedruckt u.a. in Wired Magazine. http://www.wired.com/techbiz/media/news/2002/01/49826. Version: January 2001. – zuletzt abgerufen am 06.09.2009, Kopie beim Autor archiviert (Zitiert auf Seite 90.)

Gelbukh u. Sidorov 2001
GELBUKH, Alexander ; SIDOROV, Grigori: Zipf and Heaps Laws' Coefficients Depend on Language. In: *Computational Linguistics and Intelligent Text Processing*, Springer, 2001 (Lecture Notes in Computer Science), S. 332–335 (Zitiert auf Seite 94.)

Gellens 2004
GELLENS, R.: *RFC 3676*. http://www.ietf.org/rfc/rfc3676.txt. Version: 2004. – Network Working Group / IETF (Zitiert auf Seite 63.)

GIAC 2009
GIAC: *GSEC Certified Professionals*. Webseite. http://www.giac.org/certified_professionals/listing/gsec.php. Version: September 2009. – zuletzt abgerufen am 06.09.2009, Kopie beim Autor archiviert (Zitiert auf Seite 87.)

Goldstein u. a. 2000
GOLDSTEIN, Jade ; MITTAL, Vibbu ; CARBONELL, Jaime ; KANTROWITZ, Mark: Multi-document summarization by sentence extraction. In: *Proceedings of the ANLP/NAACL Workshop on Automatic Summarization*, 2000, S. 40–48 (Zitiert auf Seite 57.)

Graham u. Denning 1971
GRAHAM, G. S. ; DENNING, Peter J.: Protection: principles and practice. In: *Proceedings of the November 16-18, 1971, fall joint computer conference*. Las Vegas, Nevada : ACM, November 1971, S. 417–429 (Zitiert auf Seite 6.)

Harrison u. a. 1976
HARRISON, Michael A. ; RUZZO, Walter L. ; ULLMAN, Jeffrey D.: Protection in operating systems. In: *Communications of the ACM* 19 (1976), August, Nr. 8, S. 461–471 (Zitiert auf Seite 6.)

Havre u. a. 2000
HAVRE, Susan ; HETZLER, Beth ; NOWELL, Lucy: ThemeRiver: Visualizing Theme Changes over Time. In: *Proceedings of the IEEE Symposium on Information Visualization 2000 (InfoVis'00)*, 2000. – ISBN 0–7695–0804–9, S. 115–123 (Zitiert auf den Seiten 18 und 58.)

Hegland 2007
Hegland, M.: The Apriori Algorithm - a Tutorial. In: *Mathematics and Computation in Imaging Science and Information Processing* (2007), S. 209 (Zitiert auf Seite 52.)

Hovy 2003
Hovy, Eduard: Text Summarization. In: Mitkov, Ruslan (Hrsg.): *The Oxford Handbook of Computational Linguistics*. New York : Oxford University Press, 2003, Kapitel 32, S. 583–598 (Zitiert auf Seite 55.)

Hsu u. Lin 2002
Hsu, C.W. ; Lin, C.J.: A comparison of methods for multiclass support vector machines. In: *IEEE Transactions on Neural Networks* 13 (2002), Nr. 2, S. 415–425 (Zitiert auf Seite 43.)

Huang u. a. 2003
Huang, J. ; Lu, J. ; Ling, CX: Comparing naive Bayes, decision trees, and SVM with AUC and accuracy. In: *Third IEEE International Conference on Data Mining, 2003. ICDM 2003*, 2003, S. 553–556 (Zitiert auf Seite 44.)

IEEE 1999
IEEE: *IEEE Standard 802.11a-1999 Wireless Lan Medium Access Control (MAC) And Physical Layer (PHY) Specifications*. http://ieeexplore.ieee.org/servlet/opac?punumber=6606. Version: 1999 (Zitiert auf Seite 79.)

ITSEC 1998
ITSEC: *Information Technology Security Evaluation Criteria — ITSEC*. Standard. https://www.bsi.bund.de/cae/servlet/contentblob/478074/publicationFile/30221/itsec-dt_pdf.pdf. Version: März 1998 (Zitiert auf Seite 83.)

Jacquemin u. Bourigault 2003
Jacquemin, Christian ; Bourigault, Didier: Term Extraction and Automatic Indexing. In: Mitkov, Ruslan (Hrsg.): *The Oxford Handbook of Computational Linguistics*. New York : Oxford University Press, 2003, Kapitel 33, S. 599–615 (Zitiert auf den Seiten 21 und 116.)

Jaro 1989
Jaro, M.A.: Advances in record-linkage methodology as applied to matching the 1985 census of Tampa, Florida. In: *Journal of the American Statistical Association* 84 (1989), June, Nr. 406, S. 414–420 (Zitiert auf Seite 26.)

Jovanoski u. Lavrac 2001
Jovanoski, V. ; Lavrac, N.: Classification rule learning with APRIORI-C. In: *Lecture notes in computer science* (2001), S. 44–51 (Zitiert auf Seite 52.)

Karypis u. a. 1999
Karypis, George ; Han, Eui-Hong S. ; Kumar, Vipin: Chameleon: Hierarchical clustering using dynamic modeling. In: *IEEE Computer Magazine* 32 (1999), Nr. 8, S. 68–75 (Zitiert auf Seite 119.)

Knowles 2009
Knowles, William: *InfoSecNews.org*. Webseite. http://www.

infosecnews.org/. Version: 2009. – zuletzt abgerufen am 08.09.2009, Kopie beim Autor archiviert (Zitiert auf Seite 10.)

Kukich 1992
Kukich, Karen: Techniques for Automatically Correcting Words in Text. In: *ACM Computing Surveys* 24 (1992), S. 377–439 (Zitiert auf Seite 27.)

Leiner u. a. 1997
Leiner, Barry M. ; Cerf, Vinton G. ; Clark, David D. ; Kahn, Robert E. ; Kleinrock, Leonard ; Lynch, Daniel C. ; Postel, Jon ; Roberts, Lawrence G. ; Wolff, Stephen S.: The past and future history of the Internet. In: *Communications of the ACM* 40 (1997), Februar, Nr. 2, S. 102–108 (Zitiert auf Seite 9.)

Levenshtein 1966
Levenshtein, V. I.: Binary Codes Capable of Correcting Deletions, Insertions and Reversals. In: *Soviet Physics Doklady* 10 (1966), S. 707 (Zitiert auf Seite 26.)

Lipton u. Snyder 1977
Lipton, R. J. ; Snyder, L.: A Linear Time Algorithm for Deciding Subject Security. In: *Journal of the ACM* 24 (1977), July, Nr. 3, S. 455–464 (Zitiert auf Seite 6.)

Mani u. a. 2002
Mani, I. ; Klein, G. ; House, D. ; Hirschman, L. ; Firmin, T. ; Sundheim, B.: SUMMAC: a text summarization evaluation. In: *Natural Language Engineering* 8 (2002), Nr. 01, S. 43–68 (Zitiert auf Seite 119.)

McCarthy 1992
McCarthy, John: Reminiscences on the History of Time-Sharing. In: *IEEE Annals of the History of Computing* 14 (1992), Nr. 1, S. 19–24 (Zitiert auf Seite 3.)

Mei u. Zhai 2005
Mei, Qiaozhu ; Zhai, Cheng X.: Discovering Evolutionary Theme Patterns from Text – An Exploration of Temporal Text Mining. In: *Proceedings of KDD '05*, 2005, S. 198–207 (Zitiert auf Seite 119.)

Minnen u. a. 2001
Minnen, G. ; Carroll, J. ; Pearce, D.: Applied morphological processing of English. In: *Natural Language Engineering* 7 (2001), Nr. 03, S. 207–223 (Zitiert auf Seite 23.)

Mitkov 1999
Mitkov, Ruslan: Anaphora Resolution: The State of The Art / School of Languages and European Studies, University of Wolverhampton. 1999. – Forschungsbericht (Zitiert auf Seite 116.)

Mitkov 2003
Mitkov, Ruslan: Anaphora Resolution. In: Mitkov, Ruslan (Hrsg.): *The Oxford Handbook of Computational Linguistics*. New York : Oxford University Press, 2003, Kapitel 14, S. 266–283 (Zitiert auf Seite 116.)

New York Times 2009
New York Times: *New York Times Archive*. Webseite. http://www.nytimes.com/ref/membercenter/nytarchive.html. Version: 2009.

– zuletzt abgerufen am 01.06.2009, Kopie beim Autor archiviert (Zitiert auf Seite 4.)

Náther 2005
Náther, Peter: *N-gram based Text Categorization*, Comenius University Bratislava, Diploma Thesis, 2005 (Zitiert auf Seite 20.)

Pallas 2009
Pallas, Frank: *Information Security Inside Organizations – A Positive Model and Some Normative Arguments Based on New Institutional Economics*, Technische Universität Berlin, Dissertation, April 2009. http://opus.kobv.de/tuberlin/volltexte/2009/2320/ (Zitiert auf den Seiten v, ix, 1, 111, 112 und 113.)

PCI Security Standards Council, LLC 2008
PCI Security Standards Council, LLC: *About the PCI Data Security Standard (PCI DSS)*. Webseite. https://www.pcisecuritystandards.org/security_standards/pci_dss.shtml. Version: Oktober 2008. – zuletzt abgerufen am 06.09.2009, Kopie beim Autor archiviert (Zitiert auf Seite 87.)

Pearson 1901
Pearson, Karl: On lines and planes of closest fit to systems of points in space. In: *Philosophical Magazine Series 6* 2 (1901), Nr. 11, S. 559–572 (Zitiert auf Seite 34.)

Porter 2006
Porter, MF: An algorithm for suffix stripping. In: *Program 1966-2006: Celebrating 40 Years of ICT in Libraries, Museums and Archives* (2006) (Zitiert auf Seite 23.)

Radev u. a. 2004
Radev, D.R. ; Jing, H. ; Styś, M. ; Tam, D.: Centroid-based summarization of multiple documents. In: *Information Processing and Management* 40 (2004), Nr. 6, S. 919–938 (Zitiert auf den Seiten 56 und 57.)

Russell u. Norvig 2004
Russell, S.J. ; Norvig, P.: *Künstliche Intelligenz: Ein moderner Ansatz*. Pearson Education Deutschland, 2004 (Zitiert auf Seite 38.)

Schrage 1999
Schrage, Dominik: Was ist ein Diskurs? – Zu Michel Foucaults Versprechen, ">mehr"< ans Licht zu bringen. In: Seier, Andrea (Hrsg.): *Das Wuchern der Diskurse. Perspektiven der Diskursanalyse Foucaults.* Frankfurt a.M. : Campus, 1999, S. 63–74 (Zitiert auf Seite 13.)

Security Focus 2006a
Security Focus: *BugTraq*. Webseite. http://www.securityfocus.com/archive/1/description. Version: 2006. – zuletzt abgerufen am 08.09.2009, Kopie beim Autor archiviert (Zitiert auf Seite 9.)

Security Focus 2006b
Security Focus: *Security-Basics*. Webseite. http://www.securityfocus.com/archive/105/description. Version: 2006. – zuletzt abgerufen am 08.09.2009, Kopie beim Autor archiviert (Zitiert auf Seite 10.)

Shlens 2005
Shlens, Jonathan: *A Tutorial on Principal Component Analysis*. http://www.brainmapping.org/NITP/PNA/Readings/pca.pdf. Version: Dezember 2005. – zuletzt abgerufen am 26.08.2009, Kopie beim Autor archiviert (Zitiert auf den Seiten 34 und 35.)

Smith 2002
Smith, Lindsay I.: *A Tutorial on Principal Component Analysis*. http://users.ecs.soton.ac.uk/hbr03r/pa037042.pdf. Version: Februar 2002. – zuletzt abgerufen am 26.08.2009, Kopie beim Autor archiviert (Zitiert auf Seite 34.)

von Solms 2000
Solms, B. von: Information Security–The Third Wave? In: *Computers & Security* 19 (2000), Nr. 7, S. 615–620 (Zitiert auf den Seiten v, ix, 1, 111 und 112.)

von Solms 2006
Solms, B. von: Information Security–The Fourth Wave. In: *Computers & Security* 25 (2006), Nr. 3, S. 165–168 (Zitiert auf Seite 112.)

Stevenson u. Wilks 2003
Stevenson, Mark ; Wilks, Yorick: Word-Sense Disambiguation. In: Mitkov, Ruslan (Hrsg.): *The Oxford Handbook of Computational Linguistics*. New York : Oxford University Press, 2003, Kapitel 13, S. 249–265 (Zitiert auf Seite 117.)

Strehl u. a. 2000
Strehl, A. ; Ghosh, J. ; Mooney, R.: Impact of similarity measures on web-page clustering. In: *Proc. AAAI Workshop on AI for Web Search (AAAI 2000), Austin*, 2000, S. 58–64 (Zitiert auf Seite 50.)

Tan u. a. 2005
Tan, Pang-Ning ; Steinbach, Michael ; Kumar, Vipin: *Introduction to Data Mining*. Addison-Wesley / Pearson, 2005 (Zitiert auf den Seiten 15, 42, 47 und 50.)

Tong u. Evans 1996
Tong, Xiang ; Evans, David A.: A Statistical Approach to Automatic OCR Error Correction in Context. In: *Proceedings of the fourth workshop on very large corpora*. Copenhagen, Denmark, August 1996, S. pp88 (Zitiert auf den Seiten 27 und 66.)

Turner u. a. 2005
Turner, Tammara C. ; Smith, Marc A. ; Fisher, Danyel ; Welser, Howard T.: Picturing Usenet: Mapping Computer-Mediated Collective Action. In: *Journal of Computer-Mediated Communication* 10 (2005), Nr. 4. http://jcmc.indiana.edu/vol10/issue4/turner.html (Zitiert auf Seite 9.)

U.S. Departement Of Justice 1999
U.S. Departement Of Justice: *Kevin Mitnick Sentenced to Nearly Four Years in Prison*. Pressemitteilung. http://www.usdoj.gov/criminal/cybercrime/mitnick.htm. Version: August 1999. – zuletzt abgerufen am 06.09.2009, Kopie beim Autor archiviert (Zitiert auf Seite 83.)

Verleysen u. François 2005
Verleysen, Michel ; François, Damien: The curse of dimensionality in data mining and time series prediction. In: Cabestany, J. (Hrsg.) ; Prieto, A. (Hrsg.) ; Sandoval, D.F. (Hrsg.) ; Springer (Veranst.): *IWANN* Springer, 2005, S. 758–770 (Zitiert auf Seite 78.)

Wagner u. Fischer 1974
Wagner, Robert A. ; Fischer, Michael J.: The String-to-String Correction Problem. In: *Journal of the ACM* 21 (1974), January, Nr. 1, S. 168–173 (Zitiert auf Seite 27.)

Ware 1979
Ware, Willis H.: Security Controls for Computer Systems: Report of Defense Science Board Task Force on Computer Security / Secretary of Defense. RAND Corporation, 1979. – Forschungsbericht (Zitiert auf Seite 3.)

Weiss u. a. 1999
Weiss, Sholom M. ; Damerau, Fred J. ; Johnson, David E. ; Oles, Frank J. ; Goetz, Thilo: Maximizing text-mining performance. In: *IEEE Intelligent Systems* (1999) (Zitiert auf Seite 78.)

Weiss u. a. 2005
Weiss, S.M. ; Indurkhya, N. ; Zhang, T. ; Damerau, F.J.: *Text Mining: Predictive Methods for Analyzing Unstructured Information*. Springer-Verlag New York Inc, 2005 (Zitiert auf den Seiten 20 und 24.)

Whitworth u. Friedman 2009
Whitworth, Brian ; Friedman, Rob: Reinventing academic publishing online. Part I: Rigor, relevance and practice. In: *First Monday* 14 (2009), August, Nr. 8. `http://www.uic.edu/htbin/cgiwrap/bin/ojs/index.php/fm/article/view/2609/2248` (Zitiert auf Seite 5.)

Witten u. Frank 2005
Witten, Ian H. ; Frank, Eibe ; Gray, Jim (Hrsg.): *Data Mining: Practical Machine Learning Tools and Techniques*. 2nd ed. Elsevier, 2005 (Morgan Kaufmann series in data management systems) (Zitiert auf den Seiten 14, 15 und 17.)

Witten. u. a. 1999
Witten., Ian H. ; Paynter, Gordon W. ; Frank, Eibe ; Gutwin, Carl ; Nevill-Manning, Craig G.: KEA: Practical Automatic Keyphrase Extraction. In: *Proceedings of the fourth ACM conference on Digital libraries*. Berkeley, California, United States, 1999, S. 254–255 (Zitiert auf den Seiten 21 und 116.)

Wright 2009
Wright, Craig S.: *Cracked, inSecure and Generally Broken*. Blog. `http://gse-compliance.blogspot.com/`. Version: September 2009. – zuletzt abgerufen am 06.09.2009, Snapshot beim Autor archiviert (Zitiert auf Seite 87.)

Yates u. Neto 1999
Yates, R.B. ; Neto, B.R.: *Modern information retrieval*. New York, Addision Wesley, 1999 (Zitiert auf Seite 30.)

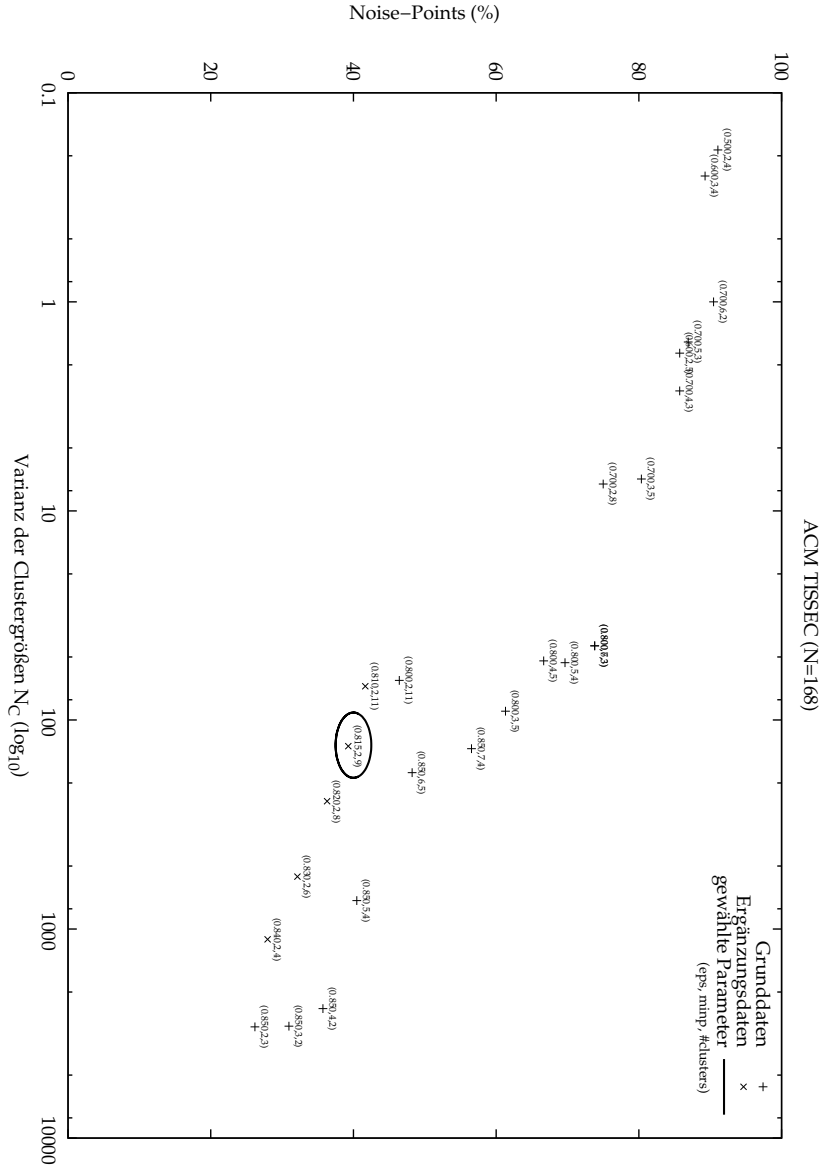
Yu u. a. 2002
Yu, C. ; Cuadrado, J. ; Ceglowski, M. ; Payne, J.S.: Patterns in unstructured data: Discovery, aggregation, and visualization. In: *Presentation to the Andrew W. Mellon Foundation* (2002) (Zitiert auf Seite 31.)

Zhang u. a. 2004
Zhang, Yongzheng ; Zincir-Heywood, Nur ; Milios, Evangelos: Term-Based Clustering and Summarization of Web Page Collections. In: *Lecture Notes in Computer Science* (2004), S. 60–74 (Zitiert auf den Seiten 56 und 116.)

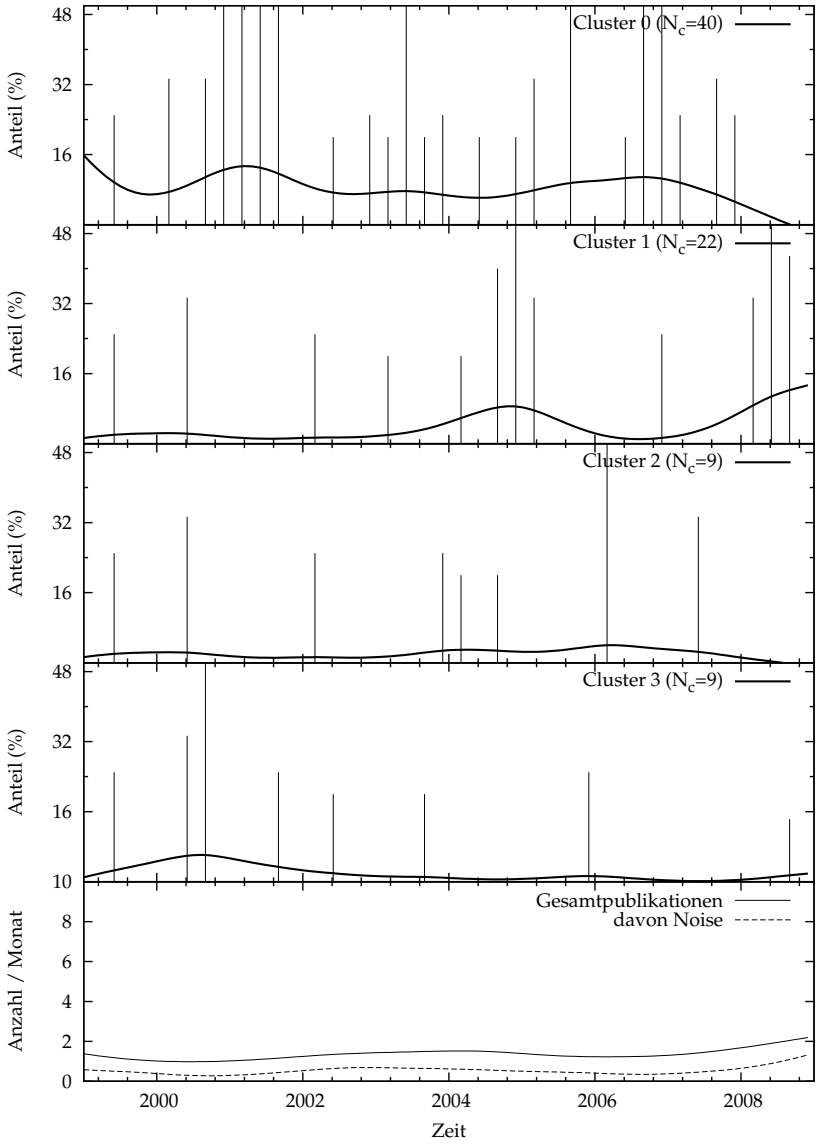
A ANHANG

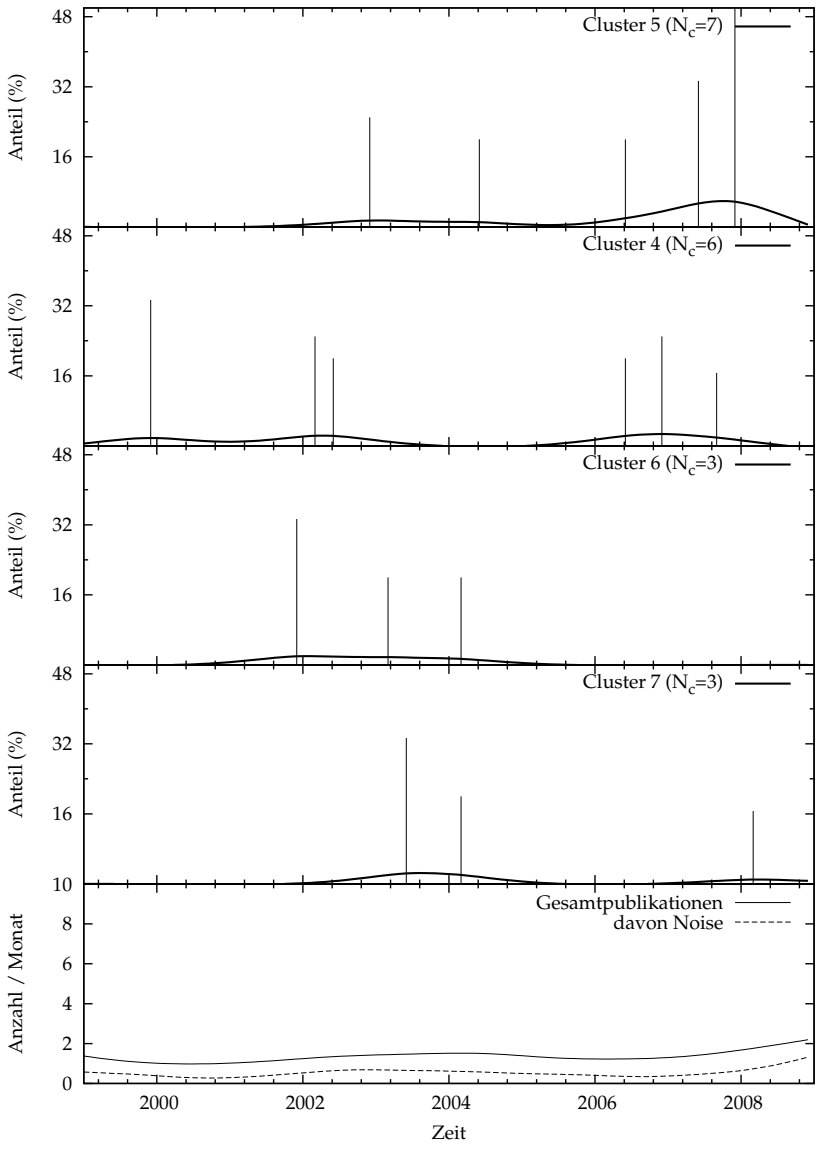
A.1 ACM TISSEC

Cluster-Parametrisierung



Publikationsvolumen




Cluster 5 (N_c=7)
Cluster 4 (N_c=6)
Cluster 6 (N_c=3)
Cluster 7 (N_c=3)
Anteil (%)
Gesamtpublikationen
davon Noise
Anzahl / Monat
Zeit
2000
2002
2004
2006
2008

Zusammenfassungen

Cluster 0 ($N_C = 40$) „Access Control“

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
role rbac password deleg certif nest tissec dictionari cooki hierarchi constraint lock gtrbac fpa assign sandhu geo ssd pso administr ucon workflow rcl stroke spatial editori dsd ura attribut admin conflict oblig editor lbac polici senior alic trbac revoc sarbac novemb xml revok dict npki subspac att xorbac unionsqtext dblarrowheadleft duti arbac princip pra schema dso access inherit guest document	Administrative Scope: A Foundation for Role-Based Administrative Models / Formal Model and Policy Specification of Usage Control / The ARBAC97 Model for Role-Based Administration of Roles / Role-Based Access Control on the Web / Flexible Access Control Policy Specification with Constraint Logic Programming / An Integrated Approach to Engineer and Enforce Context Constraints in RBAC Environments / Specification and Verification of Security Requirements in a Programming Model for Decentralized CSCW Systems / A Rule-Based Framework for Role-Based Delegation and Revocation / X-GTRBAC: An XML-Based Policy Specification Framework and Architecture for Enterprise-Wide Access Control / Security Analysis in Role-Based Access Control / Proposed NIST Standard for Role-Based Access Control / Formal	Use of Nested Certificates for Efficient, Dynamic, and Trust Preserving Public Key Infrastructure Sabanci University M. Bogazici University and CETIN K. KOC Oregon State / Control information, together with its corresponding flow policy form, the so-called flow policy attachment (Fpa). / To demonstrate feasibility, we implement each architecture by integrating and extending well-known technologies such as cookies, X.509, SSL, and LDAP, providing compatibility with / In this article, we introduce an intuitive formal language for specifying role-based authorization constraints named RCL 2000 including its basic elements, syntax, and semantics. / These cognitive studies motivate us to define a set of password complexity factors (e.g., reflective symmetry and stroke count), which define a

Cluster 1 ($N_C = 22$) „Sensor & Wireless Security“

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
sensor node alert piv hyperalert packet attack pivss pairwis correl multicast cell predistribut destip overlay atm compromis aio scheme probabl ring beacon pivc hypothes path hop aggreg attest emul game infer polynomi intrus dapp prime lch graph instruct articl liu router mote pub wireless predecessor incent xval flash round firewal inject random onion victim neighbor leaf crowd traceback som redoubt	Distributed Authentication of Program Integrity Verification in Wireless Sensor Networks / Message Dropping Attacks in Overlay Networks: Attack Detection and Attacker Identification / SDAP: A Secure Hop-by-Hop Data Aggregation Protocol for Sensor Networks / Attack-Resistant Location Estimation in Wireless Sensor Networks / Secure Time Synchronization in Sensor Networks / A Framework for Identifying Compromised Nodes in Wireless Sensor Networks / Establishing Pairwise Keys in Distributed Sensor Networks / On the Construction of Practical Key Predistribution Schemes for Distributed Sensor Networks Using Combinatorial Designs / Redoubtable Sensor Networks / Hypothesizing and Reasoning about Attacks Missed by Intrusion Detection Systems / The Predecessor Attack: An Analysis of a Threat to	Park and Shin (2005) proposed a soft tamper-proofing scheme that verifies the integrity of the program in each sensor device, called the program integrity verification (PIV), in / In addition, our method provides an intuitive mechanism (called hyperalert correlation graph) to represent the attack scenarios constructed through alert correlation. / 15 Message Dropping Attacks in Overlay Networks: Attack Detection and Attacker Identification LIANG XIE and SENCUN ZHU The Pennsylvania State University Overlaymulticastnetworks / Modeling Network Intrusion Detection Alerts for Correlation JINGMIN ZHOU University of California, Davis MARK HECKMAN and BRENNEN REYNOLDS Promia, Inc. and ADAM CARLSON and / Hypothesizing and Reasoning about Missed Attacks 603 Table I. Hyperalert Types Used in Example

Cluster 2 ($N_C = 9$) „Cryptographic Protocols“

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
ttp prime spi handshak protocol ipsec ike messag nro session nonc jfk client keynot commit hash agent cssc rsa jfkr paulson oracl cramer signatur nrr shoup smooth lemma inj fast isabel expt squiggleright encrypt cach jfki certif fair track exchang induct nonrepudi trace primeprim respond analz tildewid packet proverif credenti resumpt certifi receipt nonmal adversari master clientk sessionk bella gennaro	Just Fast Keying in the Pi Calculus MART / Accountability Protocols: Formalized and Verified / Inductive Analysis of the Internet Protocol TLS / Just Fast Keying: Key Agreement in a Hostile Internet / A Framework for Password-Based Authenticated Key Exchange / Client-Side Caching for TLS / Verifiable Encryption of Digital Signatures and Applications / Trust Management for IPsec / Signature Schemes Based on the Strong	This section presents the nonrepudiation protocol (Zhou and Gollmann 1996) and the certified email protocol (Abadi et al. 2002). Both protocols have two peers and a trusted / Client-Side Caching for TLS HOVAV SHACHAM, DAN BONEH, and Stanford University We propose two new mechanisms for caching handshake information on TLS clients. / 2.1 The JFKr Variant The JFKr protocol consists of the following four messages: Message 1 I R : N I , x I Message 2 R I : N I , N R , x R , g R , t R Message 3 I R : N I , N R , x / We make use of a trusted third party (TTP) but in an optimistic sense, that is, the TTP takes part in the protocol only if one user cheats or simply crashes. / My version hashes message components rather than messages in order to simplify the inductive definition, as a

Cluster 3 ($N_C = 9$) „Intrusion Detection“

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
alarm intrus detect train anomali roc cluster detector transduc rate gram score audit event inspect curv fals bayesian anomal niy bsm idss omega profil fallaci bracehtipupleft bracehtipupright axelsson ryu greedi attack min tangent ghosh learn lincoln rhee signatur root ripper week mine string accuraci evtsch lippmann iso classif charact host darpa normal novemb benign brodley predset probabl deviat workflow pattern	Anomalous System Call Detection / A Framework for Constructing Features and Models for Intrusion Detection Systems / Testing Intrusion Detection Systems: A Critique of the 1998 and 1999 DARPA Intrusion Detection System Evaluations as Performed by Lincoln Laboratory / Evaluation of Intrusion Detection Systems Under a Resource Constraint / Simple, State-Based Approaches to Program-Based Anomaly Detection / The Base-Rate Fallacy and the Difficulty of Intrusion Detection / Temporal Sequence Learning and Data Reduction for Anomaly Detection / Clustering Intrusion Detection Alarms to Support Root Cause Analysis / Abstraction-Based Intrusion Detection In Distributed Environments	Clustering Intrusion Detection Alarms to Support Root Cause Analysis KLAUS JULISCH IBM Research, Zurich Research Laboratory It is a well-known problem that intrusion detection / The state-based anomaly detector just described detects things that have never happened before. / 20 Evaluation of Intrusion Detection Systems Under a Resource Constraint YOUNG U. RYU and HYEUN-SUK RHEE The University of Texas at Dallas An intrusion detection system plays an / The Base-Rate Fallacy and the Difficulty of Intrusion Detection Ericsson Mobile Data Design AB Many different demands can be made of intrusion detection systems. / We present several techniques for reducing data storage requirements of the user profile, including instance-selection methods and clustering. / A Framework for Constructing

Cluster 5 ($N_C = 7$) „Trust Negotiation“

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
negoti traust credenti disclosur polici xacml ttg disclos poe trust ack alic trustbuild strategi bob prime attribut client atn gridftp attributevalu liba sensit wice collabor travers seamon articl dblarrowheadleftk oppon resourc parti boundari tree busey authoriza organiz cred winslett winsborough hess safeti functionid unack libb messag overrid doi datatyp proxi mazzoleni attrib deni lockss pol unlock shuffl libc node swampland	The Traust Authorization Service / Safety in Automated Trust Negotiation / PP-Trust-X: A System for Privacy Preserving Trust Negotiations / Supporting Structured Credentials and Sensitive Policies through Interoperable Strategies for Automated Trust Negotiation / XACML Policy Integration Algorithms / Content-Triggered Trust Negotiation / Editorial ACM and SIGSAC currently sponsor an annual symposium on access control models and technologies (SACMAT). Research results and experience reports, presented in these	2 The Traust Authorization Service ADAM J. LEE and University of Illinois at Urbana-Champaign and JIM BASNEY and VON WELCH National Center for Supercomputing Applications In / 5. A family of strategies based on the Trust Target Graph (TTG) protocol (Winsborough and Li 2002b) that supports flexibility in the search for a successful negotiation. / PP-Trust-X: A System for Privacy Preserving Trust Negotiations A. SQUICCIARINI and E. BERTINO Purdue University ELENA FERRARI Universita degli Studi dellInsubria, Varese F. PACI / Since digital credentials themselves can contain sensitive information, their disclosure can also be governed by access control policies. / 4: 2 P. Mazzoleni et al. 2005). The language proposes an approach to manage AC constraints in large enterprisesystemsthat oftenhave

Cluster 4 ($N_C = 6$) „XML Security“

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
xml document dtd acp automaton xpath tid schema automata node soe veclabel sigma polici queri gamma xverpath pack delta vdate subtre attribut murata idref element predic pend xqueri credenti encrypt iwaihara cardiolog descend xlink tag subel diagnosi safeti info bouganim corporateanalysi static enforc token crd vroot pcdata doc graph patholog patientid dld record access prime skip chemotherapi articl target fileread	XML Access Control Using Static Analysis / Relevancy-Based Access Control and Its Evaluation on Versioned XML Documents / Dynamic Access-Control Policies on XML Encrypted Data / A Fine-Grained Access Control System for XML Documents / Secure and Selective Dissemination of XML Documents / Enforceable Security Policies	In what follows, given an access control policy acp, we use subjectspec (acp), document-spec (acp), priv (acp), and prop-opt (acp) to denote, respectively, the credential / Formally, a security automaton is defined by: a countable set Q of automaton states, a countable set Q 0 Q of initial automaton states, a countable set I of input symbols, and a / We adopt an existing numbering scheme, called the Dewey order for TIDs (Tatarinov et al. 2002), whose basic idea is coming from the Dewey Decimal Classification (DDC) in library / XML Access Control Using Static Analysis MAKOTO MURATA, AKIHIKO TOZAWA, MICHIHARU KUDO, and SATOSHI HADA IBM Tokyo Research Lab Access control policies for XML typically use / Thus, secure operating environments (SOE) have become a reality on client devices

Cluster 6 ($N_C = 3$) „Linux Security“

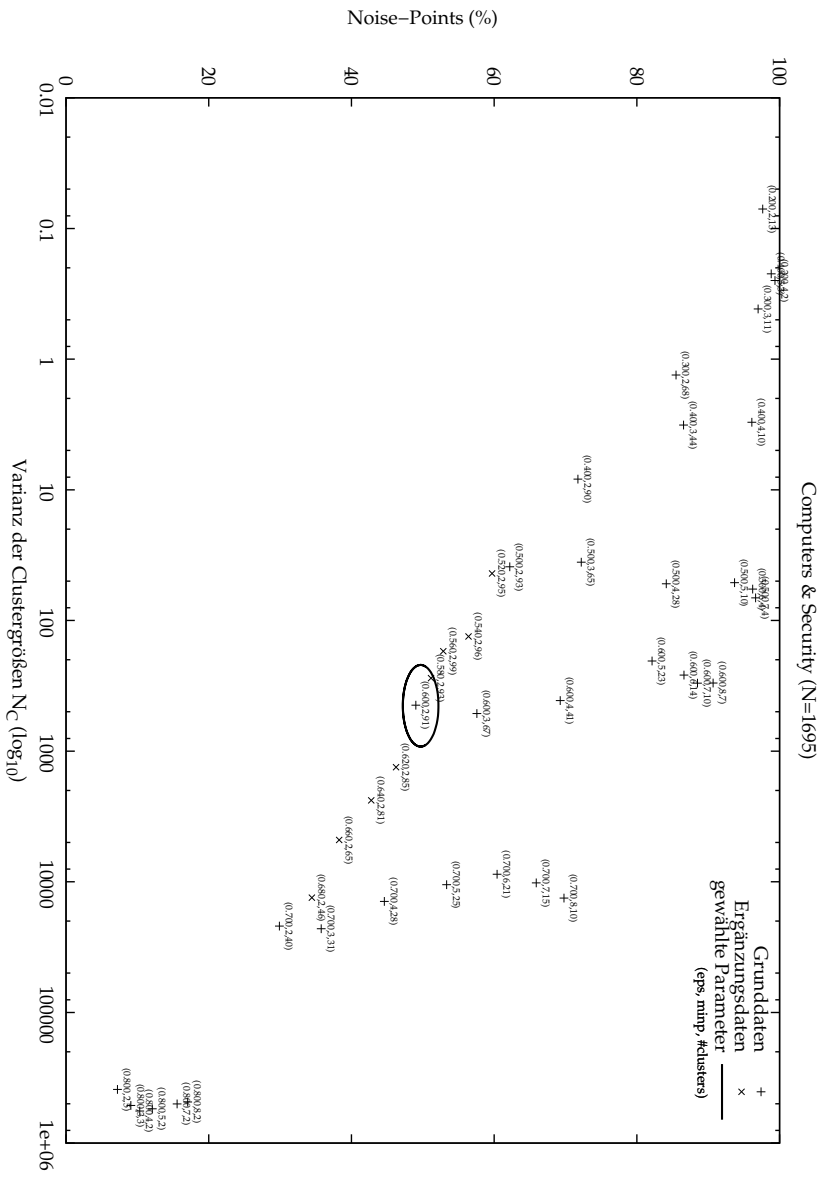
Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
bluebox kernel lsm acd hook remu file linux inod daemon jaba setuid uid modul pathnam execv fcntl placement chari bernaschi intraprocedur efid cheng socket rule sched overflow object euid invoc tocttou entri apach intrus attack namei log proc cont ino path suidp aclm privileg script subvert passwd gid patch shell static engler handler lst stackguard stackshield threat jaeger invok devic	BlueBoX: A Policy-Driven, Host-Based Intrusion Detection System / Consistency Analysis of Authorization Hook Placement in the Linux Security Modules Framework	BlueBoX: A Policy-Driven, Host-Based Intrusion Detection System SURESH N. CHARI and PAU-CHEN CHENG IBM Thomas J. Watson Research Center Detecting attacks against systems has, in / In this paper, we describe our experiences with building BlueBox, a host-based intrusion detection system. / We describe the motivation and rationale behind BlueBox, its design, implementation on Linux, and how it relates to prior work on detecting and preventing intrusions on host / Basically, the system call execution is allowed only in the case where the invoking process and the value of the arguments comply with the rules kept in an access control database / (2000) only a kernel patch implementation which required a simpler analysis and implementation was described, and 4. the access to the ACD by means of

Cluster 7 ($N_C = 3$) „SSH Security“

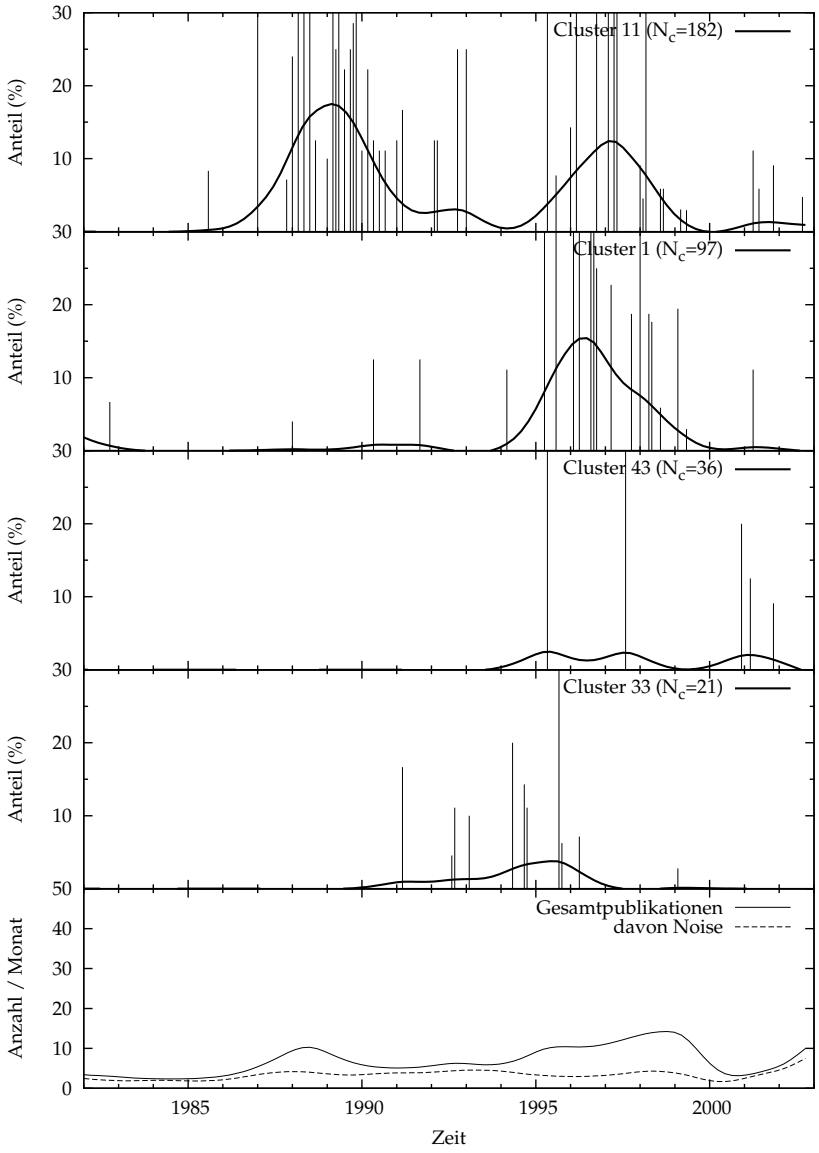
Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
ssh cca ocb ind ciphertext game pke adv adversari encrypt hatwideb rtr cbc cpa coll dpr ctr rogaway oracl bdhp ctxt decrypt ibe bellar sfctxt bpp prx npc cma cipher queri jutla kem pkae bilinear exp provabl block ptxt skie mode encod mac plaintext pad kiltz nonc rpt ruf ntz sfcca otru hatwidebp kie advind galindo cheon cmcoll mcoll offset	Breaking and Provably Repairing the SSH Authenticated Encryption Scheme: A Case Study of the Encode-then-Encrypt-and-MAC Paradigm / Provably Secure Timed-Release Public Key Encryption / OCB: A Block-Cipher Mode of Operation for Efficient Authenticated Encryption	Breaking and Provably Repairing the SSH Authenticated Encryption Scheme: A Case Study of the Encode-then-Encrypt-and-MAC Paradigm MIHIR BELLARE and TADAYOSHI KOHNO University of / Unfortunately, the current SSH authenticated encryption mechanism is insecure. / In this paper, we propose several fixes to the SSH protocol and, using techniques from modern cryptography, we prove that our modified versions of SSH meet strong new / OCB: A Block-Cipher Mode of Operation for Efficient Authenticated Encryption PHILLIP ROGAWAY University of California at Davis and Chiang Mai University MIHIR BELLARE University / OCB encrypts-and-authenticates a nonempty string M 2f0, 1g usingdjMjneC2 block-cipher invocations, where n is the block length of the underlying block cipher. / OCB refines a scheme, IAPM,

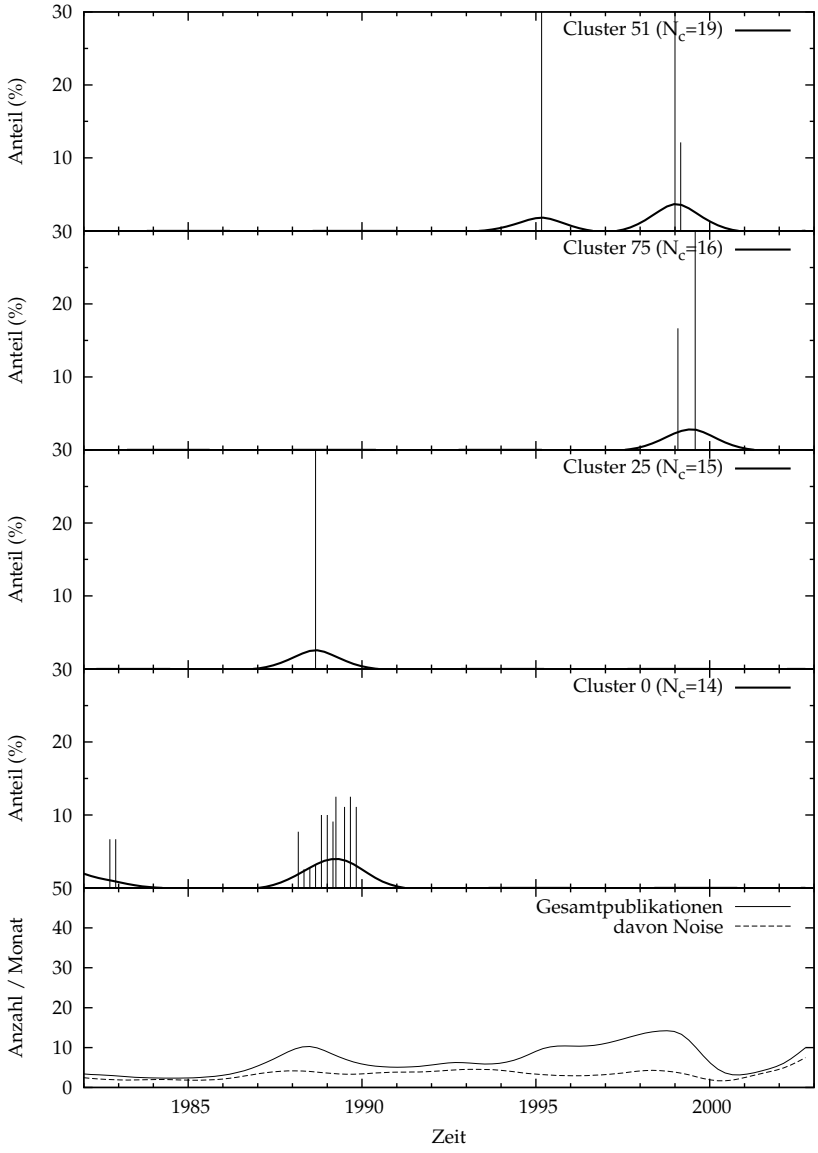
A.2 COMPUTERS & SECURITY

Cluster-Parametrisierung



Publikationsvolumen




Cluster 51 (N_c=19)
Cluster 75 (N_c=16)
Cluster 25 (N_c=15)
Cluster 0 (N_c=14)
Anteil (%)
Gesamtpublikationen
davon Noise
Anzahl / Monat
1985
1990
1995
2000
Zeit

Zusammenfassungen

Cluster 11 ($N_C = 182$) „Random Bits & Bytes“

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
viru disk virus infect macro firewal worm appl program file certif packet floppi server copi filter copyright scan boot backup router hard april client anti macintosh arc surfin microsoft ssl internet diskett byte window drive sector ram encrypt proxi februari hacker march exe disast microcomput password agenc directori web computerworld mail pgp news echo cluster command suit govern gateway interscan	Random bits bytes / Random bits bytes / Abstracts of recent articles and literature / Random bits bytes / Random bits bytes / Random bits bytes / Random bits bytes / Abstracts of recent articles and literature / Random bits bytes / Random bits bytes / Open systems security: Traps and pitfalls / Procedures to reduce the computer virus threat / Bugs in the web / Random bits bytes / Internet armour / Plan ahead for firewalls / Plugging the holes in host-based authentication / Lock IT up / Random bits bytes / Guidelines for the Protecting the Corporate against Viruses	Operating as a Web proxy, FinJans Surfin Gate 3.0 allays worries about Web security by offering server-side security for Java, ActiveX, Visual Basic Script, browser plug-ins and / B T echo off cls echo echo * ' * echo * This routine will copy hard disk recovery data from * echo * * echo * hard disks C, D, and H to floppy in drive A * echo * * echo * Use / Printed m Great Britain 1, 167-4048/98 19.00 Macro virus identification problems Vesselin Bontchev FRISK Software International, Postholf 7180, 127 Reykjavik, Iceland, E-mail: / SSL tries to reduce the overheads of key distribution and management. / The software industry is mixed on the issue, the basic poles of the dispute are represented by Apple and Lotus. / Computers Security, 11 (1992) 641-652 A Formal Definition of Computer Worms

Cluster 1 ($N_C = 97$) „Network & Web Security“

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
firewal java applet internet intranet viru web ipsec server atm disast pager activex vpn recoveri encrypt network satan pki certif mail hacker site virus netscap microsoft router file pgp ncsa sandbox esaf cert window compani tier lan rindo browser product digit plan anti scan infect password net authent onlin download key sweep kashpureff septemb novemb tester packet transarc client credit	Anti-virus belongs under the rug / Land and teardrop protection / Sun changes security framework for Java development kit / Win 95 security measures mollify users / Intranet firewalls offer no protection against the enemy within / Server -based Java security products help guard your enterprise flank / DE-light shadowed by security concerns / How to protect your data / Privacy issues stir online passions / S/MIME and OpenPGP vie for security title / Psst wanna buy some data? / Watching what you watch / Net security reawakening / Cylink, GTE target ATM encryption / Cheyenne, McAfee cure software viruses / Getting real about virtual private data nets / SOCKS push / Inner security / Pstt! Security designed for your eyes only / Puffer 2.0 buys you some e-mail security via easy encryption	Java applets are potentially a serious security threat, and one which applies to any computer network with access to the Internet. / Once likely fraud or theft targets have been established, they can then fine-tune the original scanning program to record only messages sent to the pagers they identify Pager / 4 vices are beginning to find security shortcomings as they start to move their private implementations of ATM onto public networks. / LAN Times, March 18, 1996, p. 76. Uncle Sam has three-tier scheme for data security, Gary Anthes. / Computenuorld, September 18, 299.5, p. 24. SATAN and Courtney: a devil of a team, Sean Gonxcilez. / Cowptrtcv Weekly, June 11, 1998, p. 6. VPNs now can link to two-way pagers securely, Sdvatorc Salanz rzc. V-One Corp. will give IT managers a way to

Cluster 43 ($N_C = 36$) „Audits & Wireless Security“

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
titl wireless stream audit author internet paper sherwood session window corum mcnulti fraud busi smf acf present privaci zergo solut hoath ssp koehntopp network connect cryptograph infrastructur discuss control escrow warfar multimedia lan medii unix disast backup outsourc role smart relianc msp threat ibm staff skill mulhal environ polici card compsec nice migrat risk phreak instinct atm commission commerc trail	The spook solution Now open for business / Securing Windows NT in today's network environment / Developing legal risks in multimedia / Addressing your information security skills shortages / COBIT Audit guidance on effective implementation / Automated audit Tools techniques / Smart cards and biometrics: An overview of current technologies, threats and opportunities / Securing third party connections / Internal vs. external IT audits or Mapping out a war zone? / NT server security, audit and control / Standards The need for a common framework / Penetration testing and system audit Experience gained and lessons learned during the investigation of systems within the United Kingdom / Project Satan / Unix security / Information flow within the globally connected environment / Evaluation of	That said, it is probably natural for me to become addicted to wireless technology. / The Wireless Great Divide But Getting Smaller Organizations have high hopes for wireless commerce. / I soon had it running all my favorite wireless cards. A PCCARD version of my Ricochet wireless service (which provides a nice 128kbps connection at 70mph in my car and with / COMPSEC 95 Paper Abstracts Title: Object Oriented: Another Silver Bullet Author: PJ. / Title: Wireless Network Security Author: Charles Cresson Wood, Information Integrity Investments Before your organization implements wireless such as wireless LANs, paging / COMPSEC 97 Paper Abstracts In the main, Information Technology (IT) security within the UK has been achieved as a result of the production of a System Security Policy (SSP) or

Cluster 33 ($N_C = 21$) „ITSEC & Security Evaluation“

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
oracl itsec evalu safenet criteria toe caast certif firewal fraud clef crime pankau performix internet polic fingerprint pirat ire visag server theft counterfeit product piraci raid bsa sentri tcsec window web wilson target sec browser sharp allianc card airway novemb computenuorld mci icsa databas thiev mercuri child client porn mail fledgl class octob inventor assur yard standard sponsor premis chip	Pressure builds on chip thieves / Top cop says fingerprint system is fraud-beater / More security woes plague the Internet / MCII offers secure transaction service / BSA toughens up on pirates / Fairwall certification / An evaluation of HP-UX (UNIX) for database protection using the European ITSEC / How secure is security? / Faced with a password solution / Software pirates treasure Win 95 / Computer security evaluation: Developments in the European ITSEC programme / Security evaluation criteria / Information security management: The second generation / The harmonized ITSEC evaluation criteria / Sentry cuts access to naughty bits / Distributed database security / Old viruses licked, but new ones find fodder in Windows 95 / Alarm at increase in civil service crime / Security evaluation in	Oracle extinguishes Net security fears, Karen Rodriguez. / Computers Security, 13 (1994) 547-557 Distributed Database Security Duncan Harris and David Sidwell Oracle Corporation UK, The Oracle Centre, The Ring, Bracknell, Berkshire, RGl2 / Computers Security, 10 (1991) 101-l 10 The Harmonized ITSEC Evaluation Criteria Karl Rihaczek 1. / Canadas computing privacy watchdog, the Canadian Alliance Against Software Theft (CAAST), has made its first surprise raids on companies suspected of internal sotiware copyright / US security expert, Ed Pankau, has urged users to adopt a radical safeguard and disconnect their PCs and modems when the office is empty This suggestion from Pankau follows / IREs SafeNet family of products consists of three components: SafeNet LAN, an encrypting firewall,

Cluster 51 ($N_C = 19$) „International Hacker Activities“

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
mitnick sock fingerprint bof takizawa firewal viru hacker internet orific randolph byer symantec trial spa antiviru shttp tracker interpol announc plea anti mandelson asl hamburg infect packet licens tap decemb weekli tunnel april flight wit ibm arrest agent defenc motion count compaq islat encrypt network survey schindler cellular worldcorn cnn nfr api norton polic handheld escrow filter spyglass ubc postpon	Active infections attack / Passwords could be past tense by 2002 / International agreement pees encryption at 64 / World wide web abusive use widespread / Compaq points fingerprints at security hole / Hacker-trackers hunt electronic cybercrooks / Virus threat gains momentum / Doors open to simpler VPNs / Hacker trial postponed / Interpol calls for key recovery / Kevin Mitnick finally gets a plea and a sentence / New computer system causes chaos / Users warned on CE security flaws / Will we all be secure in the new world? / Internet security boost / Interoperability drives security alliances / Stronghold of security / Hot on the hackers trail / One in three local authorities in the UK vulnerable to privacy cyberattacks	Kevin Mitnick Finally and a Sentence Gets a Plea A federal judge approved the guilty plea of computer hacker Kevin Mitnick in late March, ending one of the most high-profile / Kevin Mitnick Finally and a Sentence Gets a Plea A federal judge approved the guilty plea of computer hacker Kevin Mitnick in late March, ending one of the most high-profile / 3 Interpol Calls for Key Recovery Weighing in on the controversial subject of encryption controls, Interpols Hiroaki Takizawa today said the global police agency favors a way for / Secure Sockets (SOCKS) version 5 promises to bring a number of security and access-control improvements to the boundaries of corporate networks, including sessionlayer security, / A judge postponed for three months convicted hacker Kevin Mitnicks trial on federal

Cluster 75 ($N_C = 16$) „Cyberwar“

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
cisco falun chines taiwanes sra harden intrus gong lucent nrc dietz finalist decoy nist abilen vpn router site var internet gibraltar iss ccc exodu laci web camp trustworthi outsourc calif china torvald vranesevich mantrap taiwan hack sect sun allianc linux network mainland dougan huerta bachu research zboray reconqu bank august osag solari macro cyberwallplu acceler deliv ipsec sandon associ reiter	Hardening of an OS by putting it out for hack attacks / MS Office 2000 virus infestation Already / And if that is not enough Mainland China and Taiwan engage in cyberwar / Pessimism in cyberspace is alive and well / Making the operating system trusted A trend / China's internal war with the Falun Gong sect spills onto internet / Outsourced security The corporate information security burglar alarm hits IT / Security research alliance begins / Statistics show European security spending will quadruple by 2003 / Internet2 testbed going online at a network near you / It's vogue to capture hackers, not just watch / When Cisco talks about VPNs / Join chaos computer club for that summer retreat you have been longing to attend / Cisco announces security associate partnership programme / New IETF	Bill Hancock network core, the whole issue of how you handle differentiated services is a very hot topic right now, said Stephen Wolff, executive director of advanced Internet / The Abilene project, the key subnetwork that makes up the backbone for Internet2, will consist of a 13 OOO-mile, 2.4G-bps research network connecting 150 universities and three / Bill Hancock network core, the whole issue of how you handle differentiated services is a very hot topic right now, said Stephen Wolff, executive director of advanced Internet / Chinas Internal War With the Falun Gong Sect Spills Onto Internet The propaganda battle between the Chinese government and the founders of the Falun Gong meditation sect has / Computers Security, 18 (1999) 458-470 Security Views Dr. Bill Hancock, CISSP

Cluster 25 ($N_C = 15$) „Cryptography“

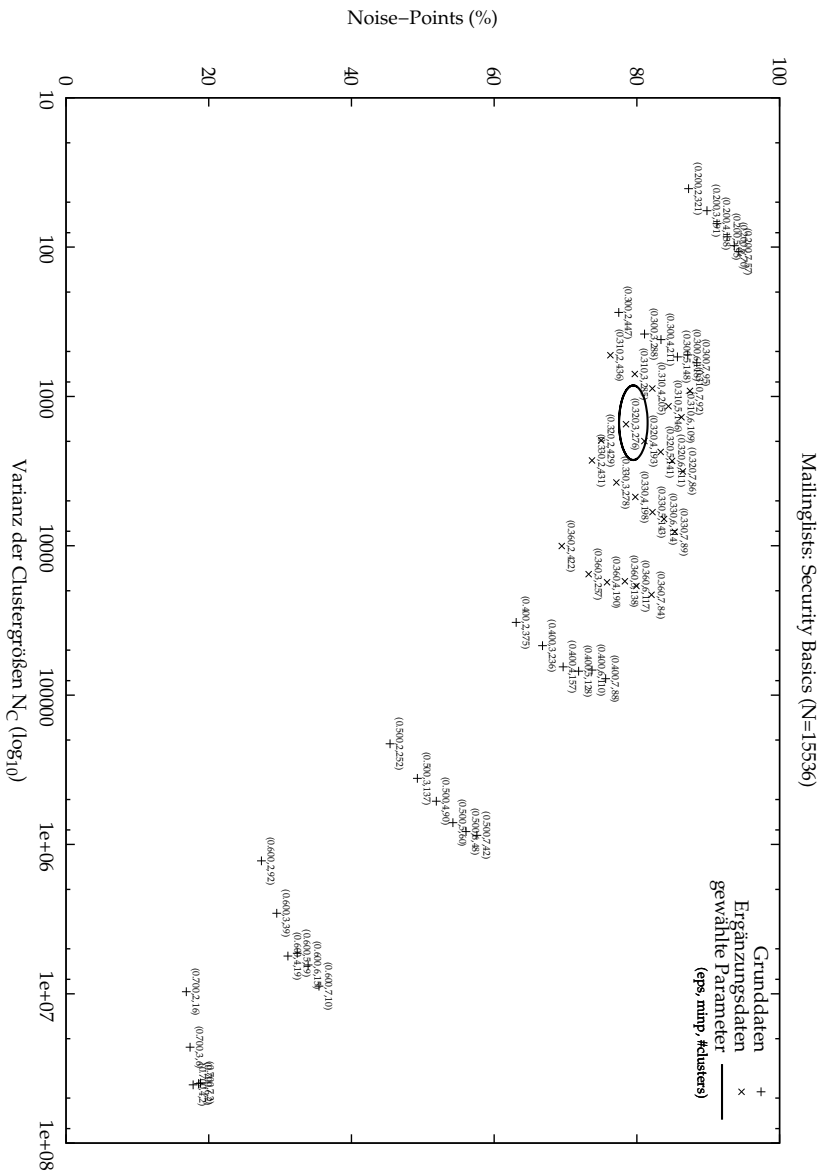
Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
wcc citat bibliographi worksit inspec noi aoi nti attitud topic gineer ramss databas softwar qualiti star intellectu reus assess cryptosystem hot estil peerci analysi spot physic congest risk manual maker health properti verif ofworksit citatiom prevel gies probert quindri sqa stof bsp baselin disast vlsi januari methodolog report doe unclassifi diagnosi gal ument exsi surement rsa video compress promot microcomput	Implementing the RSA cryptosystem / Understanding user attitudes: The cornerstone of computer documentation / Analysis of individual risk belief structures / Software Technology for Adaptable, Reliable Systems (STARS) technical program plan / Real-time fault detection and diagnosis: The use of learning expert systems to handle the timing of events / Software error experiment / DES (Data Encryption Standard) cryptographic services designed for the DOE wide band communications network / Information theory and public key cryptosystems / Expert system for software quality assurance / Data acquisition, reduction, and analysis using a microcomputer / Guidance on software package selection / Managing the data analysis process / Some complexity theory for cryptography / Microcomputer security	National Survey of Worksite Health Promotion Programs, 1985. / Computers Security, 7 (1988) 509-520 Special Abstracts Report: NTIS Computers, Control and Information Theory T he National Technical Information Center (NTIS) is a unit operating / N8816421/5/WCC. / PB /WCC. / January 1975-October 1986 (Citations from the INSPEC: Information Services for the Physics and Engineering Communities Database). / (NIN). TIB/B87–81813/WCC. / PB /WCC. / DE /WCC. PC A 03/MF A 01. / ADA 173 856/6/WCC PC AO3/MF AO1. / The reasons for this change of attitude are given (unemployment, media and associations of interested parties), information is given on information technology used at present,

Cluster 0 ($N_C = 14$) „Editorials“

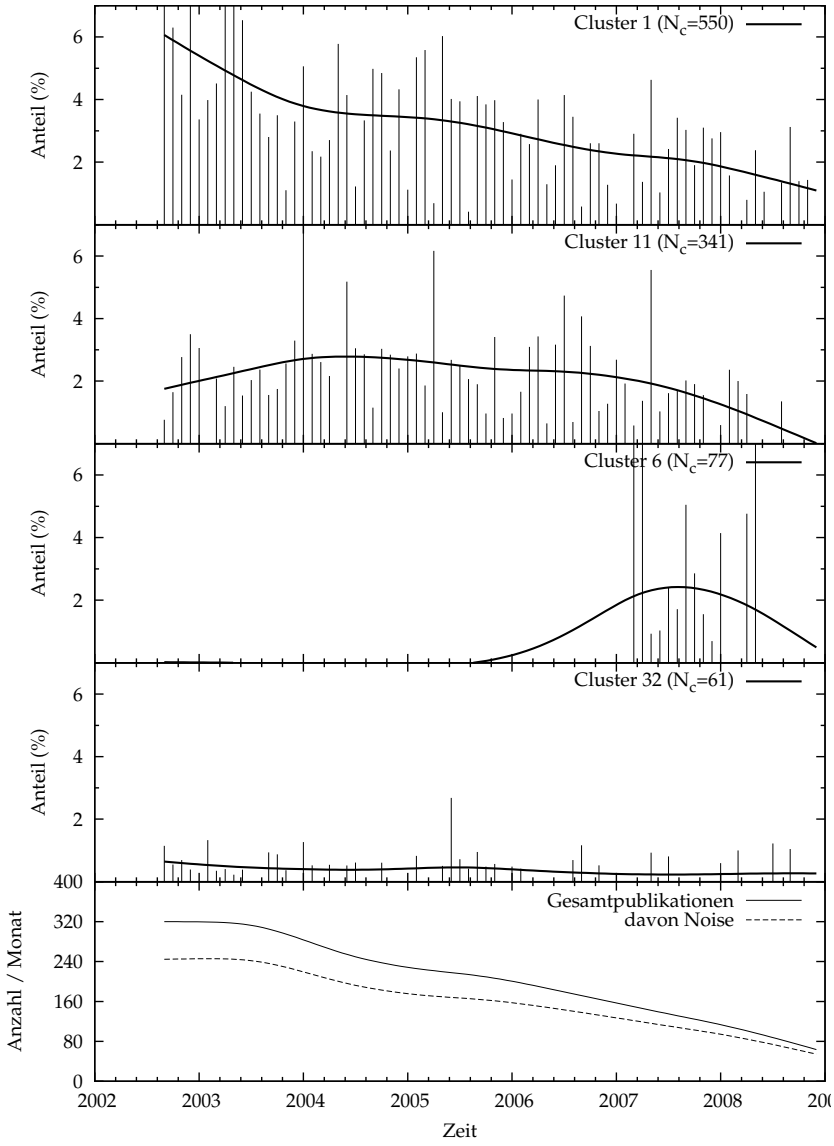
Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
editor highland william journal professor robert chief harold joseph editori coordin filter viru edror reader issu featur topic publish articl outstand virus paper rcvicwcr oxford croydon bologna board special elmont bailiff review cryptolog rcvicw univers inconceiv column profession fic rcwritc carrol york belden public menku senior murray rcferencc specialist arc arca serv soviet collcgc supplement elsevi road write justic campbel	From the editor / From the editor / From the editor / From the editor / From the editor / From the editor / From the editor / From the editor / From the editor / From the editor / Editorial / From the editor / Editorial / From the editor / Editor's note	Computers Security, 7 (1988) 228 Computers Security From the Editor Editor-in-Chief Distinguished Professor Emeritus State University of New York 562 Croydon Road Elmont, NY / Computers Security, 8 (1989) 2 Computers security From the Editor Editor-in-Chief Harold Joseph Highland, FIGS Distinguished Professor Emeritus SUNY 562 Croydon Road Elmont, NY, / Computers Security, 8 (1989) 86 Computers ecrity From the Editor Editor-in-Chief Harold Joseph Highland, FIGS Distinguished Professor Emeritus SUNY, 562 Croydon Road Elmont, NY, / In this issue we have collected several outstanding articles which cover various aspects of this topic. / Computers Security, 7 (1988) 114 Computers Security From the Editor Editor-in-Chief Distinguished Professor Emeritus State University of New York 562 Croydon

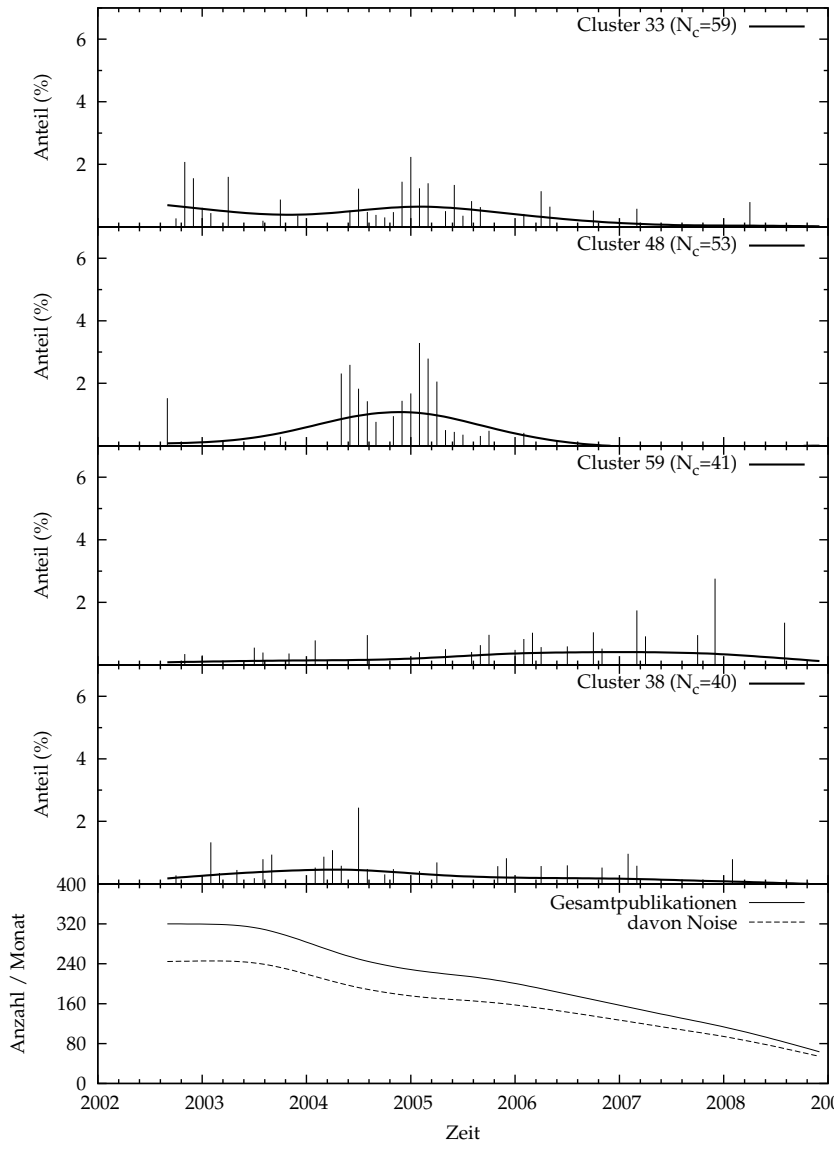
A.3 SECURITY-BASICS

Cluster-Parametrisierung



Publikationsvolumen




Cluster 33 (N_c=59)
Cluster 48 (N_c=53)
Cluster 59 (N_c=41)
Cluster 38 (N_c=40)
Anteil (%)
Gesamtpublikationen
davon Noise
Anzahl / Monat
Zeit
2002
2003
2004
2005
2006
2007
2008

Zusammenfassungen

Cluster 1 ($N_C = 550$) „Low Level Network Security“

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
port msn tcp firewal scan product server viru command window secur block network udp pix connect filter packet monitor site http netcat servic remov web protocol softwar run intern instal address level hotmail nmap access listen download exploit internet sourc client transfer zone netbio file activ unknown featur version list check machin base send joey establish queri refer vendor www	Server Compromised ? / Nmap Under the hood / Port TCP/8000 / nc help needed. / locking down my solaris box / Windows 2000 server ports, services to close. / Windows 2000 server ports, services to close. / TS Problems? / SV: Firewall Basics / Query: Filtered Ports I do not use. Should i be worried? / hacking games / TCP vs UDP / locking down my solaris box / What is this port? is it a trojan? / Remote Desktop vs VPN on Windows 2003 / TCP DNS requests / Open ports to establish a one-way trust / Very strange nmap scan results / Port TCP/8000 / Firewall and DMZ topology	Or by using the at command on the command line / PORT STATE SERVICE 25/tcp open smtp 53/tcp open domain 80/tcp open http 88/tcp open kerberos-sec 110/tcp open pop3 135/tcp open msrpc 139/tcp open netbios-ssn 143/tcp open imap / Does anyone have experience with this product? / Hello, i am using the GNU Version of netcat so i don’t know if it uses a different syntax than the “original“ netcat. / Both are W2k, and if I run ’netstat -an’ I get similar results: Server A Proto Direccin local Direccin remota Estado TCP 0.0.0.0:25 0.0.0.0:0 LISTENING TCP 0.0.0.0:80 0.0.0.0:0 / hello Is there a way to easily manage pix configurations ? / DNS uses UDP (or on some cases TCP). / May be kerio Firewall or winroute firewall / Filtering specifics instead of broad filtering. /) Keith Bucher HI, If

Cluster 11 ($N_C = 341$) „Passwords & Access Control in Networks“

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
password vpn cisco polici admin file horizonusa domain encrypt account solut log router ssl machin chang user set version copi connect server secur option window run network servic local administr access client product data instal firewal proxi remot mail check microsoft email requir port histori day book yahoo www recoveri peopl trust ssh multipl site time creat person authent home	: user default password checking tool / Minimum password requirements / Securing Service Accounts Good Practices / Account lockout / Minimum password requirements / Writing a comprehensive Network Policy / Linking Password Length to Write-down probability / A doable frequent password change policy? / Password changes more than once per day / NOC password management / Changing the domain password policy / Deploying SSL-based VPNs / passwords / VPN Client and Local Service / Terminal Services over VPN / Windows Service Accounts and Password Expiration / Password changes more than once per day / Restrict the Domain Admin / Two VPN clients on one computer / Seeking benchmark data on passwords	it seems like someone is trying to take the domain over. / VPN Encryption Static Route no Encryption. / Wow, i have a copy of the guide and it looks like there is A LOT of copy pasting here... / Upon opening the zip file there is no files in it. / You don’t need to remove the admin rights. / There is no problem with that, as long as your bank uses SSL over HTTP. / Citrix is a great solution. / There is a wipe device option in the Security Options off of the Options screen on the main page. / For windows 98 it is the TPF version 2 or 3 NOT 4. / I have never had this problem, but removing the workstation from the domain, and adding back onto the domain should fix it.

Cluster 6 ($N_C = 77$) „Audits, Compliance & Standards“

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
bdo kendal nsw craig wright attach sydney liabil legisl fax messag sender email copi commun inform file obscur test www scan risk surviv cost level secur standard pci port complianc statement control life profession loss vulner disclaim result econom modem limit attack gse partnership isc reli san audit respons nation defect asv peopl certif street experi endors model scheme read	How does a customer get PCI audited? / Concepts: Security and Obscurity / Concepts: Security and Obscurity / pen test / DSS (Passing an audit is NOT compliance!) / Nessus Scan / Getting the value of an asset and the probability of a risk to it / Web Application Vulnerability Scanner / Getting the value of an asset and the probability of a risk to it / Deny access to copy files / Concepts: Security and Obscurity / Firewall rulebase audit / How does a customer get PCI audited? / Getting the value of an asset and the probability of a risk to it / If somebody wants to sponsor it... / Database Encryption and PCI issue. / Preparing for GSEC / Concepts: Security and Obscurity / Concepts: Security and Obscurity / Concepts: Security and Obscurity	?ko, ‚Joyce. Soh bdo.com.au, ryan.pollett bdo.com.au, , ((“Joyce Soh“)), ((“Ryan Pollett“)), , Disclaiming uses custom rule where sender is Email domain bdo.com.au and / Craig Craig Wright Manager of Information Systems Direct : +61 2 Craig. Wright bdo.com.au +61 417 683 914 BDO Kendalls (NSW) Level 19, 2 Market Street Sydney NSW 2000 GPO BOX 2551 / Craig Wright Manager of Information Systems Direct +61 2 Craig. Wright bdo.com.au +61 417 683 914 BDO Kendalls (NSW) Level 19, 2 Market Street Sydney NSW 2000 GPO Box 2551 Sydney / Craig Craig Wright Manager of Information Systems Direct +61 2 Craig. Wright bdo.com.au +61 417 683 914 BDO Kendalls (NSW) Level 19, 2 Market Street Sydney NSW 2000 GPO Box 2551 / I could use more stupidly connected devices with added features that seem

Cluster 32 ($N_C = 61$) „Ethernet Security“

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
mac address comput mail compani server network tool chang modifi machin frame wireless host troubl pranav filter file lal list instal connect nic access accept secur port detect block url specif public protect larger won ethernet carri spoof relay paper switch send basic prohibit encrypt run scanner track implement firewal home app window extern client control softwar agent sticki exploit	Changing the mac address on Windows 2000 and XP / Changing the mac address on Windows 2000 and XP / Changing the mac address on Windows 2000 and XP / smtp relay tester? / Wireless security MAC Adress spoofing / MAC Authentication device / Wireless access / IP conflict and mac / Secure email? / WLAN Security, Authentication, and more... / dhcp / mac address / secure home network with entertainment center/filesharing / How to find a changing IP on ethernet network / prohibiting visitors from connecting to network / How to find a changing IP on ethernet network / Corrupt office, pdf, and other general use files / Alert when MAC address connects to network / Ethernet MAC address spoofing / Changing the mac address on Windows 2000 and XP / Using Web mail (hotmail, gmail, yahoo, etc) for	How about using a network sniffer to find out the MAC address? / Hi, Ive found which looks like a big bug this week-end on my computer during FTP file transfert. / I took this to help me creating the same in my company: / I know you can specify MAC addresses in those home DSL/Cable routers. / Well yes, the IP address inside the email header just points to the NAT address that is used for this particular location. / Seems to be a keylogger or something and is a computer behind the firewall that open it on the router when the computer start up. / Basics to defending against computer crime 2. / Go to ther router Find out PC’s MAC address via arp cache Go to the switch (s) Find out what port the MAC address is on (via CAM table) Trace cable Seva / If it’s wireless do what I did, use the MAC

Cluster 33 ($N_C = 59$)

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
yahoo mail spam harden search resourc mobil comparison host hack radiu code inform server plain port infocu tool scanner receiv gov infect network process servic www firewal app router site securityfocu linux command traffic setup enterpris machin user protect don fax box softwar origin time devic secur applic log window send run web track http detect access locat solut email	Finding person name based on personnel email account / Port 80 open without WebServer / All-In-One Printer/Fax/-Copier/Scanner Security / process accounting / Alerts of the ICMP relationship with smtp connection? / Seeking IIS v6 checklist and clarification on authentication / Network Traffic Monitor / Enterprise software for identifing user removing software / Antivirus Comparison / RealVNC Security / User Administration Security Database / Radius server sending mail alerts / Distressing, possibly life threatening emails from free accounts (yahoo, hotmail / Advise on Security audit tool / Open Ports on Cisco Router / RealVNC Security / Great Plains Segregation of Duties / changing routers and switchs passwords remotely / ystem information gathering / Tools for IIS security check	Do you Yahoo!? / Do You Yahoo!? / Thanks, Carv Do you Yahoo!? / Greetings All, I am trying to develop an AS400 (or would this be more properly titled OS400 hardening? / Do you Yahoo!? / Do you Yahoo!? / I sent the OP the code that I've used, but haven't heard back from him... Do you Yahoo!? / Gerson Sampaio Do you Yahoo!? / Do You Yahoo!? / Regards/-LINKCRAFT Do You Yahoo!?

Cluster 48 ($N_C = 53$) „Security Focus Column"

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
columnist securityfocu theft appl secur spywar solari microsoft linux white profil threat kernel factor explor minut free servic ethic sniff window phone busi privaci hous thread law anti sourc googl encrypt internet firewal control email inform time wireless peopl connect www	WIRELESS THEFT / SF new column announcement: Time to Dump IE / SF new column announcement: Linux Kernel Security is Lacking / SF new column announcement: Microsoft Anti-Spyware? / SF new column announcement: High Profile, Low Security / SF new column announcement: Windows Firewalls Lacking / SF new column announcement: Why Open Source Solaris? / SF new column announcement: Regaining control / SF new column announcement: Mac OS X ? Unix? Secure? / SF new column announcement: The Panacea of Information Security / SF new column announcement: Where's the threat? / SF new column announcement: Infected In Twenty Minutes / SF new column announcement: Online Theft / SF new column announcement: Microsoft Anti-Virus? / SF new column announcement: Cleaning Up Disclosure / SF new column announcement:	The following columnist commentary was published on SecurityFocus today: Security, 1994-2004: Then And Now By Daniel Hanson Oct 02:27PM PT Comparing the state of security in / I wouldn't go about calling it theft. / The following columnist commentary was published today on Symantec's SecurityFocus: Infection Vectors By Kelly Martin Mar 09:39AM PT It's time to pick your favorite virus. / The following columnist commentary was published on SecurityFocus today: Why Open Source Solaris? / The following columnist commentary was published on Symantec's SecurityFocus today: Practically Certified By Don Parker Mar 03:32PM PT Recent changes to the GIAC makes one / By Daniel Hanson Jul 11:45AM PT Apple's OS X is not safer or less susceptible to vulnerabilities and viruses than other OSes, and Apple'

Cluster 59 ($N_C = 41$) „Personal Certification“

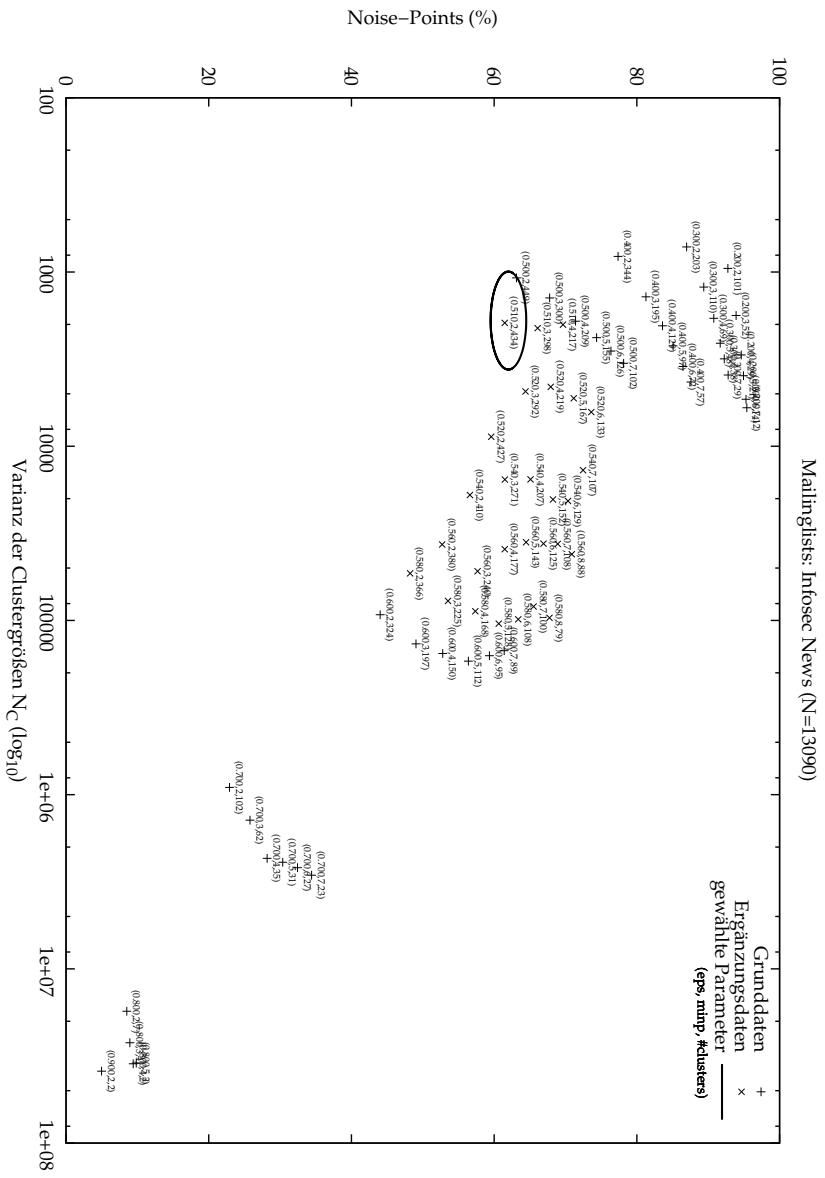
Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
certif book rememb expir cissp job user secur crl password window manag cryptographi applic rsa improv encrypt requir prompt pgp sign class archiv level mark server trust compani liabil public admin verisign mcse devic traffic suggest key test popup product start softwar remov hope www read expiri verifi visit warn understand recommend code file technic exam credibl person parti risk	CERTIFICATE / Value of certifications / Expired certificates / CERTIFICATE / Simple Effective Secure Email / Password creating Theories / Expired certificates / Crypto Book Recommendations? / OpenSSL question / good application security certification / Expired certificates / program to store passwords / CISSP Prep books? / CISSP / Which CERT to get / Restrict Access to Shared Folder with Encryption Key Rather than Password / Entry Level Certifications / University Degree or CISSP / CISSP Prep books? / Value of EC-CEH	That's exactly what I was trying to remember! / The value of a certificate or certification really depends on what you learn from it and not on the job that you get out of it. / For an update on the GIAC certifications, they are no longer requiring a practical assigment, this has now become just another exam-based certification. / Can't remember exactly where it is located but I remember is easy to find. / What do you guys know about the OPST certification? / I agree with Bob, to accompany that book get a book called the Code Book by Simon Singh as well. / I was just reading about CEH certification (Ethical Hacker) by ECCouncil, and I' d like to hear opinions about the certification, contents of the course, etc. I mean, its this / If you are interested in the OCTAVE approach there is a

Cluster 38 ($N_C = 40$) „Microsoft Security“

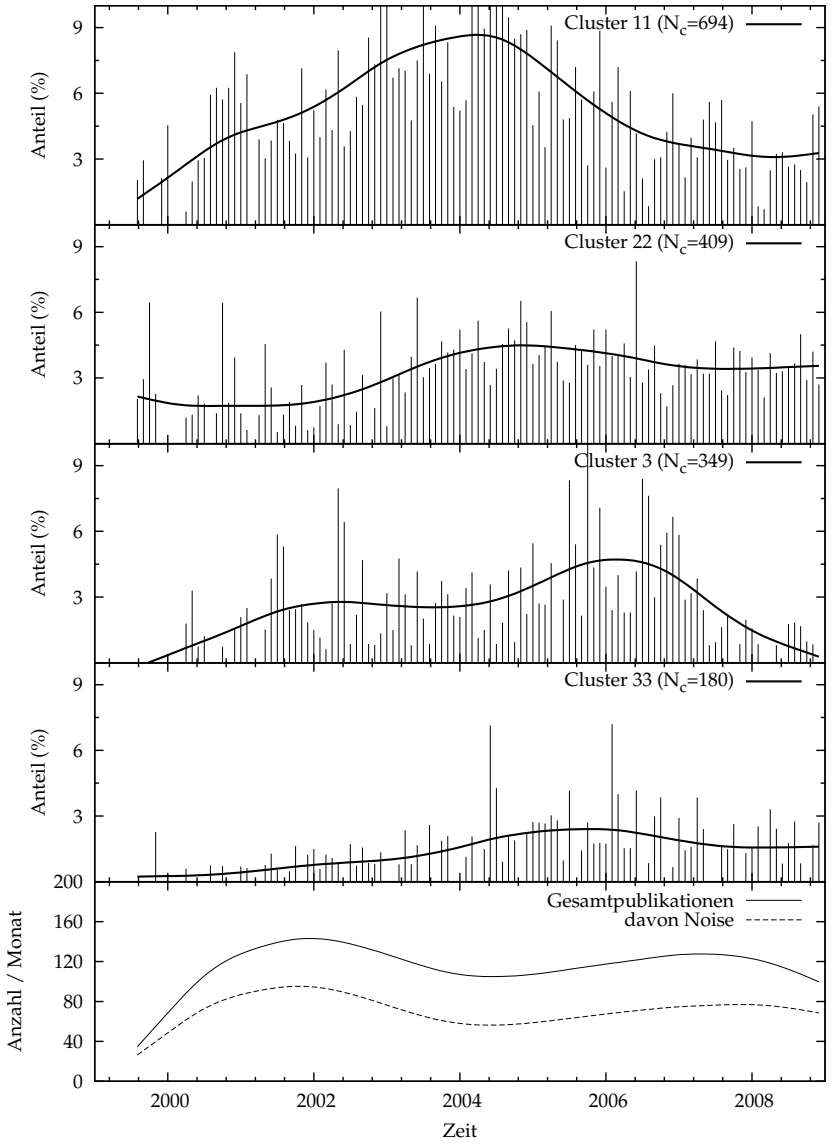
Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
updat patch server instal machin manag automat window client appli microsoft organ win servic run app test box task comput question current hfnetchkpro sequenc polici idea figur product approv configur site exe program port admin cost secur time set access don answer gerald deploy network browser sysadmin particip affect autom file msft list oper driver enforc compon softwar univers build	When is a Security patch not a patch? / Definitions / Windows patch mgmt. / Windows SUS / Administrator Rights? / Can I update a new patch for the whole windows computers of m / MS Service Packs / Can I update a new patch for the whole windows computers of my organization in my room? / Local Windows Update? / upgrading to IE6 on w2k servers / Windows SUS / Patching internet facing MS systems / IE5/IE6 Security Updates? / MS Service Packs / Trend Micro AV / Detecting File Alteration / Windows Server 2003 / IE5/IE6 Security Updates? / Deploying Microsoft patches / upgrading to IE6 on w2k servers	IE shows all updates when Windows Updates runs, and Automatic Updates can be configured to show or not show the specific updates. / But there's no alternative way to update Win boxes except using IE? / Windows Update is a service that can be configured via group policy to download ALL updates from an internal SUS server. / The users would not be able to do live update, but the server would push out updates. / We use WSUS for Windows server patching. / Same here haven't used the ISS, but I have no problem with auto updates, and Cisco is releasing signatures very quickly. / Automatic updates on a server is never a good idea. / I am having major problems with GFI patching.. just keeps crashing out half way through sending patches, dieing midway through scans etc. Having theis at every site I'

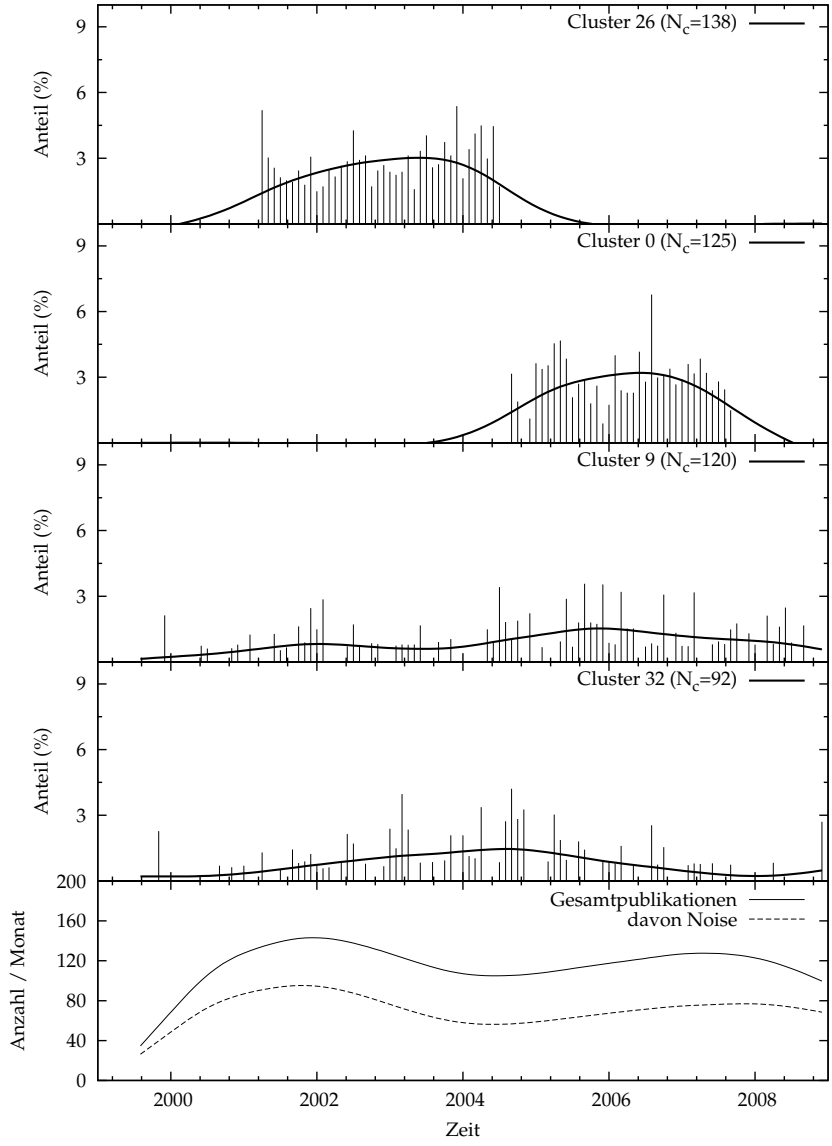
A.4 INFOSEC-NEWS

Cluster-Parametrisierung



Publikationsvolumen




Cluster 26 (N_c=138)
Cluster 0 (N_c=125)
Cluster 9 (N_c=120)
Cluster 32 (N_c=92)
Anteil (%)
Gesamtpublikationen
davon Noise
Anzahl / Monat
Zeit
2000
2002
2004
2006
2008

Zusammenfassungen

Cluster 11 ($N_C = 694$) „Linux Advisory Watch“

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
linuxsecur dave newslett linux advisori weekli chapter week rpm articl vulner watch includ rvw ftp book honeypot rslade secur spam password debian packag ssl freebsd vendor asin obido updat html honeynet exec apach amazon engard percent victoria niu soci checksum redhat dittrich cryptographi perl thawt slade mandrak red mozilla botnet caldera featur biometr server hipaa guid deb conectiva edit network	Linux Advisory Watch October 26th 2001 / Linux Advisory Watch March 15th 2002 / Linux Advisory Watch November 15th 2002 / Linux Advisory Watch February 22nd, 2002 / Linux Advisory Watch January 26th 2001 / Linux Advisory Watch January 11th 2002 / Linux Advisory Watch, Nov 24th 2000 / Linux Advisory Watch June 21st 2002 / Linux Advisory Watch March 8th 2002 / Linux Advisory Watch August 10th 2001 / Linux Advisory Watch October 19th, 2001 / Linux Advisory Watch February 2nd 2001 / Linux Advisory Watch, Nov 17th 2000 / Linux Advisory Watch January 19th 2001 / Linux Advisory Watch December 6th 2002 / Linux Advisory Watch December 14th 2001 / Linux Advisory Watch September 13th 2002 / Linux Advisory Watch December 14th 2001 / Linux Advisory Watch February 1st 2002 / Linux Advisory Watch August	Forwarded from: Lance Spitzner Last year I attempted to define and describe what honeypots are in the paper “Honeypots: Definitions and Values“. / Forwarded from: Dave Dittrich Lance points out a recent example this “expert“ missed, but they go back even farther. / By John Leyden 13 Sep 2007 Bastille Linux was forced to switch domain this week after a cybersquatter took control of the Bastille-Linux.org website. / Forwarded from: Thorsten Holz Greetings, The Honeynet Project and Research Alliance is excited to announce the release of a new paper “KYE: Tracking Botnets“. / Linux Security: Tips, Tricks, and Hackery / Published by Onsight, Inc. / 04-December-2003 / This issue sponsored by LinuxQuestions.org. / Linux Security: Tips, Tricks, and Hackery 08-June-2004 / Published by

Cluster 22 ($N_C = 409$) „Secunia Weekly Summary“

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
secunia advisori vulner summari week word defac list content ferrel attrit updat nsi valid rferrel skype psi sourc inspector exploit goeth verifi robert riaa mirror texa allda overflow cert report buffer blog microsoft script databas site cross user explor quicktim inject window net softwar trial version patch malici read inform refer viru appl arbitrari product multipl scanner denial file websit	Secunia Weekly Summary Issue: 2005-27 / Secunia Weekly Summary Issue: 2006-5 / Secunia Weekly Summary Issue: 2006-1 / Secunia Weekly Summary Issue: 2005-50 / Secunia Weekly Summary Issue: 2005-35 / Secunia Weekly Summary Issue: 2005-29 / Secunia Weekly Summary Issue: 2004-7 / Secunia Weekly Summary Issue: 2004-8 / Secunia Weekly Summary Issue: 2004-10 / Secunia Weekly Summary Issue: 2005-7 / Secunia Weekly Summary Issue: 2006-9 / Secunia Weekly Summary Issue: 2003-48 / Secunia Weekly Summary Issue: 2005-37 / Secunia Weekly Summary Issue: 2003-50 / Secunia Weekly Summary Issue: 2005-52 / Secunia Weekly Summary Issue: 2005-32 / Secunia Weekly Summary Issue: 2004-14 / Secunia Weekly Summary Issue: 2004-12 / Secunia Weekly Summary Issue: 2004-11 / Secunia Weekly Summary Issue: 2004-9	Forwarded from: Jamie Gillespie Of course not, alldas.de changed over to alldas.org in the early part of this year. / The Secunia Weekly Advisory Summary 2003-10-16 2003-10-23 This week : 30 advisories 7 New Microsoft Security Bulletins Microsoft has released no less than 7 security bulletins for / The Secunia Weekly Advisory Summary 2005-04-07 2005-04-14 This week : 87 advisories Table of Contents: 1 Word From Secunia 2 This Week In Brief 3 This Weeks Top Ten Most Read / The Secunia Weekly Advisory Summary 2003-10-09 2003-10-16 This week : 36 advisories 7 New Microsoft Security Bulletins Microsoft has released no less than 7 security bulletins for / The Secunia Weekly Advisory Summary 2008-10-09 2008-10-16 This week: 77 advisories Table of Contents: 1 Word From Secunia 2 This Week In

Cluster 3 ($N_C = 349$) „Microsoft Security“

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
microsoft window patch eey worm vista symantec flaw vulner maiffret updat bulletin soni rootkit exploit server softwar code releas apach user attack fix toorcon compani sql version affect bug infect antiviru issu red secur thompson product rutkowska critic messeng tuesday instal wmf internet malici viru run servic zotob web file pack word machin explor research marc custom program advisori site	Wait for Windows patch opens attack window / Major graphics flaw threatens Windows PCs / Symantec flaw leaves opening for viruses / Microsoft fixes 14 flaws in biggest patch day since February / Unauthorized Patch For Microsoft WMF Bug Sparks Controversy / (ISN) / Update: Microsoft says 'wait for us' as WMF threat climbs / Microsoft Plugs Code Execution Holes on Patch Day / Microsoft patches three critical security problems / Microsoft patches IE, Word, Windows / Symantec under fire for bugs, flaws / Microsoft Releases New Batch of Patches / Did FBI Ignore Code Red Warning? / Microsoft Sets New Patch Record, Fixes 26 Flaws / 'Trustworthiness' still a goal for Microsoft / Hacker code could unleash Windows worm / Microsoft Patches 20 Security Vulnerabilities / You Call This Trustworthy	Forwarded by: Jonathan Rickman The Code Red hype must be finally dying out. / By Graeme Wearden and Dan Ilett October Stuart Okin, the public face of Microsoft UK's security work, has resigned from the software giant. / Thomas Roy Garner SETIPRIME / By Joris Evers IDG News Service 08/20/02 Many Web servers running Apache-SSL remain vulnerable to attacks, although a June security alert prompted administrators to patch standard / Just wanted to let you know that ToorCon's CFP is going to be closing and pre-registration will be increasing on September 9th. / The February 2005 issue of First Monday (volume 10, number 2) is now available at / By Jim Rapoza January 12, 2004 While the Blaster worm and Sobig virus garnered the lion's share of attention and fear last year, 2003 began with a worm

Cluster 33 ($N_C = 180$) „ITL Bulletin“

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
nist laptop grade agenc fisma student sha feder school ciso fip draft district inform donat guid standard omb system checklist algorithm fcw control hash requir public davi accredit piv guidanc guidelin secur gov manag report stolen depart percent assess computerworld csrc lost brodi comput govern configur gcn offici organ implement data document duncan complianc plan process offic test recommend improv	ITL Bulletin for December 2003 / ITL Bulletin for January 2006 / ITL Bulletin for August 2008 / ITL Bulletin for November 2005 / ITL Bulletin for December 2008 / ITL Bulletin for April 2004 / ITL Bulletin for September 2008 / ITL Bulletin for August 2005 / ITL Bulletin for March 2006 / ITL Bulletin for May 2005 / ITL Bulletin for February 2006 / ITL Bulletin for October 2008 / ITL Bulletin for April 2006 / ITL Bulletin for May 2006 / ITL Bulletin for September 2004 / ITL Bulletin for November 2003 / ITL Bulletin for January 2007 / ITL Bulletin for June 2004 / ITL Bulletin for March 2007 / ITL Bulletin for March 2005	We have all seen these A F type grades for various agencies over the years. / (...) UPDATE: July 15, 1:30 PM: Fixed the case name (I had conflated "Healthcare Advocates" with "Health Advocate"). / By David Perera April 14, 2005 Steve O'Keeffe is halting his efforts to promote a for-profit forum for government and private-sector chief information security officers (CISOs). / Forwarded from: Elizabeth Lennon NIST'S SECURITY CONFIGURATION CHECKLISTS PROGRAM FOR Shirley Radack, Editor Computer Security Division Information Technology Laboratory National / By John Leyden 19th August 2005 Crypto researchers have discovered a new, much faster, attack against the widely-used SHA-1 hashing algorithm. / The FBI's top hackers apparently have been unable to break Joseph Edward Duncan III's

Cluster 26 ($N_C = 138$) „Winnetmag Security Update“

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
winnetmag window cjgsh cbw secadministr magazin net microsoft cgi bin thread cfm mag email server list flo win articleid poll index product newslett instant howto panda updat forum listserv patch user viru featur hot free vulner url windowsitsecur directori exchang articl tkip solut faq sql web worm outlook firewal wireless spam administr isa buffer wep bvg tool shavlik folder ebook	Security UPDATE, May 1, 2002 / Security UPDATE, March 5, 2003 / Security UPDATE, July 10, 2002 / Security UPDATE, February 27, 2002 / Security UPDATE, February 13, 2002 / Windows .NET Magazine Security UPDATE–September 17, 2003 / Security UPDATE, July 24, 2002 / Windows .NET Magazine Security UPDATE–September 3, 2003 / Security UPDATE, September 26, 2001 / Security UPDATE, January 16, 2002 / Security UPDATE, October 2, 2002 / Security UPDATE, May 29, 2002 / Windows .NET Magazine Security UPDATE–August 20, 2003 / Security UPDATE, January 22, 2003 / Security UPDATE, October 10, 2001 / Security UPDATE, October 16, 2002 / Security UPDATE, November 20, 2002 / Security UPDATE, October 3, 2001 / Security UPDATE, December 12, 2001 / Security UPDATE, December 18, 2002	Initially, 802.11i will provide Temporal Key Integrity Protocol (TKIP) security that you can add to existing hardware with a firmware upgrade. / **** This Security Alert is brought to you by the Windows IT Security channel on the Windows 2000 Magazine Network **** Sponsored by ONE CHANNEL WORTH FLIPPING TO! / This Issue Sponsored By Ecora Software Security Administrator * In Focus: Evaluating Intrusion Prevention Systems * Security News and Features News: XP SP2 Training for Developers / This Issue Sponsored By Windows Scripting Solutions New Web Seminar–Preemptive Email Security: How Enterprise Rent-A-Car Eliminates Spam 1. / This Issue Sponsored By CipherTrust Windows Scripting Solutions 1. / This Issue Sponsored By Ecora Software Exchange Outlook Administrator 1. / This Issue

Cluster 0 ($N_C = 125$) „WindowsITPro Security Update“

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
windowsitpro ctl bbb list window pro product free browzar server newslett email solut microsoft sql featur learn updat vip url white tip paper blog vista manag seminar complianc subscrib tool hot web sponsor exchang firefox wpa browser advertis applic patch tor ece vulner resourc download spywar recoveri disk save kma mfyqv discoveri oracl applianc filter support fbaf wireless spam access	Security UPDATE -Tweaking Wi-Fi APs for Better Security -September 21, 2005 / Top 20 List of Vulnerabilities -November 23, 2005 / Security UPDATE -WPA2 and WSP IE for Windows XP SP2 -May 11, 2005 / (ISN)Security UPDATE -Browser History: What Happened? -April 27, 2005 / Symantec's New Internet Security Threat Report / More About OS Haste, BATV / Security UPDATE -Recipe for Disaster -December 21, 2005 / Two More Portable Anonymous Web Browsers / TCP/IP Changes in Windows Vista and Longhorn / Application and Host IDS Tools / Security UPDATE -Wipe Old Hard Disks Clean Reprise -April 20, 2005 / Security UPDATE -Reading EULAs Can Help Prevent Spyware Infiltration -September 28, 2005 / Security UPDATE -Browser Security, More About Security Through Obscurity -June 8, 2005 / Alternative Firmware	Forwarded from: Mark Edwards Browzar Bashing: Is It Warranted? / How about Browzar .. can I do that? / Forwarded with permission from: Security VISIT OUR SPONSORS, WHO BRING YOU FOR FREE: KVM Over IP for the Distributed IT Environment Federal Rules of Civil Procedure and Email / Combine Save Email Discovery and Compliance The Essential Guide to Infrastructure Consolidation FOCUS: Hamachi Cross-Platform VPN NEWS AND FEATURES GoDaddy.com Abuse Policy Takes / Forwarded with permission from: Security VISIT OUR SPONSORS, WHO BRING YOU FOR FREE: Risky Business: Managing Risk Through Security Keep Unsecured Machines Off Your Network / Symantec St. Bernard Software Websense What's in the June 2006 Issue of Windows IT Security Feature: Reaping the Benefits of WPA and PEAP Access Denied Toolbox:

Cluster 9 ($N_C = 120$) „Oracle Security“

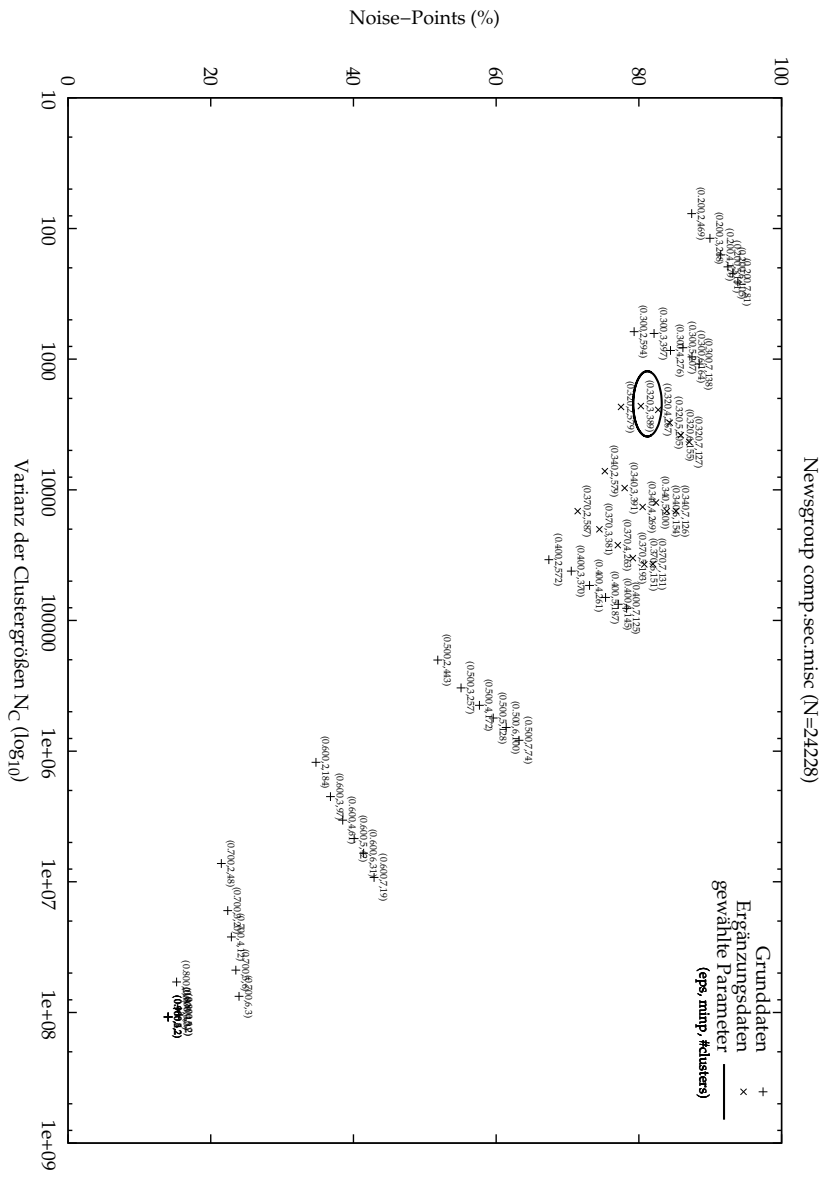
Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
oracl cent databas patch litchfield pgp survey fix flaw vulner davidson product ellison applic kornbrust softwar server bug sql updat compani wireless microsoft encrypt suit exploit secur attack releas vnunet busi user critic research custom unbreak password organis code magdych london enterpris peoplesoft data issu alert key respond inject cpu mobil network warn studi manag larri wile version worker chocol	Flaw hunters pick holes in Oracle patches / Oracle mends fences with security researchers / Security Firm: Oracle Opatch Leaves Firms Uncovered / Oracle aims to tone security muscle with Fusion / High bidders with low motives / Oracle denies researcher's security claims / Gadfly zeroes in on Oracle bugs / Oracle set to issue new bunch of patches / Security Researcher: Beware Dangling Cursors in Oracle Code / US government and security firms warn of critical Oracle flaws / Oracle plugs 101 security flaws / (ISN)Major Oracle Patch Covers Enterprise Products, Database Server / Oracle offering early warning on security fixes / Exploit circulating for newly patched Oracle bug / Oracle Patch Fixes 23 'Critical' Vulnerabilities / Oracle keeps many users waiting on April patches / Oracle Still	last time an Oracle database was broken into was 15 years ago, vs the 45 minutes he said it took for someone to break into Microsoft's first version of its Passport online / By Brian Prince eWEEK.com 2008-07-15 The Oracle Patch Update includes 45 security fixes, including 11 for the Oracle Database. / By Brian Prince eWEEK.com 2008-04-15 Oracle released fixes for a total of 41 bugs in its April Critical Patch Update, including a serious vulnerability affecting Oracle / By Brian Prince July 12, 2007 Oracle has plans to deliver 46 security fixes for its customers by July 17. / By Robert McMillan IDG news service 11 January 2008 Oracle is set to fix dozens of flaws in its software products, including critical bugs in the company's database, e-business / By Dennis Fisher December 4, 2002

Cluster 32 ($N_C = 92$) „Cybersecurity & Cyberwar“

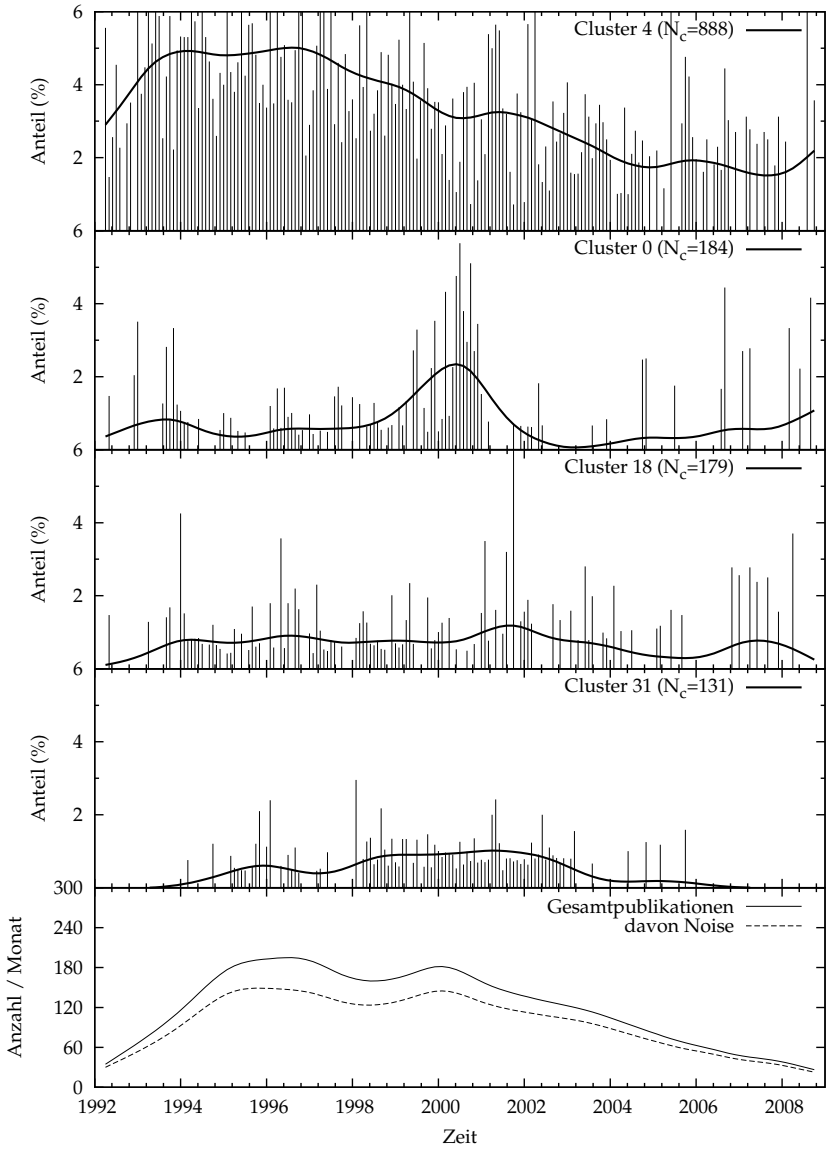
Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
cybersecur clark schmidt yoran gao homeland purdi govern nation tel sector depart fdic infrastructur agenc hous privat secretari bush critic cyber feder govnet presid white strategi cia technolog administr divis secur inform protect report intellig assist industri offici cyberspac director offic board coordin plan internet respons network share committe cyberattack budget appoint improv attack advis creat recommend ebay organ comput	U.S. cybersecurity due for FEMA-like calamity? / Anti-Terror Pioneer Turns In the Badge / Feds Falling Short on Cybersecurity / Clarke: homeland security revamp to help cybersecurity / Bush Approves Cybersecurity Strategy / Bush Needs To Ramp Up Cybersecurity In New Year / U.S. Cybersecurity Office May Relocate / Mission Possible / Howard Schmidt returns to DHS as USCERT head / Committee pushes for cybersecurity post / Cybersecurity expert warns of post-9/11 vulnerability / White House cyber czar describes next phase of Internet plan / Elimination of cybersecurity board concerns tech industry / GAO: DHS cybersecurity plans need more work / President's Top IT Security Adviser To Resign / White House Officials Debating Rules for Cyberwarfare / President's advisor predicts cyber-catastrophes	Donald "Andy" Purdy Jr. will step down as acting director of the National Cyber Security Division, part of the Department of Homeland Security. / Donald "Andy" Purdy Jr. has been acting director of the Homeland Security Department's National Cyber Security Division for 21 months. / By Terence O'Hara April 24, 2006 Amit Yoran resigned over the weekend as chief executive of In-Q-Tel, the venture capital arm of the U.S. spy community, after less than four / By Wade-Hahn Chan Sept. 1, 2006 The Government Accountability Office has released a new report that criticizes the Federal Deposit Insurance Corp.'s (FDIC) efforts to implement / By Diane Frank April 2, 2003 The National Institute of Standards and Technology's (NIST) Computer Security Division will be playing a significant role in the

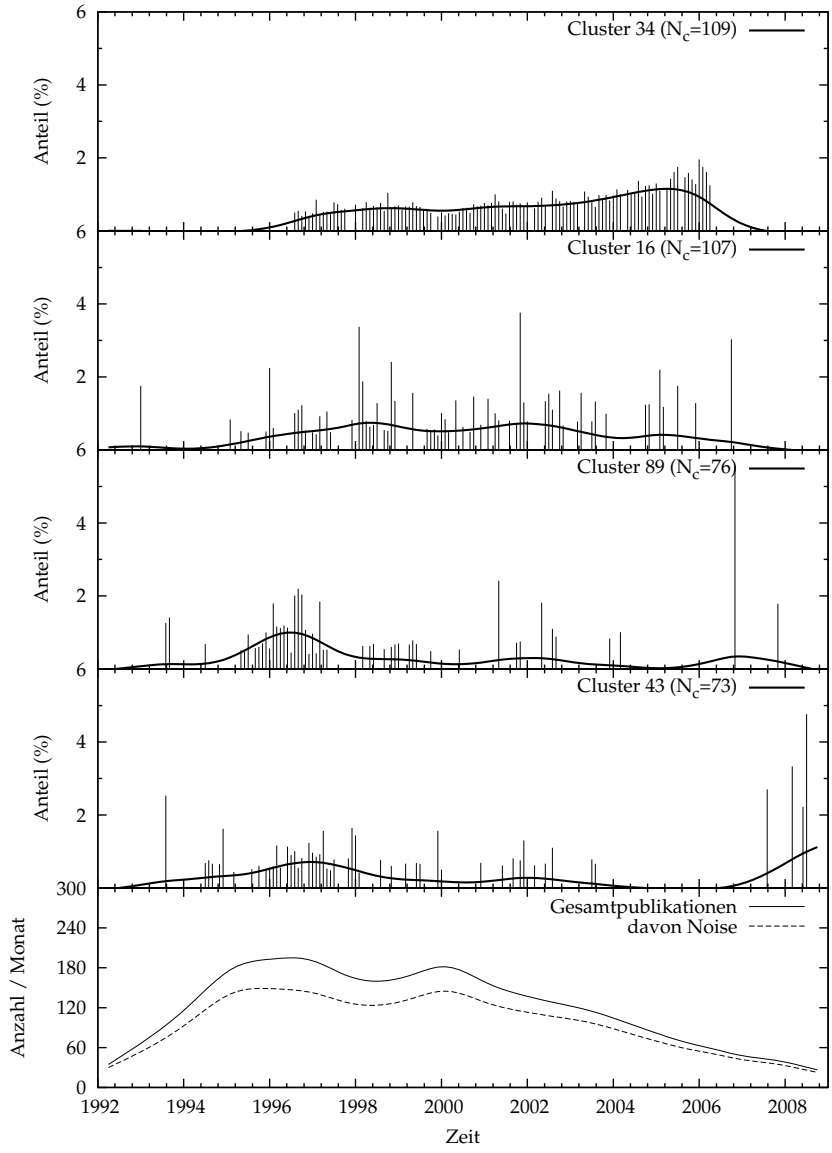
A.5 comp.sec.misc

Cluster-Parametrisierung



Publikationsvolumen




Cluster 34 (N_c=109)
Cluster 16 (N_c=107)
Cluster 89 (N_c=76)
Cluster 43 (N_c=73)
Anteil (%)
Gesamtpublikationen
davon Noise
Anzahl / Monat
Zeit
1992
1994
1996
1998
2000
2002
2004
2006
2008

Zusammenfassungen

Cluster 4 ($N_C = 888$) „Passwords & Cryptography“

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
password encrypt port pgp secur file program bio version key window inform softwar server hash crack free algorithm comput list databas user connect trojan packag product internet access report address email devic run excel word machin net command web chang packet batteri mail data udp jumper ftp bit motherboard host unix applic messag don rsa network attack www cmo sourc	Password Cracking / Stong Passowrds NOT! / Unreadable password files (was can people really have trouble memorizing long passwords?) / Strong Passwords Password Cracking (Final Version?) / Should a firewall ONLY allow access to an IP range as well as blocking ports? / Choosing secure passwords Feedback solicited / RESULT: comp.security.pgp. (announce, tech, discuss, resources) passes / Rational basis for password policy? / Password Cracking / Strong Passwords Revisited / Password Crackers / what are good passwords? / Securing a database / (Q) Why block incoming packets from low-numbered ports? / Password handling / Why unhashing is not possible? / Effectiveness of Forced Password Changing / Internet App Password Question / PGP 8.1 for Windows Mac / 2nd RFD: comp.security.pgp.* hierarchy	Hello, my name is Ulf, who can help me to calculate a simple number-password? / Hey, my name is Ulf, who can help me to calculate a simple number-password? / Please tell me mailling list address and how can i subscribe that list.. / I'm looking for a crack for Fortress 101 or someone who know how to crack it. / Hi, I want to secure my email and I don't know whether I can use the PGP , I come from China . / Hello all, Does anyone knows a e-mail software (POP) that have PGP Integrated (A windows front-end for PGP) ? / Lost my password. / I have been getting probes on our firewall port 41508 . / What NG would such problems or the first reports of a denial of service attack be first reported? / Best program and best software The software can scan and kill all the trojans, kill the virus ,

Cluster 0 ($N_C = 184$) „Web Security“

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
deja test mail implement java bit www info rpc mime post support http encrypt buy secur program applic email password server data newbi site alex html version url set certif suggest win port user crypto firewal decis faq network releas public machin trust rsa connect download format request check link train hardwar engin client protect address servic access ssl login	Encrypting RPC with SSL / Port 111 / Windows 98 Disable Cancel Button in Login Session / Advise: Choosing S/MIME solution / Netscape Security Error? Help!!! / Article:Usnet User fined-didn't post source / http to https posted data lost ! why ? / tcp port 158 / Article: Three arrested in first internet bank robbery / test / Network Security Training / security patch / Encrypting Credit Card numbers / security suggestion for online document system / Password recovery for Microsoft Access 8.0 / Internet security / email security / Java Random Numbers / Trojan Horses in Java / SSL co-processors	sorry, test only. / test test received / alt.test Test / Testing, testing, ...I'm a newbie, this is a test. / Pls disregard Sent via Deja.com Before you buy. / test failed. / Does anyone know how to prevent a Java applet from getting your IP address, without disabling Java? / Use lpwa.com Sent via Deja.com Before you buy. / Get the scoop Sent via Deja.com Before you buy. / Sent via Deja.com Before you buy.

Cluster 18 ($N_C = 179$) „Public Key Infrastructures"

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
certif rsa cert ftp patent netscap client ssl key crypto server sign trust root advisori browser instal openssl user secur public encrypt privat domain site author pgp verifi authent verisign digit file web compromis list bit pub issu creat import inform mail gener expir email date valid tool draper institut sender chain signatur filter comput certifi respons code messag pki	What is a Certificate? / Invalid certificate on 'security' site. / CA Certificate Built Into Browser Confuse Me / Is VeriSign lying??? / What is CERTIFICATE / RFI: Is there a net site with background info on FORTEZZA? / Issuing certificate / Server authentication / P2P Authentication / PKI / CA -Public Key Private Key / CERT Address/Phone/Email / SSL Certificate Self-Generated vs 3rd Party ? / Private key / Signatures and Certificates / private key in PKI / Security Sites / Which browser is best? / SSL certificate modification / Microsoft's Certificate Server Problem Solving S/Mime setup / Strange SSL Issue	Hi, Has anybody programmed using ' Wei Dai's Crypto++ ' crypto-library? / My server is not RSA-compatible. / Our RSA servers persistently stop authenticating and require a restart of the RSA service or reboot to start authentication again. / when i try to import a user cert and it's asssociated ca cert from a .uct file netscape can't find the ca cert . / I have a patent on "The Stealth Security Pad" and am looking to market it or sell the patent. / Well, according to Applied Cryptography by Bruce Schneier, there are two patents: In the US: Patent 4, 200, 770 In Canada: Patent 1, 121, 480 A group called Public Key Partners / Anybody know where I can get hold of the Netscape SSL reference implementation. / We are building an application where we need the certificates from various persons.

Cluster 31 ($N_C = 131$) „Firewall FAQ"

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
comp misc firewal post unix secur linux faq question common modifi file comput mail host url window network router attack traffic access bastion proxi servic server march win permit product internet screen alt rout cert level block setup packet ftp connect port ibm address web viru applic list user softwar game protect data intern machin infosystem januari tcp polici rule	Firewalls FAQ / Firewalls FAQ / Firewalls FAQ / Firewalls FAQ / Firewalls FAQ / Firewalls FAQ / Firewalls FAQ / Firewalls FAQ / Firewalls FAQ / Firewalls FAQ / Firewalls FAQ / Firewalls FAQ / Firewalls FAQ / Firewalls FAQ / Firewalls FAQ / Firewalls FAQ / Firewalls FAQ / Firewalls FAQ / Firewalls FAQ / Firewalls FAQ	Please keep this wash out of the linux groups. / Hi, does Linux have any shells which allow me to use xscan and xlswins commands ? / From Usenet newsgroups: at.mail.firewalls (Subscribe to at.mail.firewalls) (1) New! / THIS DOESN'T COMP.OS.LINUX.MISC or COMP.OS.LINUX.NETWORKING ignore it then, like you will this. / From Usenet newsgroups: at.mail.firewalls (Subscribe to at.mail.firewalls) (1) New! / From Usenet newsgroups: at.mail.firewalls (Subscribe to at.mail.firewalls) (1) New! / Where can I find comp.security.announce, comp.security.misc and comp.security.unix archived ??? / , comp.sys.amiga.advocacy, comp.sys.hp.hpux, comp.sys.ibm.pc.games.action, comp.sys.ibm.pc.games.strategic, alt.sex.anal, comp.sys.ibm.pc.hardware.chips, / As the comp.virus group is not active, this posting to

Cluster 34 ($N_C = 109$) „SSL FAQ“

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
consensu faq ssl talk corpor copyright develop cite modifi netscap author contain version pst accuraci electron list notic appel copi shannon socket reserv distribut layer form document right nov charg print post inform error purpos correct discuss secur suggest line www html mail includ comput txt ftp pdt sep dec meat pepper text grill sslref crock onion celeri skewer pot	(SSL-Talk List FAQ) Secure Sockets Layer Discussion List FAQ v1.1.1 / (SSL-Talk List FAQ) Secure Sockets Layer Discussion List FAQ v1.0.1 / (SSL-Talk List FAQ) Secure Sockets Layer Discussion List FAQ v1.1.1 / (SSL-Talk List FAQ) Secure Sockets Layer Discussion List FAQ v1.1.1 / (SSL-Talk List FAQ) Secure Sockets Layer Discussion List FAQ v1.1.1 / (SSL-Talk List FAQ) Secure Sockets Layer Discussion List FAQ v1.1.1 / (SSL-Talk List FAQ) Secure Sockets Layer Discussion List FAQ v1.1.1 / (SSL-Talk List FAQ) Secure Sockets Layer Discussion List FAQ v1.1.1 / (SSL-Talk List FAQ) Secure Sockets Layer Discussion List FAQ v1.1.1 / (SSL-Talk List FAQ) Secure Sockets Layer Discussion List FAQ v1.1.1 / (SSL-Talk List FAQ) Secure	Content-type: text/x-usenet-FAQ, version1.0.3, title“ (SSL-Talk List FAQ) Secure Sockets Layer Discussion List FAQ v1.0.3“ Archive-name: computer-security/ssl-talk-faq / Content-type: text/x-usenet-FAQ, version1.0.3, title“ (SSL-Talk List FAQ) Secure Sockets Layer Discussion List FAQ v1.0.3“ Archive-name: computer-security/ssl-talk-faq / Content-type: text/x-usenet-FAQ, version1.0.3, title“ (SSL-Talk List FAQ) Secure Sockets Layer Discussion List FAQ v1.0.3“ Archive-name: computer-security/ssl-talk-faq / Content-type: text/x-usenet-FAQ, version1.0.3, title“ (SSL-Talk List FAQ) Secure Sockets Layer Discussion List FAQ v1.0.3“ Archive-name: computer-security/ssl-talk-faq / Content-type: text/x-usenet-FAQ, version1.0.3, title“ (SSL-Talk List FAQ) Secure Sockets Layer Discussion List

Cluster 16 ($N_C = 107$) „Web-Proxy Security“

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
proxi ssl server tomcat lock apach connect client browser microsoft public firewal certif web secur deja mod exchang applic netscap port address access instal http user chain encrypt revers laptop code softwar support data tought url virtual modem java webproxi run list authent site hardwar bsafe cach provid set socket layer issu page trial owasp rsa protocol product tunnel wiltshir	SSL handshake question / Is HTML Get Method secure? / ISP DNS, proxies and security / SSL traffic via proxy server being ’intercepted’? / How to pass through a SSL connect by a http proxy? / Anonymous Proxy question / Multiple Proxy servers / Writing a secure server socket application / Multiple proxy Servers / Help with SSL More info / encrypted web page caching / SSL 3.0 Problem? / Problems running VeriSign trial sertificate in Tomcat. / Proxy, SSL, and CONNECT simple questions / Web Publishing And Access Control / Proxy 2.0 Authentication / Proxy Chaining???Question? / SSL can you insite on having certificate? / Sever connect problem / Tomcat secure configuration	What are the security implications / tradeoffs for using Tomcat’s built in web server (rather than running Tomcat standalone)? / But, should I add some SSL features to Tomcat as well? / Hello I’ve read somewhere that it is possible to use the attacks against SSL 2.0 in SSL 3.0 by using the “roll-back“ : compatibility between SSL 2.0 3.0 which can force an / is it possible in SSL for the receiver to reorder SSL record blocks that arrive out of order? / I think the first would be: don’t run Tomcat as root. / I want to know about software lock mechanism ! / I’m using Apache-Tomcat (sometimes just Tomcat standing alone). / Anyone have any experiences with proxy ftp and/or proxy telnet they could share with me? / Is there any difference between using a program, such as A4Proxy, as opposed to

Cluster 89 ($N_C = 76$) „SSH Security“

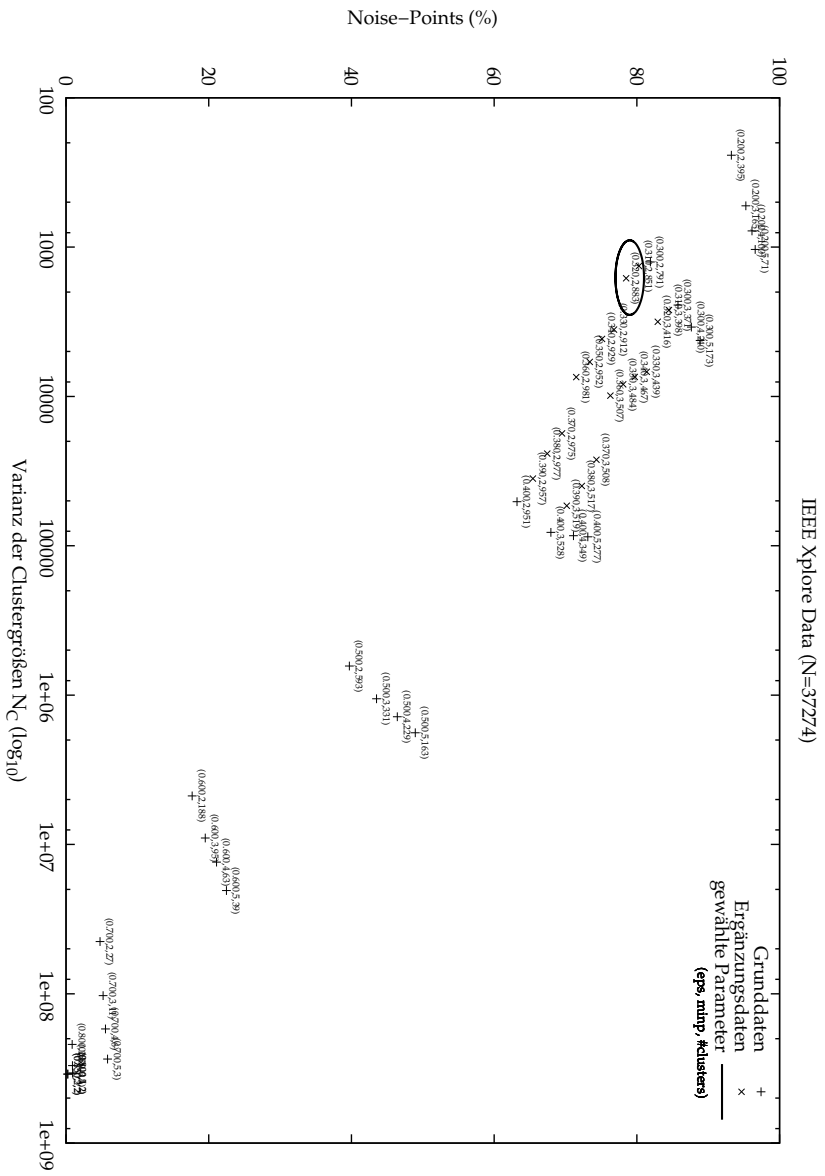
Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
ssh vote faq gss api ftp comp newsgroup secur sftp intranet discuss clinet datafellow unmoder telnet client cfv voter votetak scp tunnel mail firewal charter protocol post ssl port server infosystem rfd connect list propon abstain propos karlsruh announc macintosh product count crypto resourc usenet implement dogwood uni gateway issu window rcp rsh forward fellow line transfer resrocket redirect file	CFV: comp.security.ssh / 2nd CFV: comp.security.ssh / RESULT: comp.security.gss-api passes 159:21 / 2nd CFV: comp.infosystems.intranet / CFV: comp.infosystems.intranet / RFD: comp.security.ssh / 2nd CFV: comp.os.ms-windows.nt.admin.security / CFV: comp.os.ms-windows.nt.admin.security / RESULT: comp.security.ssh passes 642:11 / RFD: comp.security.gss-api / RFD: comp.infosystems.intranet / RESULT: comp.infosystems.intranet passes 158:22 / Secure file transfer / RFD: comp.security.firewalls / RESULT: sci.crypt.research moderated passes 348:23 / 2nd CFV: comp.security.firewalls / CFV: comp.security.firewalls / What is the difference between ftp encryption types SSL, TLS, SFTP and SSH ? / RESULT: comp.security.unix passes 590:17 / RESULT: comp.security.firewalls passes 647:88	yes, ssh-3.x just speeks SSH protocol v2. / GSS-API is not a protocol, it is an API. / or We have a SSH resource page on it gives several links for different OS's. / I've heard that a product which makes calls to GSS-API and uses dynamic linking for the GSS-lib is subject to the same export restrictions as cryptographic software. /) It appears that this is connected to a known ssh v1 issue described at If you have ssh v1 fallback enabled for your ssh v2 system, then you may well be vulnerable, as the ssh v2 / I'm thinking about proposing a Usenet newsgroup for discussion of the GSS-API. / Unmoderated group comp.security.unix Voting closed on 29 September 1993 at 23:59:59 GMT. / I've read teh RFCs, but they are (at least for me) quite unclear :- (the GSS API allows applications to call

Cluster 43 ($N_C = 73$) „Security Conferences“

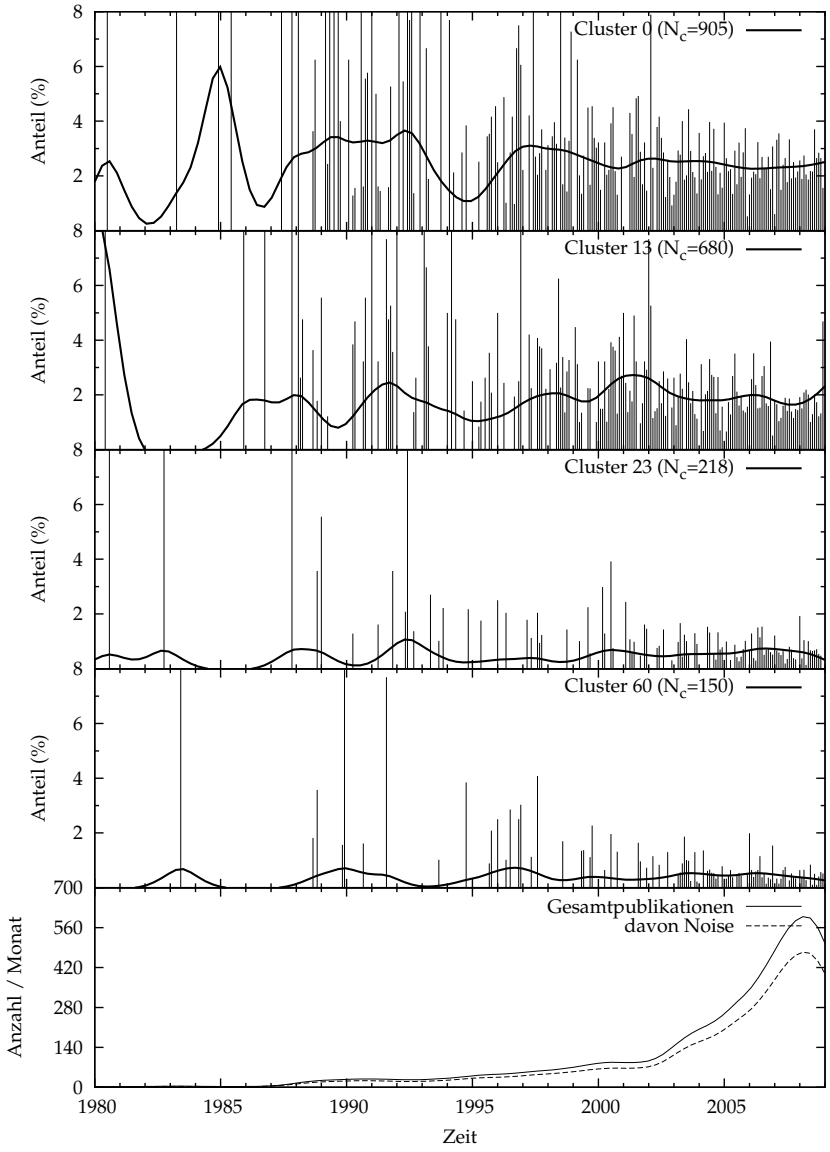
Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
usenix lisa paper confer symposium submiss administr refere tutori org invit session technic system submit abstract registr committe program present event secur attende associ octob extend author talk window email topic chair august sage accept fax propos bof advanc panel sponsor proceed sec workshop track comput materi regist phone hotel feather guidelin inform bird wednesday progress particip research act manag	FINAL CALL FOR PAPERS 10th Systems Admin. Conference (LISA'96) / CALL FOR PAPERS - LISA 1996 Conference (Abstract deadline May 7!!) / LISA '96 - Announcement Call for Participation / Submissions Due 5/7/96 10th USENIX Systems Administration Conf (LISA'96) / 10th USENIX Systems Administration Conference (LISA '96) / 7th USENIX Security Symposium Call for Papers / 7th USENIX Security Symposium Call for Papers / 11th SYSTEMS (LISA '97) Call For Papers / ANNOUNCEMENT AND CALL FOR Security Conference / 11th SYSTEMS (LISA '97) Call for Papers / 7th USENIX Security Symposium Call for Papers / 7th USENIX Security Symposium Call for Papers / 7th USENIX Security Symposium Call for Papers / Extended Abstracts due June 3 11th SYSTEMS (LISA '97) Call for	The USENIX Security Symposium is just around the corner. / System Administrators–Plan Ahead–Save these Dates December 6-11, SYSTEMS (LISA '98) Boston, Massachusetts Call for Participation now at: Sponsored by USENIX, The Advanced / I'm writing to remind you that the Early Bird Registration Deadline for the 17th USENIX Security Symposium is Monday, July 14, 2008. / Dear Colleague: You're invited to join us at the 17th USENIX Security Symposium, July 28-August 1, 2008, in San Jose, CA. USENIX Security '08 will help you stay ahead of the game / Sponsored by USENIX, The Advanced Computing Systems Association Co-Sponsored by SAGE, the System Administrators Guild SEEKING: Refereed Papers, Invited Talks, Works-in-Progress / 15th Systems Administration Conference (LISA 2001) December 2-7, 2001

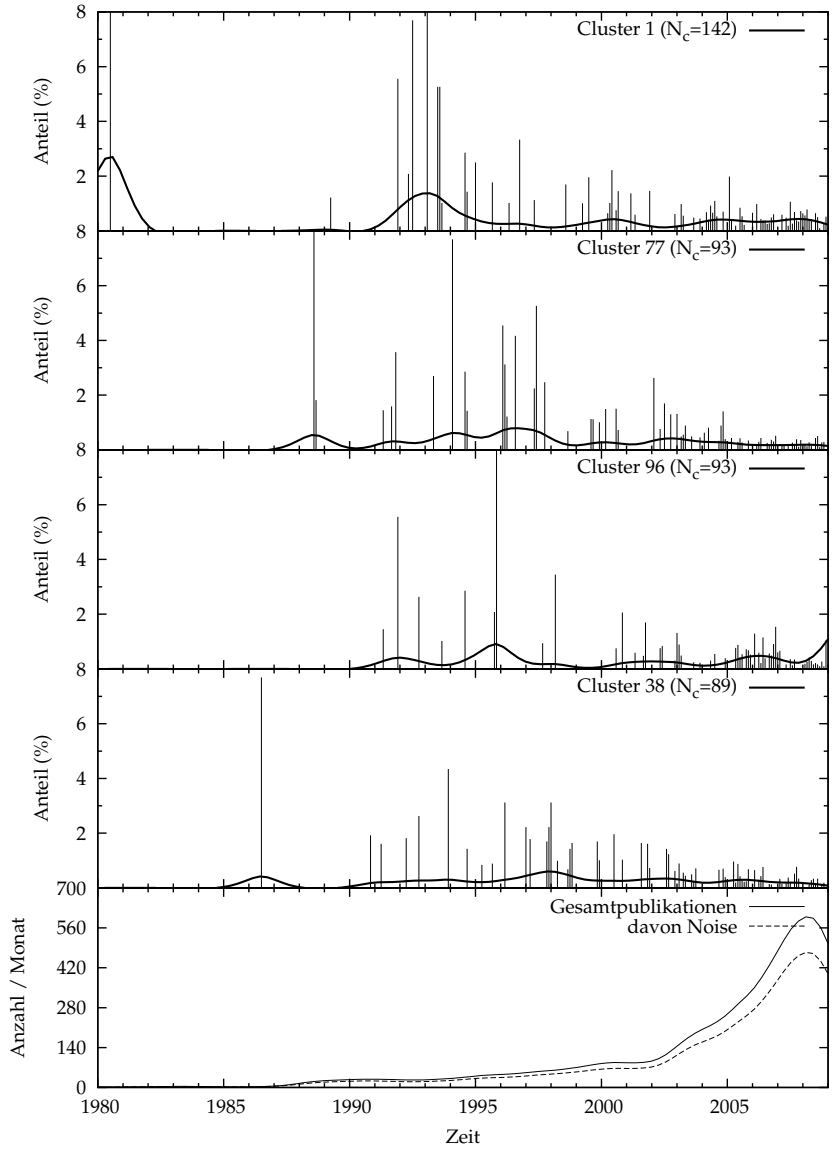
A.6 IEEE XPLORE

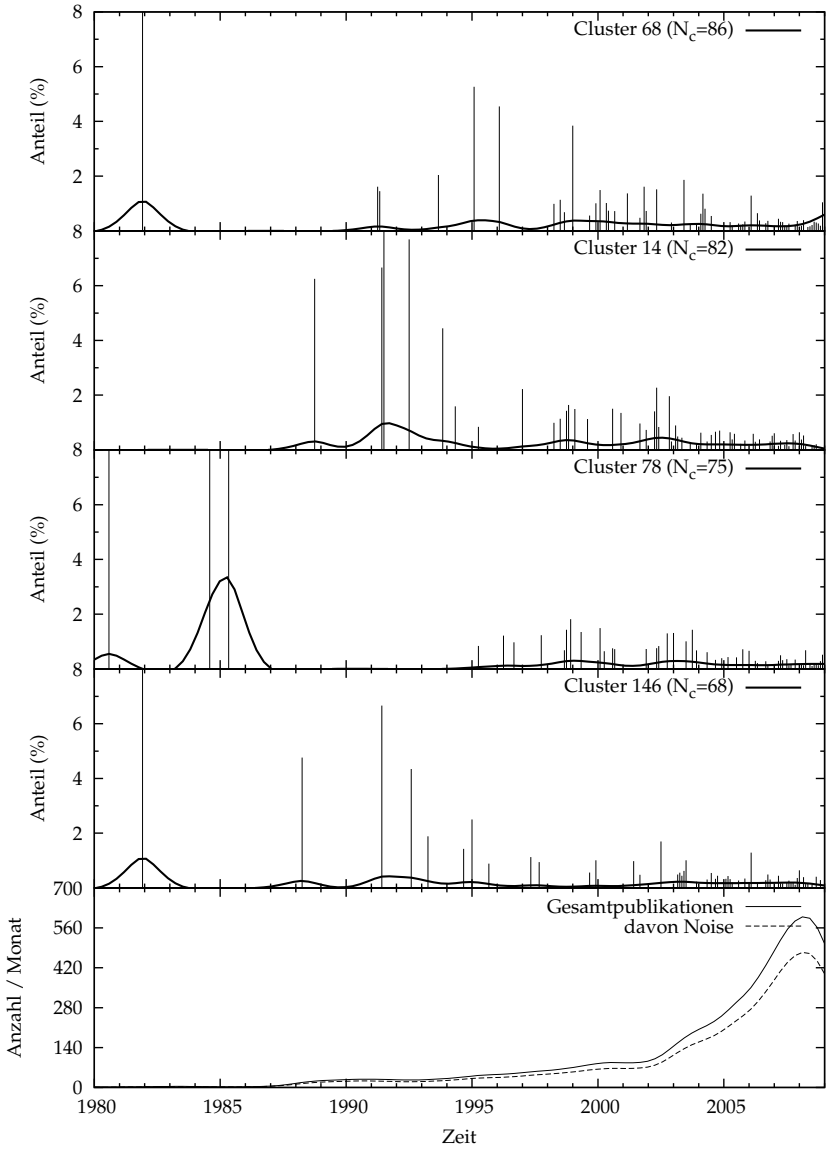
Cluster-Parametrisierung

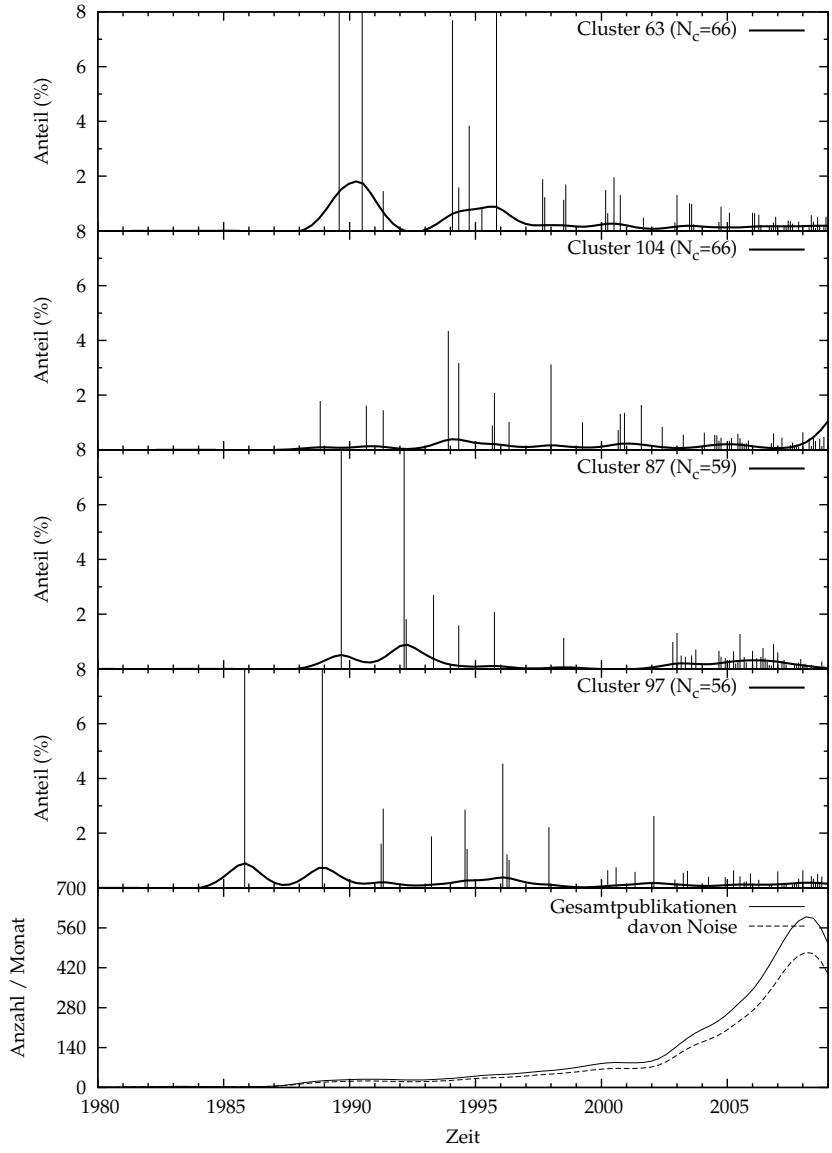


Publikationsvolumen




Cluster 1 (N_c=142)
Cluster 77 (N_c=93)
Cluster 96 (N_c=93)
Cluster 38 (N_c=89)
Anteil (%)
Gesamtpublikationen
davon Noise
Anzahl / Monat
Zeit
1980
1985
1990
1995
2000
2005


Cluster 68 (N_c=86)
Cluster 14 (N_c=82)
Cluster 78 (N_c=75)
Cluster 146 (N_c=68)
Anteil (%)
Gesamtpublikationen
davon Noise
Anzahl / Monat
Zeit
1980
1985
1990
1995
2000
2005


Cluster 63 (N_c=66)
Cluster 104 (N_c=66)
Cluster 87 (N_c=59)
Cluster 97 (N_c=56)
Anteil (%)
Gesamtpublikationen
davon Noise
Anzahl / Monat
Zeit
1980
1985
1990
1995
2000
2005

Zusammenfassungen

Cluster 0 ($N_C = 905$)

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
node watermark agent attack imag trust packet power sensor user protocol polici messag servic scheme network authent server signatur mobil encrypt detect rule file signal voltag load host flow card rout algorithm storag client manag session video devic intrus deleg secret fault grid block vehicl probabl embed router model code energi access content ofth request privaci prime frequenc proxi databas	Energy efficient watermarking on mobile devices using proxy-based partitioning / Analysis of payment transaction security in mobile commerce / Mitigating Denial-of-Service File Attacks on the Chord Overlay Network: A Location Hiding Approach / Comparison of security protocols in mobile wireless environments: tradeoffs between level of security obtained and battery life / Design issues in mobile agent programming systems / Private wars in secret life / eXpert-BSM: a host-based intrusion detection solution for Sun Solaris / Securing Medical Sensor Environments: The CodeBlue Framework Case / TVA: A DoS-Limiting Network Architecture / Anonymous connections and onion routing / An AAA Study for Service Provisioning in Vehicular Networks / The VersaKey framework: versatile group key management	But the four classes should be defined for the informations semanteme following the expressions according the BSCM. / A forged signature by the extension 1 (1152-bit, e 3) m pkcs-1v2-1.doc (8) H (m) s se 76c3 0000 B B e-th root computation over real number is easy. / To solve the problem, the paper proposes a microaggregation algorithm for sensitive attribute diversity l-MDAV algorithm. / The IP interconnect adapter of the Jini surrogate host created a process servicing the surrogate protocol, and managed the connection with the surrogate host and the BPLC home / "+, |, X+6EE 'U1u D ++, X+ 0L!!O6 , Xs)V / 4!O, X6s) PWM1, 2 DSP , X97 PWM (V cl , V c2 97PE|+C+ C+ V ref /2 H6+C, XK+ (!EW (05 +) DPWM1, DPWM2 4H6, X PWM "6 T d !OK ref 2) 2 ln (CC d CC V VV T ' refCCVV ln 2dT H6, X / , *

Cluster 13 ($N_C = 680$)

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
node imag sensor agent power attack packet network servic watermark trust messag detect protocol signatur authent server mobil polici user scheme rout signal voltag grid certif encrypt fault password code wireless object cluster flow camera path algorithm pattern detector access energi client manag channel simul load proxi model pixel block secret router session data ofth hash attribut privaci video secur	Security and Privacy for Distributed Multimedia Sensor Networks / Toward Secure Low Rate Wireless Personal Area Networks / JANUS: A Framework for Scalable and Secure Routing in Hybrid Wireless Networks / VPN Analysis and New Perspective for Securing Voice over VPN Networks / Dynamic PKI and secure tuplespaces for distributed coalitions / FIRE: flexible intra-AS routing environment / Security issues in on-demand grid and cluster computing / Applications of information networks / Securing VoIP and PSTN from integrated signaling network vulnerabilities / Civitas: Toward a Secure Voting System / Security implications of typical Grid Computing usage scenarios / Use of Presence and Location Information for Situational Awareness / Delegate: A Proxy Based Architecture for Secure Website Access	John M '239 Elliott. / May 30June 3, 2006 5 D A T uesdaySaturday , May 30June 3 34 fulland halfday tutorials Learn from e xpert instructors T opics range from Ajax to V o IP 3-D A Y TECHNICAL PROGRAM / Table 1: Collisions for 3-vass HAVAL (s31 xi (Xi) I Values I xi (X,) I Values x4 (X4) I I xg (Rg) 1 IV. / This route can be a path of communication between the ball and the outside of the rubble. / On the other hand, it is also possible to have nested certificate paths that can be verified more efficiently. / The trusted degree of the normal orientation semanteme is the range of (0..6). / 2. The Compass server sends the add-account-record page to the user, which contains the users Compass ID CID and the encrypted Compass random number EdCRN). / Similar to the Set-Cookie definition, the

Cluster 23 ($N_C = 218$)

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
worm polici trust attack imag host protocol detect packet node agent fuzzi signatur network rule servic filament user sensor object messag scheme cell port benign ring server event algorithm router client authent watermark access scan cabl stream alic peer certif filter linkabl flow firewal manag channel risk model encrypt password ofth infect protect univers slave pollut secur hash ssir power	Malware defense using network security authentication / Security as services in active networks / The Content Pollution in Peer-to-Peer Live Streaming Systems: Analysis and Implications / A distributed active response architecture for preventing SSH dictionary attacks / AntiWorm NPU-based Parallel Bloom Filters for TCP/IP Content Processing in Giga-Ethernet LAN / Mesh Network Firewalling with Bloom Filters / Securing the Borealis Data Stream Engine / Usable Global Network Access Policy for Process Control Systems / Psachno: a dynamic and generic discovery framework within a peer-to-peer network model / Integrating SSL to the JaCoWeb security framework: project and implementation / Systematic Signature Engineering by Re-use of Snort Signatures / Analyzing the energy consumption of security	According to the definition of patching-hybrid benign worm, we can get the equation of U (t) as follows: 21 dU (t) dP t tU tSt t ItU t dt dt + (16) According to the same / It includes general viscose filament and viscose high tenacity yarn. / We compare the ring signature schemes in (9) and (6) with our scheme in Table 1. / In the experiment, two peers in Brooklyn and Hong Kong were used respectively as sampling peers to gather pollution information. / Assuming that the channel between the requester and SSIRS is , SSSIRS C , then the formalization of requester likes expression (1): , , , , , . / 1. Introduction Fraud and deception are ubiquitous in human life and society. / 4) Neighborhood rogue access point: The access point is setup by other company in the close vicinity. / If f)g303,

Cluster 60 ($N_C = 150$)

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
steveknight node imag herbertsmith steve user knight detect sensor network attack agent power polici protocol servic intrus rout languag trust ofth scheme chaotic devic encrypt certif authent deleg access triz control server malwar packet seed prime modbu manag messag speech corona particip voltag role object neural mobil request msql ipsec rate rule action cluster reactiv load organ modul attribut train	Micro-firewalls for dynamic network security with distributed intrusion detection / The Harpoon security system for helper programs on a Pocket Companion / Contract performance assessment for secure and dynamic virtual collaborations / Towards Secure Information Sharing and Management in Grid Environments / An integrated security framework for XML based management / A policy language for a pervasive computing environment / Optimising malware / A Scenario for Identity Management in Daidalos / Proceedings 23rd International Conference on Distributed Computing Systems Workshops / Distributed control enabling consistent MAC policies and IDS based on a meta-policy approach / The Compositional Security Checker: a tool for the verification of information flow security properties / Introducing	231 Anti-Virus * Defaults notalwaysoptimal - Norarethesignaturefiles Notalwaysaccurate NAI: Keyloggerdetected as low-risk"adware" Tune managementsettingscarefully * / Both KMS and TRIZ have advantages and disadvantages KMS has powerful storing function but no innovative functions, TRIZ is just opposite. / The data field is built in each Modbus function class because each one of these functions has different contents for this field. / For the high-voltage transmission lines are all almost n split-conductors, the conductor surface charge due to corona discharge can be solved through Gauss Law if each phase / deptno number foreign-key departments startdate date, enddate date, budget number (8, 2)) This MSQL command creates a relation called PROJECTS whose name and schema information / To

Cluster 1 ($N_C = 142$)

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
packet node watermark attack detect imag fault power authent servic trust scheme posit protocol vodafon network twoack server speech user agent erfit content path messag laser devic sensor home rule frequenc traceback rout honeypot chaotic walsh multicast mark session encrypt cluster router document weight hash train voltag monitor data link mobil camera model reactiv counterterror algorithm signal code intrus output	On the Security of Distributed Position Services / Key Agreement in Peer-to-Peer Wireless Networks / On optimal placement of intrusion detection modules in sensor networks / Honeypot back-propagation for mitigating spoofing distributed Denial-of-Service attacks / Sender Access Control in IP Multicast / Multicast authentication in fully adversarial networks / 3G-WLAN Convergence: Vulnerability, Attacks Possibilities and Security Model / Secure communication for multipath ad hoc network / Performance Evaluation of Trust Management in Pervasive Computing / Real-time and forensic network data analysis using animated and coordinated visualization / TWOACK: preventing selfishness in mobile ad hoc networks / Securing electronic commerce: reducing the SSL overhead / Securing Mobile Ad Hoc	THE INSTITUTION ENGINEERS Colloquium organised by Professional Group E7 (Telecommunication networks and services) to be held at Savoy Place on Friday, 3 February 1995 PROGRAMME / A. The TWOACK Scheme The TWOACK scheme can be implemented on top of any source routing protocol such as DSR. / For the expected number of false positives, because ERFIT saves 1 bit occupied by FIT for distance calculation, 1 extra bit can be used to store fragment information, and thus has / TherearedefiniteinZ field, Z includes components of N: M: a speech information data frame length, sample rate of 8 kbits, c j : carrier speech code stream, s i : a frame of secret / 5.2 Intra-AS Propagation In intra-AS honeypot back-propagation, honeypot sessions at the HSM of an AS are used to further pin down attack hosts.

Cluster 77 ($N_C = 93$)

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
watermark node servic packet agent curtail seismic imag power imput gravit hyper tfpf attack protocol rout multicast jgram detect univers network afrl fault honeynet vasp outli voltag negoti object pennsylvania cach server cafe outlier partit cell dvmd messag default mpeg harmon ipsec trust stabil shadow british columbia price traffic dugelay embed odrl educ warp transact sensor algorithm linear cluster java	2000 IEEE Power Engineering Society Winter Meeting. Conference Proceedings (Cat. No.00CH37077) / Run-time support for extensible protocol stacks / Insider Attacker Detection in Wireless Sensor Networks / Randomized Intrusion-Tolerant Asynchronous Services / A trust model based routing protocol for secure ad hoc networks / Performance evaluation of anonymous routing protocols in MANETs / Secure and customizable software applications in embedded networks / Denial-of-service in automation systems / NOMS 2000. 2000 IEEE/IFIP Network Operations and Management Symposium 'The Networked Planet: Management Beyond 2000' (Cat. No.00CB37074) / Preventing Service Oriented Denial of Service (PreSODoS): A Proposed Approach / JGram: rapid development of multi-agent pipelines for real-world tasks /	B. Characteristic Analysis of Curtailment Model Taking a line congested in power system for example, i.e., 1l , the curtailment model is simplified as (this analysis can be / For a -player coalitional game with real-valued gain function , a unique imputation of the total gain to each player , denoted by , is given by Shapley value or Shapley theorem / Accordingly gravitational field turbulence due to seismic wave reach instantaneously whereas seismic wave arrive much later (Fig. / 3. VWL-TFPF on seismic data The tests of VWL-TFPF are made on the synthetic seismic data and common shot point seismic data respectively here to indicate its performance on / . 2 Hyper-Cluster Network Architecture Traffic in a large network is generally non-uniform, network nodes can usually be grouped into

Cluster 96 ($N_C = 93$)

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
imag node sensor agent obstacl meat polici power itemset fingerprint decoy network server protocol mail dicom minutia authent host scheme packet encrypt robot track skew alic guardian platform enterpris episod secconfig ofth layer detect signatur password watermark queri core quantum homomorph attack intrud mobil synchron chaotic traceabl servic algorithm ipsec anonym hypervisor conting ultrason differenti region secur caustic negoti ring	Addressing IT Security for Critical Control Systems / Worms vs. perimeters: the case for hard-LANs / DICOM Image Secure Communications With Internet Protocols IPv6 and IPv4 / Dynamic VPN communities: implementation and experience / Comparing the Trust and Security Models of Mobile Agents / Towards Secure E-Commerce Based on Virtualization and Attestation Techniques / Optimizing Security Computation Cost for Mobile Agent Platforms / A remote password authentication scheme for multiserver architecture using neural networks / Analysis of IPSec overheads for VPN servers / Toward a ubiquitous consumer wireless world / Clock Skew Based Node Identification in Wireless Sensor Networks / A novel mobile IP registration scheme for hierarchical mobility management / A Random Key Distribution Scheme	industry competitiveness and especially break the international technical barrier in meat product. / When the front six pieces of sensors detect the obstacle and ISR begins to change direction toward free-space areas, at the same time, the range information values start to / From the detailed Apriori algorithm in (2), we can find the main parts of the Apriori Algorithm are composed ofthree essential procedures: the function Join, the function Prune, / Dashed and dotted line: the performancewithout decoy method. / In DICOM, the open system interconnection (OSI)) basic reference model is used to model the interconnection of medical-imaging equipment, as shown in Fig. / BG WakeUp mode In BG WakeUpmode, the communication controller can transmit a wakeup message to the Bus Guardian and give

Cluster 38 ($N_C = 89$)

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
imag agent nanocryst power node alarm fingerprint sensor dise encrypt packet patient network cluster buss role code task messag medic stream access attack certif secreci trust health nois authent tree watermark secret mcml ident card frank grate grid load protocol traffic paranoid tunnel rout match current mosfet user princip mobil correl subproblem quantum transform server neural feder snoopi honeynet ispwad	SPX: global authentication using public key certificates / Itinerant agents for mobile computing / Privacy-preserving electronic health record linkage using pseudonym identifiers / Paranoid: a global secure file access control system / Reliable software and communication. I. An overview / Chip cards-the application revolution / DISE: a programmable macro engine for customizing applications / SIP roaming solution amongst different WLAN-based service providers / IEEE 802.16 broadband wireless technology and its application to the military problem space / Mining alarm clusters to improve alarm handling efficiency / Wireless broadband multimedia health services: Current status and emerging concepts / An architecture for exploiting multi-core processors to parallelize network intrusion	The random noise at 10 A (Fig.2 (c)) is almost the same as the reference MOSFETs noise without Si nanocrystals. / We show DISE implementations of two ACFs-memory fault isolation and dynamic code decompressionand their composition. / 8. 10-13 and 17-20 Busses 1-6, 8, 10-13, 16-22 and 24 7, 9, 14, 15 and 23 I, 21 1, 7, 9, 14-16 and 21-24 3 4 5 6 Step 6 of the proposed algorithm has indicated that the terminal / Au Denial of Service Malicious agent consumes an excessive amount of the platform resources. / Roles and tasks have been defined, which provide maximum individual security, while allowing detailed access to electronic commerce services. / It can also be seen from Figure 2 that the dissipated power cross over point of a conventional 0.18mCMOS inverter and an MCML buffer is about 3.5GHz

Cluster 68 ($N_C = 86$)

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
prof node power bollard voter vote pvid requisito payment packet china univers transport watermark voltag test market attack protocol reactiv mobil load tunnel trade wavelet price vehicl network imag vlan router wireless attribut aroma rout user informaci shdsa terror vulner detect messag port algorithm agribazaar softwar reserv path instrument conntrack traffic signal peke electr intrus robot australia bond univ line	Using Mobile Agents to Detect and Recover from Node Compromise in Path-based DoS Attacks in Wireless Sensor Networks / Area-Wide System Protection Scheme Against Extreme Contingencies / Zombie Identification Port / Network management for ATCS communications systems / Detecting critical nodes for MANET intrusion detection systems / Risk-constrained congestion dispatch in deregulated power systems / Spot pricing of capacities for generation and transmission of reserve in an extended Poolco model / High-throughput programmable cryptocoprocessor / A security adaptive protocol suite: Ranked Neighbor Discovery (RND) and Security Adaptive AODV (SA-AODV) / Cyberinsurance in IT Security Management / Electric power for the digital age / Study on model of power grid operation security cost in market	Xiaoling. W., Wuhan Institute of Technology, China Prof. Yi Z., Huazhong Normal University, China Prof. Yi. L., Nanjing Normal University, China Prof. Ying L, Shanghai / Depending upon the difficulties encountered at this stage, and the need to investigate other bollard types, the project might be widened to include cast steel and similar bollards. / nIDrnmm d i ed bbs mod)) ((mod nIDrnIDrm d i d i ed bs mod) (mod) (nIDnmrPVID d ibsi mod) (mod 1 PVID i is the sign of voters selected ID i . / Muchos investigadores han reconocido la necesidad de integrar el anlisis de requisitos y la definicin del control de acceso mediante la especificacin de requisitos de seguridad en / me disp.uniroma2.it M-Payments mean that the mobile phone, becoming a personal trust device (PTD), and mobile devices

Cluster 14 ($N_C = 82$)

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
imag microwork prime watermark triangl blind signatur ecsm agent scheme fault attack fuzzi fingerprint peer user antenna quantum node ngerprint protocol power phish messag array parti server victim embed role network latch mobil infer feeder napletsocket bank sensor biometr servic quantiz rfid stego secret qualiti outag signer popey detect polici authent command load path copyright frequenc cost payload partial market	A reliable and secure connection migration mechanism for mobile agents / Access Control and Authentication for Converged Wireless Networks / Access control in peer-to-peer collaborative systems / Honeypots for distributed denial-of-service attacks / 2001 IEEE/PES Transmission and Distribution Conference and Exposition. Developing New Perspectives (Cat. No.01CH37294) / POPEYE: A simple and reliable collaborative working environment over mobile ad-hoc networks / Identification of IT security-relevant system characteristics / Vehicular implementations of public key cryptographic techniques / Trust Management in Peer-to-Peer SIP Using the Security Assertion Markup Language / The ISO reference model entities / Error Detection and Fault Tolerance in ECSM Using Input Randomization / Security	mDNP Users Group did Interoperability demonstrations when it was first introduced TRIANGLE MIcROWORKS. / 1. PV module after the ECSM module. / portredistributionbetweenarbitrarysetsofshareholders. We identieda vulnerabilityin Desmedtand Jajodias redistribution protocol and proved that two conditions, SHARES-VALID and / Here, we consider three aspects for the security analysis of our quantum signature scheme. / The send this list, along with the enhanced fingerprints, to the Department of Dactiloscopy in Argentina, where all the ngerprints of the people in Argentina are kept and where / The result shows that among the four types of phishing attacks, banks in both places are well prepared to handle bogus Web sites but are inadequately prepared to handle phishing / The second one is a

Cluster 78 ($N_C = 75$)

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
watermark imag peer attack node power sensor intrus reput wavelet niap detect signatur agent nsol zahir encrypt messag wireless polici cipher mrsa load vein trust trustworthi sink reload bittorr packet credenti vehicl keyphras globalstar control utter applianc chaotic servic antenna transform mine digit mobil wormhol space palmieri nsec rebroadcast network osek malici moca footstep file softwar certif host cryptosystem decrypt	Security as a new dimension in embedded system design / Wireless Intrusion Detection: Not as easy as traditional network intrusion detection / Security Mechanisms for the IPv4 to IPv6 Transition / A BitTorrent-driven distributed / Cooperating systems for Global Intrusion Detection and Tolerance / A New Development of Symmetric Key Cryptosystem / Internet Security: An Intrusion-Tolerance Approach / Requirements for policy languages for trust negotiation / A block cipher cryptosystem using wavelet transforms over finite fields / A BuyerSeller Watermarking Protocol Based on Secure Embedding / Privacy Preserving Mobility Control Protocols in Wireless Sensor Networks / Research directions for automated software verification: using trusted hardware / Military Message Systems: Current Status and	NIAP continues to build important relationships with government agencies and industry in a variety of areas to help meet current and future IT security challenges affecting the / This phenomenon of selfish behavior harms the equity and usability in Peer-to-Peer system. / Both generation rescheduling (GR) and load shedding (LS) are allowed to decrease NSOL index. / * * P. Palmieri. A. Zahir IEE seminar, London November 13th. / 3. Overview of Identity Based Mediated RSA (IB-mRSA) Mediated RSA (mRSA) involves a special entity, called a SEM, which is an on-line partially trusted server. / 2) The vein is in body. / For example, let us assume that East Field decides that the interaction that it is going to carry with West Field has an extremely high financial value and hence would like to / The

Cluster 146 ($N_C = 68$)

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
imag cluster peer agent humanoid watermark attack robot posit curvelet node load packet mobil shuffl authent chlac traffic conting encrypt devic password anonym network host algorithm forecast eyelid hash server temperatur power interleav migrat scrambl particl client passkey scheme suspend chaotic user protocol signal session messag signatur stabil rout coeffici tupl cost sensor transform test bitstream napletsocket subregion detector friar	Secure multicast conferencing / XScale hardware acceleration on cryptographic algorithms for IPSec applications / Security of ATCS wireless railway communications / A reliable connection migration mechanism for synchronous transient communication in mobile codes / A Practical Approach to provide Communication Privacy / Building secure user-to-user messaging in mobile telecommunication networks / An Identity-Based Key Management Framework for Personal Networks / Defending against Distributed Denial of Service (DDoS) Attacks with Queue Traffic Differentiation over Micro-MPLS-based Wireless Networks / Designing Secure Peer-to-Peer Voice Applications in Ad Hoc Wireless Networks / SASO: Security Adaptive Self-Organization for Wireless Sensor Networks / A multipath ad hoc routing approach to	In the present paper, a concept of human support system using humanoid robots is proposed, humanoid robots exist in our environment and protect us from dangers. / The robustness of the curvelet domain algorithm against the attacks listed in Table.1 is better than that of the ridgelet domain method. / However, we adopt the spatial shuffling encryption approach that maintains syntax compliance. / The Cow GaitRecognitionUsing CHLAC Shu Mimura Keichi Itoh Takumi Kobayashi Tomohiro Takigawa Atsushi Tajima Atsushi Sawamura Nobuyuki Otsu United TechnologiesInstitute, 1-1-1 / proposed to fit the eyelid by two straight lines, which, however, trades accuracy for simplicity. / Without knowledge of spreading code or interleaver/deinterleaver, its impossible to recover the desired users signal. / 4

Cluster 63 ($N_C = 66$)

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
chaotic mobil voltag book frame synchron reactiv cyclon rotor watermark imag agent lock spnego home evalu ipojo cylind messag asset ident fingerprint scheme packet coordin power error risk buyer manag tunnel subsystem synchronis theft signal channel authent margin dispers custom video layer encrypt signatur peer protocol logo preston ldap confere test node privat wavelet attack network block regul peopl scrip	InfoShield: a security architecture for protecting information usage in memory / A contextual framework for combating identity theft / A lightweight VPN connection in the mobile multimedia metropolitan area network / Mobile IP and WLAN with AAA authentication protocol using identity-based cryptography / IPv6 future approval networking / The latest in virtual private networks: part I / A novel peer-to-peer payment protocol / An Improved Identity-based Fault-tolerant Conference Key Distribution Scheme / A secure kernelized architecture for multilevel object-oriented databases / On the Assumption of Equal Contributions in Fingerprinting / Universally Composable Key-Evolving Signature / Shape-Driven Gabor Jets for Face Description and Authentication / Demand-Side Management to Improve Power	Some of the issues in the book are of interest only to security professionals, but many of us nowadays have home networks for which we have to provide the security ourselves. / Cyclone uses 256 x equations, each equation uses 4 bytes pre-generakd data, and 4 bytes of key. / It is designed and developed on the technology of Java GSS-API, JAAS, Kerberos and SPNEGO. / This allows pervasive applications built and deployed on iPOJO platforms to obtain self-management support from the underlying middleware runtime, hence requiring no extra effort / More consideration needs to go to P-frames since, unlike Iand B-frames, not all P-frames are of equal importance. / Although the main research on mobile agents does not focus on its applicability in management, several publications regard it as a

Cluster 104 ($N_C = 66$)

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
node imag democraci ocdma trust alert object rfid itemset constraint scheme film video server user mobil sensor logic energi packet traffic clariti udmv encrypt obfusc monoton cluster arteri train attack privaci code moham sequenc webcam pairwis reader polici encod atfa rule network detect program hyperbol manag messag client locat neighbor finger path passag document consumpt power wireless protect index authent	Transaction Oriented Text Messaging with Trusted-SMS / Source routing based pairwise key establishment protocol for sensor networks / Safeguarding and charging for information on the Internet / Binary tree based public-key management for Mobile Ad Hoc Networks / Reputation-based trust management in wireless sensor networks / ISE02-4: L2VPN over Chord: Hosting Millions of Small Zeroconf Networks over DHT Nodes / TinyRNG: A Cryptographic Random Number Generator for Wireless Sensors Network Nodes / Radio frequency-based personnel location systems / Genetic Algorithm Based Secure Authentication Protocol with Dual Central Server and Token Authentication in Large Scale Mobile Ad-Hoc Networks / Security in the Spring name service / A UML model for multi-level security using the IPSEC ESP	Stuart Anderson Massimo Felici LFCS, School of Informatics, The University of Edinburgh Mayfield Road, Edinburgh EH9 3JZ, UK CUs0a, mfeliciCV inf.ed.ac.uk Abstract Recent / 1 Working principle of OCDMA PON 100 SC1.S2.2 (Invited) 11:00 11:30 . / The results show, by this model, the alerts reduction rate on LLDOS1.0 finally reaches 99.08 , and 98.37 on LLDOS2.0. / 10. Use C REST to filter the complete set of itemsets to get the finally results. / The Security Equipment Assessment Laboratory has carried out a series of trials using different combinations of solar gain control reflective films and screen printed privacy / 2. The method based on the Weighted Blocking Discrete Cosine Transform (WBDCT) The conventional view thinks that the curve of image clarity evaluation function always has

Cluster 87 ($N_C = 59$)

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
fingerprint collud realm rcddl collus qosc authent silo watermark raif file channel attack frame cluster pana path gate user selfish sensor network wireless polici onpp trie client node devic precollus ipsec asset healthcar storag reus tibet home imag servic traffic wddl konpp prime bandwidth detect copi isotop packet iscsi cooper server multimedia trust hdtv encrypt precharg shype power buffer content	A distributed vulnerability detection system for WLANs / Traitor-Within-Traitor Behavior Forensics: Strategy and Risk Minimization / Secure Embedding of Spread Spectrum Watermarks using Look-up-Tables / Shamon: A System for Distributed Mandatory Access Control / DEEJAM: Defeating Energy-Efficient Jamming in IEEE 802.15.4-based Wireless Networks / A coordination and bandwidth sharing method for multiple interfering neighbor networks / A Study on the Call Admission and Preemption Control Algorithms for Secure Wireless Ad Hoc Networks Using IPSec Tunneling / Integrated OTP-Based User Authentication Scheme Using Smart Cards in Home Networks / A wireless LAN architecture using PANA for secure network selection / Forensic analysis of nonlinear collusion attacks for multimedia fingerprinting / A	Moreover, compromising realm hp. / In this paper, we propose a novel design methodology for DPAresistant Secure ICs using Reduced Complementary Dynamic and Differential Logic (RCDDL) (12). / 2) Probability of Being Detected: Given the colluded copy as in (4), the fingerprint extracted from the th frame is shown in (6) at the bottom of the page, where contains terms / 1) State 0 A QoSC does not have any pending share bandwidth requests when it is in Stateo. / The probe silo lets the host query the ACT for its functionality. / 2.3 RAIF meta-data Small files may occupy only a portion of a single stripe. / In the proposed network architecture, we use PANA (Protocol for carrying Authentication for Network Access) (1), which is a link-layer agnostic network access authentication / 4.3

Cluster 97 ($N_C = 56$)

Wort-Zusammenfassung	Titel-Zusammenfassung	Satz-Zusammenfassung
radar trust vmsoar proxi sequenc fingerprint imag flash artifact clutter ercot patch detect signal neural adcvc cipher hindsight power collud split entiti client interfer camera market rebalanc voltag code espm convert bibd collabor signatur price polici teamspac sysadm user drive capac scheme network sgkmp portfolio attack block comiti hardwar frequenc server servic strategyproof signer strategi quandari video file manag coordin	Network firewalls / Collusion Secure Scalable Video Fingerprinting Scheme / The use of spread spectrum to improve information hiding in images / Implementing QoS-adaptation in coordination artifacts by enhancing Cougaar multi-agent middleware / Identification of Baulks and the Application Manners of Setting Electronic Commerce Into Effect in Iran / Detection of low observable targets within sea clutter by structure function based multifractal analysis / A Cognitive Approach to Intrusion Detection / Systematic Policy Analysis for High-Assurance Services in SELinux / Enabling inter-company team collaboration / Neuromorphic vision chips / Fault and attack management in all-optical networks / Arbitrarily dirty paper coding and applications / Trust relationships in secure systems-a distributed	The VMSoar project is using these cognitive structures to construct an agent with human-like reasoning and comunication abilities. / Suppose a forger wants to forge the individual proxy certificatev prime n 1 . / MAPPING BETWEEN VULNERABILITY AND COUNTERMEASURE Vulnerability Countermeasure Password Exposure Data Recovery S/W Bug Hardware without Packaging Hash Function O O Wiping / The notion of a Role makes the abstract conception of operating instructions more concrete: a client of a Coordination Artifact always plays a specific Role relative to that / An important finding of our study is that sea clutter data is weakly nonstationary in the time scale range of 0.01 s to a few seconds. / ERCOTs market has been designed to be consistent with an Energy-Only market with features that ensure