Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Bachelorarbeit

Fabian Simroth

Konzept zur automatischen Identifikation von Risiko-merkmalen in der Versicherungsbranche

Fakultät Technik und Informatik
Department Informatik

Faculty of Engineering and Computer Science
Department of Computer Science

Fabian Simroth

Konzept zur automatischen Identifikation von Risikomerkmalen in der Versicherungsbranche

Bachelorarbeit eingereicht im Rahmen der Bachelorprüfung

im Studiengang Bachelor of Science Wirtschaftsinformatik
am Department Informatik
der Fakultät Technik und Informatik
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuender Prüfer: Prof. Dr. Klaus-Peter Schoeneberg
Zweitgutachter: Ingo Leusing

Thema der Arbeit

Konzept zur automatischen Identifikation von Risikomerkmalen in der Versicherungsbranche

Schlüsselbegriffe

Versicherung, Text Mining, Java, Natural Language Processing, Pattern-Matching

Zusammenfassung

In dieser Arbeit wird ein Konzept zur automatischen Identifikation von Risikomerkmalen der Versicherungswirtschaft aus von einem Sachbearbeiter gepflegten Kommentarfeldern erarbeitet. Durch die Auswahl und anschließende Anwendung von gängigen Textmining-Methoden sind für den Computer bislang unerreichbare, aber für das Tagesgeschäft relevante Informationen maschinell auswertbar zu machen, um die Prozesse bei der Sachbearbeitung zu verbessern und diese stärker zu automatisieren.

Title of the thesis

Concept for automatic identification of risk characteristics in the insurance business

Key words

Insurance, Text Mining, Java, Natural Language Processing, Pattern Matching

Abstract

A concept for automatic identification of risk characteristics in the insurance business is to be developed by analyzing comment fields filled by the company's employees. Through selecting and applying adequate text mining methods, relevant information for daily business shall be extracted which was not accessible for digital processing up to now, in order to improve and increasingly automate tasks in daily business.

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

Abkürzungsverzeichnis

Abkürzung	*Bedeutung*
AHB	Allgemeinen Versicherungsbedingungen für die Haftpflichtversicherung
CSV	Comma-Separated Values
DDL	Data Definition Language
EDV	Elektronische Datenverarbeitung
ETL	Extraction Transforming Loading
FTP	File-Transfer-Protocol
GD	Generali Deutschland Gruppe
GDIS	Generali Deutschland Informatik Services GmbH
GEV	Generali Versicherung AG
JDK	Java Development Kit
JVM	Java Virtual Machine
NER	Named Entity Recognation
POS	Part-Of-Speech
SQL	Structured Query Language
SUH	Sach, Unfall, Haftpflicht

1 Einleitung

In diesem Kapitel wird dem Leser eine Einführung in das in dieser Thesis behandelte Thema gegeben und der Hintergrund aufgezeigt. Motivation und Problemstellung werden kurz dargelegt sowie die Zielsetzung definiert. Um das Lesen zu vereinfachen und die Bestandteile dieser Arbeit aufzuzeigen, wird außerdem ein kurzer Überblick über den Aufbau gegeben.

1.1 Motivation und Problemstellung

Heutzutage kommt es in vielen Bereichen der EDV bei der manuellen Bearbeitung von Daten, ob Einfügen neuer Datensätze, deren Änderung oder Löschung, häufig zu Fehlern durch falsche oder unvorhergesehene Nutzung von Masken und Dialogen. Masken und Dialoge bezeichnen Benutzeroberflächen, die es ermöglichen, Daten in ein System einzutragen, die für geschäftliche Prozesse benötigt werden. Dies sind beispielsweise Eingabefelder für Vornamen und Nachnamen sowie für die Adresse, aber auch Felder für Beträge oder Checkboxen für optionale Optionen. Neben den üblichen und notwendigen Feldern, die in der Versicherungsbranche für die Vertragsneuerfassung existieren, sind auch Felder für zusätzliche, über den Dialog nicht abgebildete, Eingabemöglichkeiten implementiert, um dem Sachbearbeiter höchst mögliche Flexibilität zu schaffen. So kommt es vor, dass der Sachbearbeiter bei der Erfassung von Kundendaten von ihm als wichtig angesehene Informationen, Kommentare oder Erinnerungen in diesen zusätzlichen Feldern als Freitext formuliert. Neben diesen Daten sind auch wichtige Informationen abgelegt, die für die Tarifierung, also die Berechnung des Beitrages, herangezogen werden. Auch für die spätere Bearbeitung eines Vertrages ist die Verwendung der Inhalte entscheidend. Da diese Informationen aber in Sätzen verpackt sind, gestaltet sich eine maschinelle Auswertung als schwierig.

1.2 Zielsetzung

Ziel dieses Projektes ist die Entwicklung eines prototypischen Vorgehens, um die Freitextfelder auszuwerten und daraus Risikomerkmale für Verträge und Policen abzuleiten. Risikorelevante Merkmale sind Elemente, welche ausschlaggebend für das Versiche-

rungsrisiko sind. So ist bei einer KFZ-Versicherung das versicherte Risiko das Kraftfahrzeug und bei einer Hausratversicherung sind es alle Gegenstände innerhalb eines Haushalts wie Einrichtung, Bekleidung aber auch Bargeld und Wertgegenstände. Versicherungsprozesse, insbesondere in den Bereichen des Neugeschäftes und der Schadenregulierung, sind damit zu verbessern, transparenter zu gestalten und stärker zu automatisieren. Somit führen sie zu einer Arbeitserleichterung bei den Sachbearbeitern. Dazu werden die bisher angesammelten Daten aus dem Quellsystem ausgelesen, aufbereitet, analysiert und mit passenden Text Mining-Methoden maschinell auswertbar gemacht. Abschließend sind die gewonnenen Informationen bzw. Merkmale im Quellsystem anhand einer geeigneten Datenstruktur zu ergänzen.

1.3 Aufbau der Arbeit

Diese Arbeit unterteilt sich inhaltlich in fünf Teile und beginnt mit dieser Einleitung, in welcher der Rahmen und Umfang dieser Thesis aufgeführt werden.

In Kapitel 2 werden der informelle Hintergrund erläutert und der aktuelle Stand der Forschung aufgezeigt. Zusätzlich dient dieses Kapitel als Einführung besonders in das Thema Text Mining, auf dem diese Arbeit basiert. Daneben wird der Hintergrund der Versicherungsbranche, explizit der von der Generali Gruppe und den involvierten Konzernunternehmen, beschrieben.

Als nächster Schritt wird die Ist-Situation analysiert, um daraus Regeln abzuleiten. Die Analyse und die umzusetzenden Regeln werden in Kapitel 3 beschrieben. Dafür wird ein Blick auf die derzeit vorliegenden Inhalte der Freitextfelder geworfen und geprüft, inwiefern sich die im vorherigen Kapitel erarbeiteten Verfahren des Text Minings darauf anwenden lassen. Aus diesem Wissen wird ein Konzept für die zukünftige automatisierte Auswertung der Daten erarbeitet.

Diese Ergebnisse werden als Basis für die Umsetzung, welche in Kapitel 4 beschrieben ist, verwendet, und es wird ein Prototyp erarbeitet. Es werden die Schritte, die zur Aufbereitung nötig sind, umgesetzt und die für diese Problemstellung besten Verfahren implementiert. Anschließend werden die Daten in das Quellsystem zurückspielt, die erziel-

ten Ergebnisse mit den Erwartungen verglichen und die Algorithmen auf ihre Fehlerquote untersucht. Außerdem werden kurz mögliche zukünftige Entwicklungen des Konzeptes und des Prototyps aufgezeigt.

Im letzten Kapitel wird ein kritischer Blick auf die gewonnenen Erkenntnisse geworfen und Probleme mit deren Lösungen noch einmal kurz dargestellt.

2 Grundlagen

In diesem Kapitel werden zunächst die versicherungswirtschaftlichen Hintergründe dieser Arbeit und die etablierten Prozesse dargelegt. Anknüpfend daran wird der aktuelle Stand der Forschung des Gebietes Text Mining erläutert.

2.1 Versicherungswirtschaft

Diese Arbeit baut in großen Teilen auf die in einer Versicherung erhobenen Daten auf, welche im laufenden Tagesgeschäft durch die Arbeit von Sachbearbeitern entstehen. Zu dem Tagesgeschäft gehört unter anderem die Policierung von Verträgen, was bedeutet, dass ein Kunde einen Versicherungsvertrag mit der Versicherung abschließt. In diesem Vorgang wird die Police erstellt, welche alle relevanten Vertragsdaten umfasst.

Diese Arbeit entsteht in der Kooperation mit der Generali Versicherung AG (GEV). Das Unternehmen bietet den nötigen Rahmen, die Datengrundlage für Auswertungen und den Anwendungsfall. Die GEV gehört zu der Generali Deutschland Gruppe (GD), welche durch den Mutterkonzern, die *Assicurazioni Generali* in Italien, geführt wird. Die GD ist mit ihren Konzernunternehmen der zweitgrößte Erstversicherer in Deutschland[1]. Während die GEV für den Endkunden Services rund um Versicherungen erbringt, stellt die Generali Deutschland Informatik Services GmbH (GDIS) innerhalb der GD IT-Dienstleitungen zur Verfügung.

Die GD hat im Verlauf der Zeit viele verschiedene Versicherer mit unterschiedlichen Systemen und Strukturen aufgekauft, was zu einer heterogenen Systemlandschaft geführt hat. Es wurden im Laufe der Zeit viele unterschiedliche Anforderungen und Anpassungen von verschiedenen Stakeholdern umgesetzt.

Das in dieser Arbeit erarbeitete Konzept wurde durch die GDIS begleitet, welche Infrastruktur und Wissen zur Verfügung gestellt hat.

[1] Vgl. Generali Deutschland: Risikoatlas

2.2 Text Mining

Der Begriff Text Mining bezeichnet im Allgemeinen die Extraktion von Wissen aus dem Wissensrohstoff Text[2]. Bei Text in dessen ursprünglicher Form, also der Aneinanderreihung von Zeichen, handelt es sich um Daten, genauer gesagt, um unstrukturierte Daten. Unstrukturiert heißt, dass die Daten nicht in einem gegebenen Schema vorliegen und auslesbar sind, so wie es zum Beispiel bei Datentabellen der Fall ist. Auch wenn Texte in der Regel gewissen syntaktischen und semantischen Regeln unterliegen, nämlich dem Satzbau, so gibt es ohne weiteres keine Möglichkeiten, die Information aus dem Text zu extrahieren.

Es werden dabei die Begriffe Daten, Informationen und Wissen unterschieden. Wie bereits erwähnt, handelt es sich bei Texten zunächst um Daten. Für die Interpretation dieser Daten wird ein Interpretationsschema angewendet. Dieses Schema wird als Grammatik bezeichnet und bei dessen Anwendung werden die Daten in Informationen transformiert. Werden jetzt verschiedene Informationen miteinander verknüpft, um daraus beispielsweise neue Erkenntnisse oder Lösungen für Probleme zu generieren, wird von Wissen gesprochen. Die Zusammenhänge sind in der folgenden Grafik am Beispiel von Wettermessdaten dargestellt.

[2] Vgl. Marinschek, M./Daume, H. (2001) zitiert nach Zietsch, C. / Zänker, N. (2011, S.12)

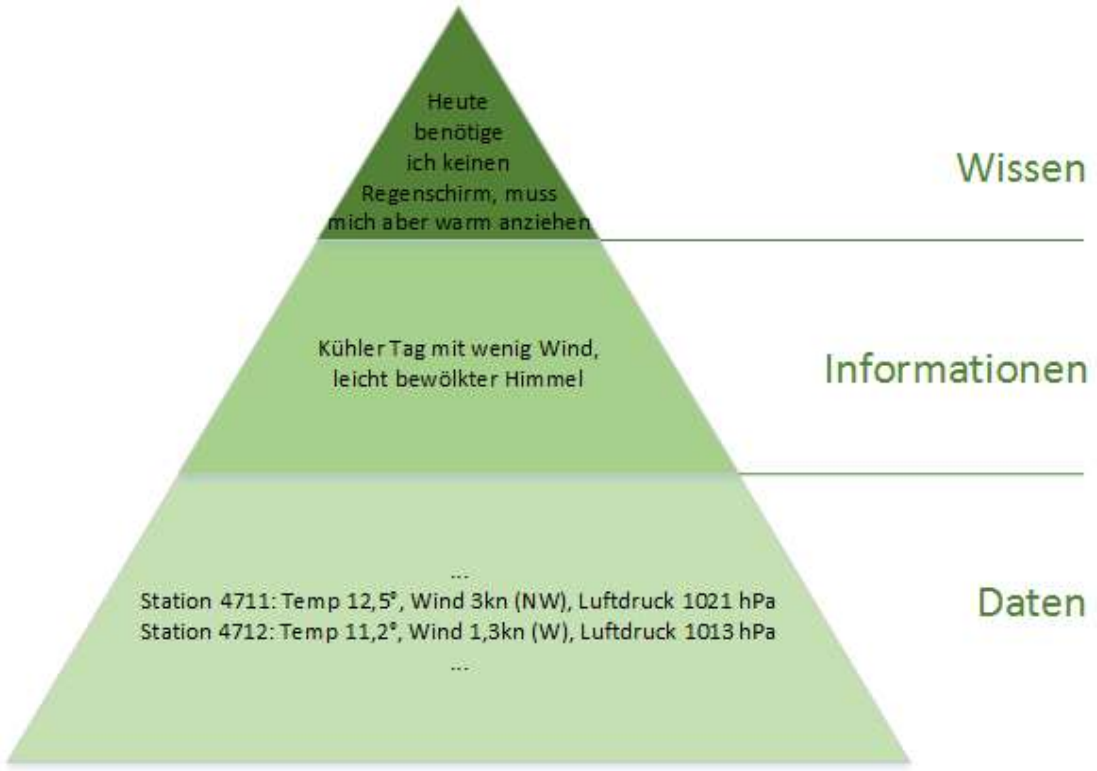

Abbildung 1: Wissenspyramide anhand von Wetterdaten

Auf der untersten Ebene befinden sich Messdaten von Wetterstationen. Diese sind beispielsweise einfach in Tabellen abgelegt. Durch die Interpretation dieser Daten lassen sich verschiedene Informationen daraus ableiten. So können ein niedriger Luftdruck zu schlechtem Wetter und niedrige Temperaturen zu Schneefall oder Glätte führen. Diese Informationen allein sind für uns Menschen erst dann sinnvoll und auch wertvoll, wenn wir die Informationen mit anderen Informationen oder Erfahrungen verknüpfen und daraus Wissen erzeugen. Wird anhand der Informationen „ein regnerischer Tag" erwartet, greift man als Reaktion darauf zum Regenschirm und fährt bei Schneefall statt mit dem Fahrrad mit der Bahn. Erst die Verknüpfung von Informationen und Erfahrungswerten ist gewinnbringend.

Gerade Texte spielen in diesem Kontext eine besondere Rolle, da sie seit ihrer Schaffung dazu dienen, Wissen festzuhalten und weiterzugeben. Umso erstaunlicher ist es, dass

das maschinelle Lesen und Verstehen von Texten bisher noch nicht vollumfänglich möglich ist und es ungelöste Probleme birgt[3].

Verschiedene Verfahren des Text Minings – insbesondere in Bezug auf die oben genannte Problemstellung – werden im Folgenden dargestellt und erläutert. Abschließend wird kurz geprüft, welche Verfahren grundsätzlich für die Problemstellung geeignet sind und welche sich von vornherein ausschließen lassen.

2.2.1 Zipfsche Gesetze

Bei den Zipfschen Gesetzen handelt es sich um eine statistische Regelmäßigkeit, die auftritt, wenn der Rang eines Wortes mit dessen Häufigkeit multipliziert wird. Der Rang ergibt sich aus der Position des Wortes bei einer Sortierung nach der Häufigkeit des Auftretens in einem Text (Korpus). Das Gesetz besagt, dass das Produkt beider Werte für alle Wörter annähernd gleich ist und somit eine Konstante darstellt. Daraus folgt, dass die Häufigkeit ungefähr umgekehrt proportional zu dem Rang ist[4]. In der folgenden Tabelle ist dieser Sachverhalt dargestellt. Während der Rang und in die ermittelte Häufigkeit relativ stark variieren, bleibt das Produkt aus den beiden Werten konstant zwischen 14 und 20 Millionen. Eine Ausnahme bildet der erste Eintrag.

Rang r	Wortform	Häufigkeit n	r * n = k
1	der	7.377.897	7.7377.897
2	die	7.036.092	14.072.184
3	und	4.813.169	14.439.507
4	in	3.768.565	15.074.260
5	den	2.717.150	13.585.750
10	sich	1.680.106	16.801.060
100	immer	197.502	19.750.200
1000	Medien	19.041	19.041.000
10000	vorläufige	1.664	16.640.000

Tabelle 1: Wortformen sortiert nach Rang

[3] Vgl. Heyer, G. / Quasthoff, U. / Witting, T. (2006, S. 14ff.)
[4] Vgl. Zipf, G. K. (1935) zitiert nach Heyer, G. / Quasthoff, U. / Witting, T. (2006, S. 88)

2.2.2 Differenzanalyse

Bei der Differenzanalyse werden die Wörter eines Textes, des Analysekorpus, in vier Klassen eingeteilt und es wird damit versucht, diskriminierende Terme zu identifizieren. Dazu wird die Häufigkeit eines Terms mit der Häufigkeit desselben Terms aus einem Referenzbestand (Referenzkorpus) verglichen. Der Referenzkorpus wird aus allgemeinen und wenig fachbezogenen Texten, wie Tageszeitungen oder Romanen, erstellt. Anhand des Verhältnisses der Häufigkeiten des Terms aus beiden Korpora lassen sich die folgenden vier Klassen ableiten[5]:

Klasse	Beschreibung	Bedeutung
Klasse 1	Der analysierte Term kommt nicht im Referenzkorpus vor	Bei diesen Termen handelt es sich um Fachausdrücke, die in der allgemeinen Sprache nicht geläufig sind.
Klasse 2	Der analysierte Term kommt im Verhältnis zum Referenzkorpus häufig vor	Bei solchen Termen handelt es sich um Fachterme. Für die Identifizierung solcher Terme sind Schwellenwerte zu definieren, ab denen ein Term *auffällig* häufig vorkommt
Klasse 3	Der analysierte Term kommt sowohl im Analysekorpus, als auch im Referenzkorpus mit einem ähnlichen Verhältnis vor	Diese Terme sind meist Stoppwörter, Artikel oder allgemeine Begriffe, die keine themenspezifischen Inhalte abbilden
Klasse 4	Der analysierte Term kommt im Verhältnis zum Referenzkorpus seltener vor	Diese Terme stellen keine weitere Relevanz für die Analyse dar

Tabelle 2: Die vier Klassen der Differenzanalyse [5]

[5] Vgl. Heyer, G. / Quasthoff, U. / Witting, T. (2006, S. 95f.)

Für die Analyse von Texten sind besonders die ersten beiden Klassen relevant, da sie alle Terme umfassen, die es ermöglichen, den Text zu kategorisieren. Terme aus den Klassen 3 und 4 stellen inhaltlich keine Relevanz dar und werden deshalb meist vernachlässigt. Für die Definition des Schwellwertes zwischen den Klassen 2 und 3 gibt es keinen festen Wert, sondern dieser wird an die produzierten Ergebnisse angepasst[6].

2.2.3 Probalistische Sprachmodelle

Probalistische Sprachmodelle gehen einen Schritt weiter und betrachten nicht nur einzelne Wörter, sondern Wortgruppen. Sie bieten die Möglichkeit, komplexe Sprachgebilde sowohl auf ihre syntaktische, als auch ihre semantische Richtigkeit bzw. die Wahrscheinlichkeit, dass die Wortgruppen korrekt sind, zu prüfen.

Auch wenn die natürliche Sprache grammatikalischen Regeln folgt, gestalten sich die Analyse und das Verständnis eines Satzes oder Textes für Computerprogramme als schwierig, da sich durch die Kombinationen von Wörtern nach den grammatikalischen Regeln eine Vielzahl von gültigen Sätzen erzeugen lässt. Als Größenordnung sei hier der deutsche Wortschatz genannt, der zwischen 300.000 und 500.000 Wörter beinhaltet, auch wenn ein normaler Sprecher lediglich bis zu 16.000 verwendet[7]. Grammatikalische Regeln ermöglichen dem Computer zwar Sätze auf ihre syntaktische Korrektheit zu prüfen, jedoch ist die semantische Überprüfung von Inhalten keine triviale Aufgabe. Dazu betrachten wir einen Beispielsatz in verschiedenen Ausprägungen:

Satz	Korrektheit
Der Fußball flog in hohem Bogen in das Tor.	Satz ist syntaktisch und semantisch korrekt
Der Fußball in hohem Bogen flog in das Tor.	Satz ist zwar nicht syntaktisch, aber semantisch korrekt
Das Tor flog in hohem Bogen in den Fußball.	Satz ist zwar syntaktisch, nicht aber semantisch korrekt

Tabelle 3: Satzbeispiele für syntaktische und semantische Fehler

Die zweite Version des Satzes stellt eine semantisch gültige Aussage dar und kann durch einen Menschen trotz der fehlenden Befolgung der deutschen Grammatikregeln verstanden werden. Die Semantik beschreibt hier die Sinnhaftigkeit. Die dritte Version jedoch stellt zunächst eine syntaktisch richtige Aussage dar, die jedoch keinen Sinn ergibt,

[6] Vgl. Heyer, G. / Quasthoff, U. / Witting, T. (2006, S. 97)
[7] Vgl. Bibliographisches Institut GmbH: Duden | Zum Umfang des deutschen Wortschatzes

da ein Tor nicht fliegen kann, und schon gar nicht in einen Ball hinein. Demnach beschreibt die Syntaktik hier eine gültige Abfolge von Wörtern. Eine Bewertung dieses falschen Sachverhaltes gestaltet sich für die Maschine insofern schwierig, als dass sie keine Erfahrung und Vorstellung der Objekte hat.

Deshalb wird versucht, die Wahrscheinlichkeit von Wortfolgen zu Hilfe zu nehmen, indem die Wahrscheinlichkeit des Auftretens vor oder nach anderen Wörtern berechnet wird. Als Grundlage dazu dient ein Trainingskorpus, durch den solche Wahrscheinlichkeiten festgelegt werden. In der Praxis hat sich gezeigt, dass es reicht, zwei vorhergehende Wörter in die Berechnung mit einzubeziehen, da eine Erweiterung der Spanne bei stark erhöhtem Rechenaufwand keine deutlichen Verbesserungen mit sich bringt[8]. So liegt die Wahrscheinlichkeit, dass auf die Wortfolge *„Das Tor“* das Wort *„flog“* folgt, relativ niedrig, da im Trainingskorpus die Kombination eher selten vorkommt. Folgende Satzstellung ist zwar gültig, aber im normalen Sprachgebrauch eher unüblich: „Auf das Tor flog der Fußball zu“. Nun werden für alle Wörter eines Satzes deren Wahrscheinlichkeiten berechnet und am Ende multipliziert. Die Wahrscheinlichkeit, dass ein Wort in Bezug auf seine Vorgänger gültig ist, und die Satzwahrscheinlichkeit, dass der komplette Satz gültig ist, sind in den folgenden beiden Gleichungen dargestellt. p_{w_n} berechnet sich durch die Anwendung des Satzes von Bayes, also durch die bedingte Wahrscheinlichkeit. Die nachfolgenden Gleichungen sind angelehnt an Heyer, G. / Quasthoff, U. / Witting, T.: *Text Mining: Wissensrohstoff Text* (2006, S. 102 ff.)

Formel 1: Wahrscheinlichkeit des Auftretens eines Wortes unter Betrachtung seiner zwei Vorgänger

$$p_{w_n} \approx \frac{|w_{n-2}, w_{n-1}, w_n|}{|w_{n-2}, w_{n-1}|}$$

Formel 2: Berechnung der Wahrscheinlichkeit, dass ein Satz gültig ist

$$p_{Satz_{gültig}} \approx \prod_{n=1}^{m} p_{w_n}$$

Durch das Auftreten von Fachausdrücken, die im Trainingskorpus nicht vorkommen und durch deren Wahrscheinlichkeit des Auftretens $p_{w_n} = 0$ kann es passieren, dass der Satz

[8] Vgl. Heyer, G. / Quasthoff, U. / Witting, T. (2006, S. 104)

als ungültig eingestuft wird. Um dem zu entgehen werden sogenannte *Smoothing*-Algorithmen verwendet. So sei zum Beispiel die Einbeziehung von den Wahrscheinlichkeiten von Uni- bzw. Bigrammen genannt. Bei Uni- und Bigrammen werden weniger als drei Wörter beachtet. Stattdessen werden auch die Wahrscheinlichkeit des Wortes selbst und die Wahrscheinlichkeit des Wortes in Kombination mit einem weiteren Wort herangezogen. So ergeben sich die neuen Gleichungen:

Formel 3: Berechnung der Wahrscheinlichkeit eines Wortes unter Zuhilfenahme eines Smoothing-Algorithmus

$$p_{w_n} \approx \left(\lambda_1 * \frac{|w_{n-2}, w_{n-1}, w_n|}{|w_{n-2}, w_{n-1}|} + \lambda_2 * \frac{|w_{n-1}, w_n|}{|w_{n-1}|} + \lambda_3 * \frac{|w_n|}{\left|\sum_{k=1}^{m} w_k\right|}\right)$$

Formel 4: Wahrscheinlichkeit, dass ein Satz gültig ist, unter Zuhilfenahme eines Smoothing-Algorithmus

$$p_{Satz_{gültig}} \approx \prod_{n=1}^{m} \left(\lambda_1 * \frac{|w_{n-2}, w_{n-1}, w_n|}{|w_{n-2}, w_{n-1}|} + \lambda_2 * \frac{|w_{n-1}, w_n|}{|w_{n-1}|} + \lambda_3 * \frac{|w_n|}{\left|\sum_{k=1}^{m} w_k\right|}\right)$$

$$\text{mit } 0 \leq \lambda_i \leq 1 \text{ und } \sum_{i=1}^{3} \lambda_i = 1$$

Der Faktor ʎ dient zur Gewichtung der einzelnen Terme. Durch die erweiterte Betrachtung der Uni- und Bigramme ist eine Auswertung, die sich auf Null beläuft, sehr unwahrscheinlich, da einzelne Wortformen in ihrem alleinigen Auftreten wahrscheinlicher sind als in der Kombination mit anderen.

Ein weiterer Punkt, welcher bei den probalistischen Modellen eine Rolle spielt, ist die Methode des „Part-Of-Speech"-Taggings (kurz POS-Tagging). Dabei wird, angelehnt an die eben erläuterte Berechnung der Wahrscheinlichkeit für das Auftreten eines Wortes, die Wahrscheinlichkeit des Auftretens einer Wortart berechnet. Die Wortarten sind z.B. Artikel, Konjunktionen, Nomen, Eigennamen, Negationen, Satzzeichen und Weitere[9]. Damit ergibt sich eine Satzstruktur, in der alle Wörter mit ihrer Wortart getaggt sind. An folgendem Beispiel wird das Tagging dargestellt:

[9] Vgl. University of Pennsylvania: Penn Treebank P.O.S. Tags

Der *[Artikel]* Fußball *[Nomen]* fliegt *[Verb]* in *[Präposition]* hohem *[Adjektiv]* Bogen *[Nomen]* in *[Präposition]* Manuels *[Eigenname]* Arme *[Nomen]*.

Alternativ ist es möglich, einen regelbasierten Tagger zu verwenden, der in zwei Schritten arbeitet. Schritt 1 ist das Durchsuchen eines Wörterbuches, um für ein Wort festzustellen, welcher Kategorie es zuzuordnen ist und anschließend in Schritt 2 Mehrdeutigkeiten aufzulösen.

Zum Verständnis der Mehrdeutigkeit seien folgende zwei Sätze aufgeführt:

Ich **stimme** für dieses Vorhaben.
Mit lauter **Stimme** warb ich für dieses Vorgehen.

In dem ersten Satz steht „stimme“ für ein Verb, im zweiten Satz jedoch für ein Nomen. Um diese Konflikte aufzulösen werden Regeln spezifiziert, wie zum Beispiel, dass meistens nach einem Personalpronomen ein Verb kommt. Solche Regeln zu erfassen ist aufwendig und kann sehr komplex werden, weshalb häufig der oben beschriebene stochastische Tagger zum Einsatz kommt[10].

2.2.4 Hidden Markov Modell

Ähnlich dem probalistischen Sprachmodell arbeitet auch das Hidden Markov Modell, welches anhand der Wörter die Wahrscheinlichkeit für das darauffolgende Wort ermitteln kann. Die Funktionsweise entspricht der eines endlichen Automaten, der sich zunächst in einem Startzustand befindet. Im Laufe der Ausführung wechselt der Automat durch das Eingeben von Eingabesymbolen in andere Zustände. Die Übergänge in andere Zustände werden als Übergangsfunktionen dargestellt. Die Übergangsfunktion berechnet eine Übergangswahrscheinlichkeit, um vom aktuellen in den nächsten Zustand überzugehen. Dabei wird entweder nur der aktuelle Zustand oder vorherige Zustände einbezogen[11].

[10] Vgl. Manning, C. / Schütze, H. (1999, S. 357f.)
[11] Vgl. Brants, T. (1999, S. 3f.)

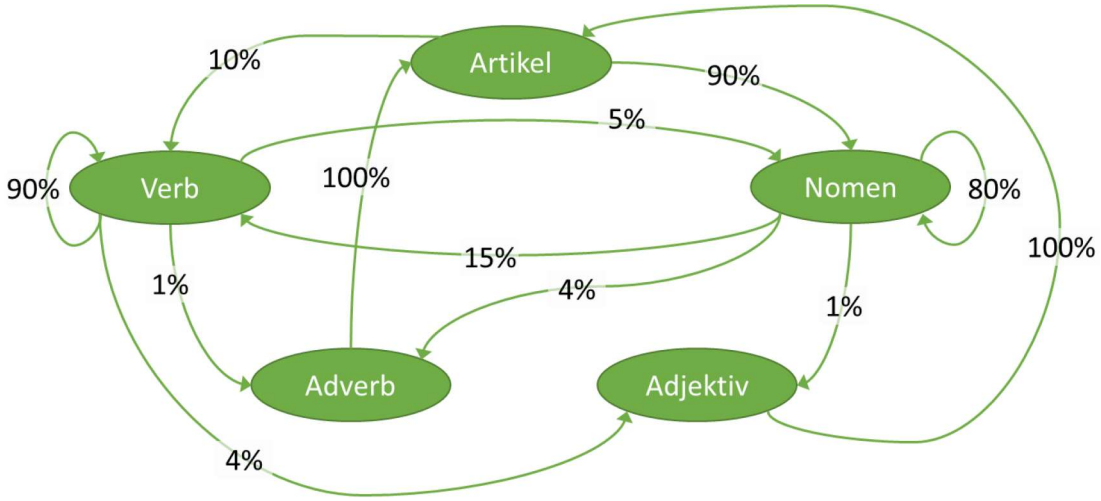

Abbildung 2: Vereinfachtes Markov Modell deutscher Wortarten [12]

In der oben gezeigten Abbildung 2 ist ein vereinfachtes Markov Modell dargestellt, welches die Übergangswahrscheinlichkeiten von verschiedenen Wortarten zu anderen Wortarten angibt[12]. Beim Hidden Markov Modell emittiert ein Zustand zu bestimmten Wahrscheinlichkeiten Wortformen. Der Zustand „Adjektiv" emittiert zum Beispiel *der* mit einer Wahrscheinlichkeit von 40%, *die* mit einer Wahrscheinlichkeit von 30% und *das* ebenfalls mit 30%[12].

Hidden Markov Modelle finden genau wie bei den probalistischen Sprachmodellen bei der Zuweisung von Wortarten Anwendung und gehören damit ebenfalls zu einer Implementierung von POS-Taggern. Neben dem reinen Klassifizieren von Wörtern lassen sich Hidden Markov Modelle außerdem anwenden, um ganze Sätze und Texte maschinell zu erstellen[13].

2.2.5 Kookkurrenz

Bei der Kookkurrenz werden Zusammenhänge zwischen Begriffen hergestellt, die häufig zusammen in sogenannten Textfenstern (meist Sätzen) auftreten. Es werden die *signifikante Satzkookkurrenz* und die *signifikante Nachbarschaftskookkurrenz* unterschieden[14]. Während bei der signifikanten Satzkookkurrenz beide Terme eine enge textuelle

[12] Vgl. Scheffer, T. / Vanck, T.: Hidden Markov Models - HMMs (2010, Folie 4)
[13] Vgl. Szymanski, G. / Ciota, Z.: Hidden Markov Models Suitable for Text Generation (2002, Seite 2ff.)
[14] Vgl. Bußmann, Hadumod (1990, S. 375)

Nachbarschaft aufweisen, ohne dabei direkt nebeneinander zu stehen, stellt die signifikante Nachbarschaftskookkurrenz das stärkere Konstrukt dar, bei dem beide Terme direkt nebeneinander stehen.

Für die Berechnung der Kookkurrenz werden vier Werte benötigt: *a* sei dabei die Anzahl der Sätze, in denen der Term *A* vorkommt, *b* sei die Anzahl der Sätze, in denen *B* vorkommt, *k* sei die Anzahl der Sätze, in denen sowohl *A* als auch *B* vorkommen und *n* sei die Anzahl aller Sätze.

Für die Berechnung der Signifikanz wird nun mit diesen vier Werten unter Zuhilfenahme der Poisson-Verteilung und einer Vereinfachung folgende Formel aufgestellt, auf deren Herleitung an dieser Stelle verzichtet wird[15]:

Sei $\lambda = \frac{ab}{n}$, dann wird die Signifikanz unter der Voraussetzung, dass $\frac{(k+1)}{\lambda} > 2{,}5$ gilt, was in der Regel bei einem hinreichend großen Korpus der Fall ist, folgendermaßen definiert:

Formel 5: Berechnung der Kookkurrenz

$$sig(A,B) = \frac{\lambda - k * \log\lambda + \log k!}{\log n}$$

Durch die Anwendung der Formel auf einen Trainingskorpus mit fünf Millionen Sätzen lässt sich folgende Wertetabelle aufstellen[16]:

A	B	a	b	k	λ	Signifikanz
Romeo	Julia	343	1080	124	0,075	**51,85**
Stadt	Einwohner	37053	2611	272	19,364	**30,47**
Steuergelder	Verschwendung	251	373	54	0,019	**24,58**
Polizei	verhaftet	20550	1928	131	7,924	**16,06**
Dorf	Kirche	3870	8740	64	6,764	**5,81**
Dorf	Pfarrer	3870	2095	15	1,622	**1,44**

[15] Vgl. Foata, D. / Fuchs, A. (1999, S. 32)
[16] Vgl. Heyer, G. / Quasthoff, U. / Witting, T. (2006, S. 140)

Unfall	Krankenhaus	1987	3350	11	1,331	**1,01**
Romeo	Shakespeare	343	612	4	0,042	**1,03**

Tabelle 4: Berechnete Signifikanzwerte für Beispielwortformen [16]

Der Schwellenwert, bei dessen Überschreitung eine Kookkurrenz wirklich signifikant ist, lässt sich nicht eindeutig bestimmen. Bei der Wahl eines höheren Wertes wird die Qualität der Selektion besser, jedoch nimmt gleichzeitig der Zahl der kookkurrierenden Terme deutlich ab. Andersherum verhält es sich bei der Wahl eines kleinen Wertes, bei dem zwar mehr Termpaare identifiziert werden, darunter jedoch mehr lose bis gar nicht zusammenhängende Terme vorhanden sind.

2.2.6 Pattern-Matching

Beim sogenannten Pattern-Matching handelt es sich um eine klassische Methode des Text Minings, die auch noch stark verbreitet ist und wenig mit den modernen Methoden gemeinsam hat, die beispielsweise auf Machine-Learning, neuronale Netze oder künstliche Intelligenz setzen. Bei diesem Verfahren werden Regeln bzw. Muster (Pattern) aufgestellt, nach denen Texte durchsucht werden. Bei einem Treffer (Match) wird der Satz weiter verarbeitet. Es ist auch möglich, Muster zu erstellen, die nicht nur das Vorhandensein prüfen, sondern sich auch für die Extraktion von Termen unter bestimmten Voraussetzungen eignen. Ein Beispiel dazu ist eine Regel, die aus einem Text alle Vorstellungen mit Namen von Personen heraussucht. Die nachfolgenden Regeln stellen exemplarisch eine Möglichkeit dafür dar:

1. Satz enthält einen der folgenden Terme: „ich heiße“, „mein Name ist“, „ich bin“
2. Prüfe Wortarten nach den obigen Termen auf Personennamen
3. Extrahiere alles, was hinter dem gefundenen Term ein Personenname ist

Mit diesen drei einfachen Regeln lassen sich aus Sätzen wie „Hallo, mein Name ist Max Mustermann“ bereits viele Personennamen extrahieren. Vorstellungen wie „Max Mustermann, guten Tag“ werden jedoch nicht gefunden. Zudem kann es auch zu Falschinterpretationen kommen, wenn die Terme in einem anderen Kontext stehen als erwartet. Die Definition von Regeln kann sehr komplex werden, da viele Konstellationen zu beachten sind.

2.2.7 Wortstammreduktion

Eine weitere Vorbereitung, die im Bereich des Text Minings zum Einsatz kommt, ist die Stammformreduktion (engl. „Stemming“). Bei dieser Methode werden Wörter in ihre Stammform transformiert, wodurch mehrere Ausprägungen zu einem gemeinsamen Wortstamm führen. Beispielsweise werden „Wörter“ und „Wortes“ zu „Wort“ dekliniert und „springen“ und „springt“ auf „spring“ reduziert. Durch die Reduktion der Ausprägungen lassen sich gleiche Terme leichter identifizieren und die Auswertung vereinfachen. Wenn beispielsweise in einem Text die Terme „springen“ und „springt“ vorkommen, dann sind beide Terme beispielsweise für eine Wortzählung zusammenzufassen.

Da es sich beim Stemming um eine Vereinfachung bzw. Reduzierung des Wortes handelt, kann es vorkommen, dass verschiedene Wörter auf den gleichen Wortstamm reduziert werden und an der Stelle Informationsgehalt verloren geht. Das Stemming von Eigennamen birgt das Risiko, diese fälschlicherweise zu vereinfachen. Nach den Regeln der Stammformreduktion wird beispielsweise der Vorname „Johannes“ auf „Johann“ abgeändert, was im Kontext von Versicherungsnehmern oder zusätzlich versicherten Personen gravierende Auswirkungen hat.

2.2.8 Tauglichkeitsprüfung auf Anwendungsfall

Im Bereich des Text Minings existieren neben den hier vorgestellten Verfahren noch weitere. Dabei handelt es sich um Methoden, die andere Schwerpunkte, beispielsweise die Emotionsanalyse, setzen. Bevor die aufgeführten Methoden angewandt werden, wird zunächst eine kurze Analyse durchgeführt, um festzustellen, welche zu der Problemstellung passen und sich für die Anwendung eignen. Nicht alle Methoden lassen sich auf alle Szenarien anwenden und wiederum andere weisen Schwächen auf, an denen andere Modelle und Methoden besser zum Einsatz kommen.

Zunächst werden die eigentlichen Daten betrachtet. Dazu werden in der nachfolgenden Tabelle einige typisierende Sätze, wie sie in der Datenbasis vorhanden sind, aufgeführt. Hierbei handelt es sich nur um einen sehr kleinen Ausschnitt. Weitere Beispieldaten sind im Anhang A II aufgeführt. Eine tiefere Analyse der Daten wird im nächsten Kapitel durchgeführt.

#	Satz
1	Mitversichert gilt eine Einbauküche im Wert von 13.000 Euro
2	Versichert gilt 1 Labrador
3	Mitversichert gilt der Hausrat des Sohnes
4	Der sonstige Vertragsinhalt bleibt unverändert.
5	Diverse
6	Berichtigung des Ablaufes
7	Der Dauernachlass ist im Beitrag bereits enthalten.
8	Besondere Bedingungen und Risikobeschreibungen für die Betriebs-Haftpflicht-versicherung für Betriebe des Bauhaupt- und Baunebengewerbes AMEX BHV-LGS BBR Stand 01.07
9	Mietsachschäden durch Brand und/oder Explosion
10	Weitere Deckungserweiterungen/Selbstbeteiligungen

Tabelle 5: Kleiner Teilausschnitt der Datenbasis

Die vorliegenden Daten sind repräsentativ für die Vielfältigkeit der abgelegten Informationen. Es werden unter anderem einfache Kommentare, Ein- und Ausschlüsse, Beitragsinformationen sowie zusätzliche Versicherungsklauseln aufgeführt. Insgesamt führen die Freitextfelder recht kurze Sätze, die eine durchschnittliche Satzlänge von 9,8 Wörtern mit 70 Zeichen aufweisen.

Auf dieser Basis werden nun die vorgestellten Text Mining Methoden auf ihren Nutzen und ihre Einsatzfähigkeit überprüft. Dazu wird das erwartete Ergebnis einer Methode mit dem Nutzen in diesem Szenario verglichen und geprüft, inwieweit die vorliegenden Daten zu der Arbeitsweise passen.

Die **Zipfschen Gesetze** geben Auskunft über Anzahl der Vorkommen und Rang eines Wortes. Anhand der Anzahl lässt sich gegebenenfalls einschätzen, welche Wörter Stopp- oder Füllwörter sind und welche wichtige Erkenntnisse mit sich bringen. Durch die kurzen Sätze besteht jedoch die Gefahr, dass häufig vorkommende Versicherungsterme

ebenfalls als Stopp- oder Füllwörter deklariert werden und somit wichtige Informationen verloren gehen. Deshalb wird dieser Algorithmus keine Anwendung finden, stattdessen wird auf eine einfache Wortzählung zurückgegriffen. Zur Identifikation von Füll- und Stoppwörtern und zu deren Eliminierung wird eine Liste verwendet.

Mithilfe der **Differenzanalyse** lassen sich Fachterme aus einem Text extrahieren. Dazu wird ein Referenzkorpus verwendet, auf dessen Basis ein Vergleich mit den Freitexten durchgeführt werden kann um dort relevante, versicherungstechnische Begriffe oder andere informationstragende Terme zu identifizieren. Diese Begriffe helfen dabei, fachliche Regeln aufzustellen und wesentliche vorkommende Konstellationen besser zu identifizieren und zu verstehen. Dieses Verfahren wird wegen des erwarteten Nutzens angewendet.

Probalistische Modelle und **Hidden Markov Modelle** dienen zur Berechnung der Wahrscheinlichkeit von Wörtern und zur Berechnung von Wortarten. Das Ergebnis, mit welcher Wahrscheinlichkeit ein Satz gültig ist, eignet sich nur bedingt. Es ist denkbar, ihn als Schwellenwert für die Zulassung zur Verarbeitung heranzuziehen. Viel mehr sind die Ergebnisse der Einordnung zu Wortgruppen sinnvoll, da sich mit deren Hilfe beispielsweise Personennamen besser aus den Texten extrahieren lassen. In dieser Arbeit wird eine Implementierung einer der beiden Algorithmen eingesetzt.

Bei der Erstellung der fachlichen Regeln zur Informationsextraktion lassen sich die Ergebnisse der **Kookkurrenzanalyse** heranziehen, um wichtige, stark zusammenhängende Terme zu identifizieren. So ist es beispielsweise naheliegend, dass Konstellationen wie „in Höhe von“ und „Euro“ vermehrt vorkommen oder vereinfacht die Wörter „Höhe“ und „Euro“ eine auffällig starke Nachbarschaftskookkurrenz aufweisen. Diese Analyse kann helfen, weitere solche Verhalte aufzudecken, weshalb auch diese angewendet wird.

Ein Algorithmus, der auf jeden Fall Anwendung finden wird und die eigentliche Arbeit übernimmt, ist der **Pattern-Matching Algorithmus**. Um verwertbare Informationen aus den Texten zu extrahieren und diese für die maschinelle Verarbeitung zur Verfügung zu stellen, werden nach dem Aufstellen der fachlichen Suchmuster die Daten durchsucht.

Bei einem Treffer wird die Information anhand der Regeln herausgesucht und in einer geeigneten Datenstruktur abgelegt.

Die **Wortstammreduktion** reduziert die Komplexität von Sätzen, indem Wörter vereinfacht werden. Auswertungen und Analysen sind durch die verringerte Zahl der Ausprägungen leichter durchzuführen und liefern übersichtliche Ergebnisse. Aus diesem Grund wird die Wortstammreduktion als weiteres Hilfsmittel vor der Durchführung der Analysen angewendet.

3 Konzeption

In diesem Kapitel werden die Datenbasis eingehend analysiert und ein Regelwerk aufgestellt. Anhand des Regelwerkes werden Informationen zu Personen, Beträgen und Merkmalen aus den Freitexten ermittelt. Bei der Analyse des Datenbestandes werden wesentliche Faktoren und Eigenschaften der Freitexte dargestellt sowie im Text Mining verbreitete Algorithmen zur Analyse, wie zum Beispiel eine Wortzählung, und eine Differenzanalyse durchgeführt.

Anhand der bei der Analyse gewonnenen Kenntnisse wird geprüft, welche Schritte der Datenvorbereitung und welche umsetzungsrelevanten Algorithmen für die Erstellung des Prototyps Anwendung finden werden.

3.1 Herkunft der Daten

Die Daten stammen aus einer Tabelle aus dem Sach-Unfall-Haftpflicht-Bereich (SUH). Die Tabelle enthält neben Feldern zur Identifikation der zugehörigen Versicherungspolice und des Versicherungsscheins ein Feld, welches Freitexte bis zu einer Länge von 1200 Zeichen aufnehmen kann. Es handelt sich bei den Daten um vom Sachbearbeiter erfasste Zusatzhinweise, Kommentare oder Erklärungen zu Verträgen und Vertragskomponenten. Durch ein anderes Projekt wurde bereits ein Teil der gesamten Daten analysiert und verarbeitet. Dabei handelt es sich um Freitexte zu Verträgen, die zu Kleinunternehmen wie Friseuren, Kosmetikern und Handwerkern gehören. Die verwendeten Daten werden auch für dieses Projekt herangezogen und dienen als Datenbasis. Ziel des bereits stattgefundenen Projektes war die Ermittlung, ob ein Dauernachlass in dem Versicherungsbetrag bereits enthalten ist oder nicht.

In dieser Arbeit wird sich deshalb auf Inhalte konzentriert, die Risikomerkmale enthalten. Solche sind beispielsweise Informationen zu mitversicherten oder von der Versicherung ausgeschlossenen Objekten, wie besondere Möbel, Hausrat vertragsfremder Personen oder Gebäudeteile. Unter den Daten befinden sich verschiedenste Ausprägungen, welche durch deren alleinige Betrachtung keine weitere Relevanz für die Erhebung relevanter Risikomerkmale haben (z.B. Datensätze wie „Diverse“, „Bitte geben Sie uns die

Wohnfläche bekannt“, „KUMUL BERICHTIGT“ oder „AHB“ für *Allgemeine Versicherungsbedingungen für Haftpflichtversicherungen*). Solchen Datensätzen wird aufgrund des fehlenden Informationsgehaltes keine weitere Beachtung geschenkt.

3.2 Beschaffenheit der Daten

Für die Ermittlung der notwendigen Bereinigungsschritte und vor dem Aufstellen eines Regelwerks, wird hier zunächst die Struktur der Freitexte analysiert. Wiederkehrende Muster und Auffälligkeiten werden gesucht, die Ursache untersucht und Lösungsansätze erarbeitet.

In den Freitexten kommen wiederholte Leerzeichen vor, was auf die Darstellung der Daten in der Eingabemaske des Sachbearbeiters zurückzuführen ist. Die Maske wird in einer Emulation des Großrechners dargestellt, deren Breite auf 70 Zeichen pro Zeile begrenzt ist. Abzüglich eines Darstellungsrahmens von 10 Zeichen beläuft sich der Platz für Texte auf 60 Zeichen. Bei der Speicherung werden Freiräume am Ende der Zeile mit Leerzeichen aufgefüllt. Der Fall, dass ein Text alle verfügbaren Zeichen einer Zeile einnimmt, ist zu beobachten. Dies führt bei der Speicherung dazu, dass Leerzeichen zwischen Wörtern fehlen, wenn die beiden Zeilen aneinandergehängt werden. Die folgende Grafik veranschaulicht den Unterschied der Darstellung und der Speicherung. Zur Veranschaulichung enthält der Text in den Zeilen nur 24 Zeichen und die Leerzeichen werden als Unterstrich dargestellt. Der eben beschriebene Fall der fehlenden Leerzeichen zwischen zwei Zeilen ist in den letzten beiden Zeilen bei den Wörtern „Leistungen“ und „in“ zu sehen.

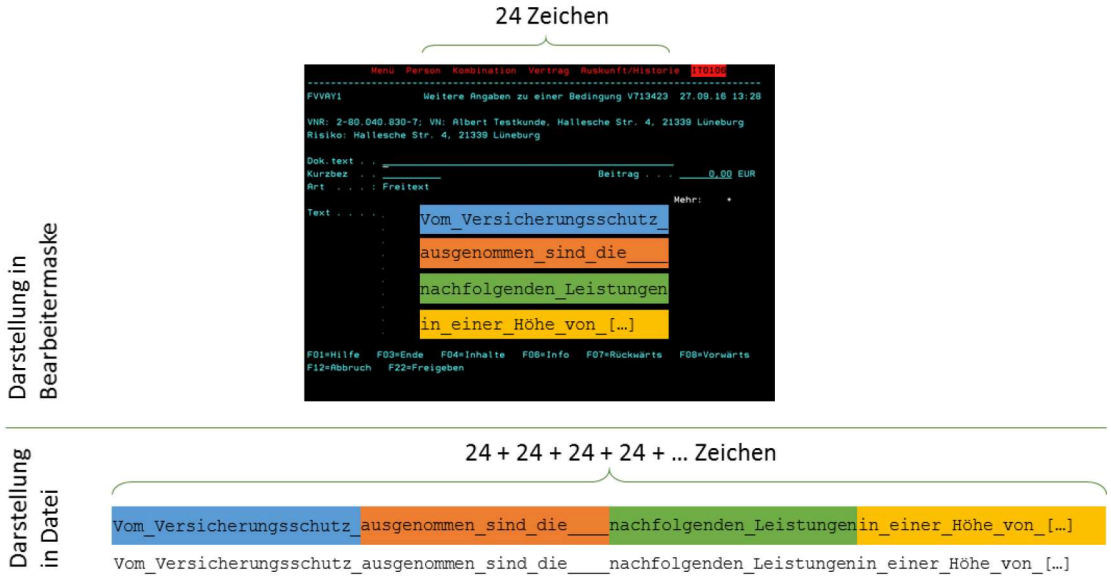

Abbildung 3: Darstellung und Speicherung der Freitexte

In der unten dargestellten Tabelle sind die Durchschnittslänge und die durchschnittliche Anzahl Wörter pro Freitextfeld dargestellt. Wird die durchschnittliche Anzahl der Wörter pro Freitextfeld mit klassischen Werten wie die durchschnittliche Anzahl von Wörtern pro Satz von Romanen (12,98) oder Berichten (16,37)[17] verglichen, fällt auf, dass die Sätze in Freitextfeldern deutlich kürzer sind. Dies ist darauf zurückzuführen, dass diese entweder unvollständig oder sehr einfach aufgebaut sind. Durch Freitexte, in denen nur ein Wort vorkommt, sind die Ergebnisse leicht verfälscht, da diese die durchschnittliche Wortzahl senken.

Anzahl Datensätze	Ø Anzahl Zeichen	Ø Anzahl Wörter	Maximale Anzahl Zeichen	Maximale Anzahl Wörter
5.665	70,33	9,79	1024	163

Tabelle 6: Durchschnitts- und Maximalwerte der Freitexte

Durch die kurze Satzlänge ist zu erwarten, dass die Sachverhalte und Inhalte in den Texten prägnant und auf das Wesentliche reduziert dargestellt sind.

In der nachfolgenden Tabelle sind Beispielsätze aus den Freitextfeldern, die den Term „mitversichert“ oder eine Variation dessen enthalten, aufgeführt.

[17] Vgl. Pieper, U. (1979, S. 50)

#	Satz
1	Versichert gelten Wertsachen im Wert von 78.000 Euro.
2	Mitversichert gilt der Hausrat in Höhe von 10.000,- Euro in einem angemieteten Keller am Leipziger Platz.
3	Mitvers.: Technische Geräte im Wert von 119.000,- DM
4	Die zeitweise gelagerten Lederwaren gelten n i c h t mitversichert.
5	Mitversichert gelten ab 05.09.2002 Mietsachschäden im Rahmen der beigefügten Bedigungen.

Tabelle 7: Beispielsätze die den Teilterm „mitvers“ oder „versichert“ enthalten

Durch das freie Eintragen und Verfassen der Texte treten unter anderem auch Rechtschreibfehler auf, so wie in dem Satz mit der Nummer 5 der Tabelle 7 zu sehen ist. In diesem Satz wurde das Wort „Bedingungen“ falsch geschrieben. Im Satz mit der Nummer 4 wurde die Negation durch das Wort *nicht* so verdeutlicht, dass die einzelnen Buchstaben durch Leerzeichen getrennt wurden. Rechtschreibfehler oder andere syntaktische Fehler führen bei der Analyse zu Problemen oder gar zu Falschinterpretationen. Dennoch werden solche Probleme aufgrund des Umfanges dieses Themas in dieser Arbeit nicht weiterverfolgt. Ein möglicher Ansatz zur Herangehensweise an dieses Problem wird in Kapitel 4.6 beschrieben.

3.3 Inhaltliche Analyse

Vor der der inhaltlichen Analyse sind die Daten mit den in Kapitel 4.2 beschriebenen Methoden bereinigt und aufbereitet worden. Überflüssige Leerzeichen und fälschlicherweise zusammenhängende Wörter wurden aufgelöst, Stoppwörter entfernt und eine Stammformreduktion durchgeführt. Für die nachfolgenden Analysen wurden zudem ähnliche Terme gebündelt und unter einem Term harmonisiert. Beispielsweise werden die drei Terme „versichert“, „mitversichert“ und „mitvers.“ unter dem Term „mitversichert“ gebündelt. Ein weiteres Beispiel ist das Währungskennzeichen Euro, welches neben der ausgeschriebenen Form sowohl in der Kurzform „EUR“, als auch nur als Euro-Zeichen enthalten ist.

Die in Tabelle 8 dargestellten Ergebnisse zeigen die absolute und die relative Häufigkeit der 40 meist auftretenden Terme absteigend sortiert nach deren Häufigkeit. Die relative

Häufigkeit bezieht sich auf die Vorkommen in Sätzen des verarbeiteten Gesamtkorpus, der aus 55.464 Wörtern in 5.665 Sätzen besteht. Die Terme „mitversichert“, „dauernachlass“ und „beitrag“ treten am häufigsten auf. Werden nun Sätze begutachtet, in denen diese Terme vorkommen, fallen mehrere Regelmäßigkeiten auf: Sätze, in denen „versichert“, „mitversichert“ oder „mitvers.“ vorkommen, beschreiben Objekte, die zusätzlich unter den Versicherungsschutz fallen oder explizit von diesem ausgeschlossen werden. Sätze, die den Term „dauernachlass“ enthalten, beschreiben, dass der Dauernachlass[18] im Versicherungsbeitrag enthalten ist. Dieser Term tritt zusammen mit dem Term „Beitrag“ auf.

#	Term	Abs.	Rel.	#	Term	Abs.	Rel.
1	mitversichert	1856	0,328	**21**	gesetz	461	0,081
2	dauernachlass	1743	0,308	**22**	versicherungsschutz	436	0,077
3	beitrag	1086	0,192	**23**	piercing	421	0,074
4	nicht	944	0,167	**24**	haftpflicht	418	0,074
5	schad	830	0,147	**25**	ziff	401	0,071
6	enthalt	821	0,145	**26**	jahresbeitrag	396	0,070
7	gilt	783	0,138	**27**	faltenunterspritz	396	0,070
8	kein	773	0,136	**28**	gelt	386	0,068
9	besond	710	0,125	**29**	haarentfern	384	0,068
10	vorhand	645	0,114	**30**	dauerhaft	384	0,068
11	risik	627	0,111	**31**	permanent-make-up	378	0,067
12	umweltris	624	0,110	**32**	ausgewies	373	0,066
13	deckungskonzept	615	0,109	**33**	tattoo	368	0,065
14	vertrag	612	0,108	**34**	rahm	364	0,064
15	versicherungsnehm	597	0,105	**35**	reduziert	294	0,052
16	beding	595	0,105	**36**	vereinbart	286	0,050
17	anspruch	564	0,100	**37**	genannt	276	0,049
18	hoh	530	0,094	**38**	betrieb	246	0,043
19	eur	519	0,092	**39**	gmbh	242	0,043
20	berucksichtigt	517	0,091	**40**	abweich	226	0,040

Tabelle 8: Wortzählung mit vorheriger Anpassung der Wörter

Neben der einfachen Wortzählung und der Sortierung nach Häufigkeit wird eine Differenzanalyse, wie in Kapitel 2.2.2 beschrieben, durchgeführt. Für den Referenzkorpus

[18] Der Dauernachlass kennzeichnet einen dauerhaften Rabatt einer Versicherungspolice, den der Sachbearbeiter individuell für Verträge gewähren darf.

wurde ein Aufsatz von Georg Dehio[19] verwendet. Bei dem Vergleich der Freitexte mit dem Referenzkorpus stellt sich folgende Termliste auf:

Term	Rel. Basis	Rel. Ref.	Differenz	Klasse
mitversichert	0,03346	0,00000	0,03346	Klasse 1
dauernachlass	0,03143	0,00000	0,03143	Klasse 1
risik	0,01131	0,00000	0,01131	Klasse 1
umweltris	0,01125	0,00000	0,01125	Klasse 1
deckungskonzept	0,01109	0,00000	0,01109	Klasse 1
versicherungsnehm	0,01076	0,00000	0,01076	Klasse 1
			Klasse 1: insgesamt	2027
beitrag	0,01958	0,00005	0,01953	Klasse 2
schad	0,01497	0,00007	0,01489	Klasse 2
enthalt	0,01480	0,00020	0,01461	Klasse 2
gilt	0,01412	0,00020	0,01392	Klasse 2
besond	0,01280	0,00126	0,01154	Klasse 2
er	0,00020	0,01164	0,01144	Klasse 2
			Klasse 2: insgesamt	90
eur	0,00936	0,00005	0,00931	Klasse 3
hoh	0,00956	0,00076	0,00879	Klasse 3
gesetz	0,00831	0,00047	0,00784	Klasse 3
noch	0,00061	0,00814	0,00752	Klasse 3
ziff	0,00723	0,00003	0,00721	Klasse 3
gelt	0,00696	0,00032	0,00664	Klasse 3
			Klasse 3: insgesamt	696

Tabelle 9: Ergebnisse der Differenzanalyse

In der Tabelle sind neben dem Term die relative Häufigkeit des Vorkommens in den Freitexten, die relative Häufigkeit des Terms in dem Referenzkorpus, die Differenz beider Werte und die Einteilung in die jeweilige Klasse aufgeführt. Der Schwellenwert für die

[19] Vgl. Dehio, G.: Kunsthistorische Aufsätze

Unterscheidung nach Klasse 2 und 3 wurde auf 0,001% festgelegt, da sich nur so eine ausreichend große Menge an Elementen in Klasse 2 befindet. Auf die Auswertung und Darstellung von Elementen der Klasse 4 wurde verzichtet, da diese keine Relevanz besitzen. Pro Klasse sind die ersten sechs Elemente nach deren Differenz absteigend sortiert aufgelistet.

Besonders relevant sind die Terme der Klasse 1, also die Terme, welche nicht im Referenzkorpus vorkommen. Die Klasse enthält besonders unter den ersten 60 Termen viele versicherungsspezifische Begriffe. Kosmetische Begriffe wie „piercing“, „tattoo“ und ähnliche tauchen wie in der einfachen Wortzählung ebenfalls auf. Durch die Ergebnisse der nachfolgenden Kookkurrenzanalyse lässt sich feststellen, dass diese Dienstleistungen in den Versicherungsschutz mit aufgenommen oder von diesem ausgeschlossen sind.

Die nachfolgende Abbildung stellt die Ergebnisse der Kookkurrenzanalyse grafisch dar. Die Linien zwischen den Termen zeigen auffällig oft auftretende bzw. starke Zusammenhänge zwischen den Termen. Je dicker die Linie, desto häufiger treten beide Wortvorkommen im gleichen Satz auf. Die Anzahl der aufgetretenen Verbindungen ist absolut angegeben und es werden der Übersicht halber nur Verbindungen dargestellt, die häufiger als 200 Mal aufgetreten sind. Einen Ausschnitt der gesamten Matrix, sowie eine weitere grafische Darstellung, befinden sich im Anhang A III und 0.

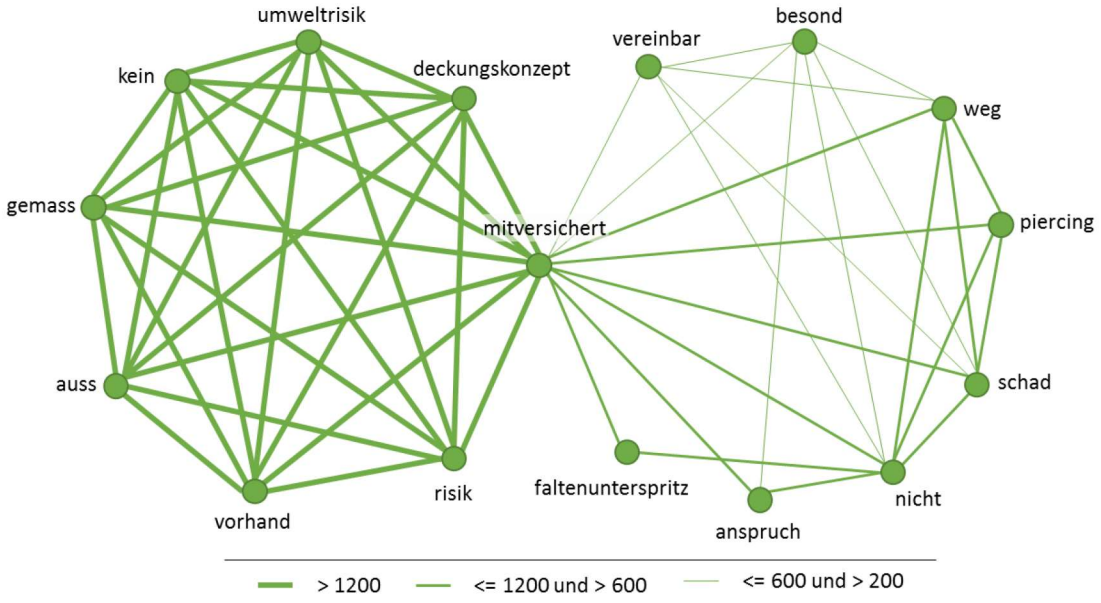

Abbildung 4: Grafische Darstellung von kookkurrierenden Termen

Gut zu erkennen ist, dass bestimmte Terme immer in direkter Kombination vorkommen. Es bilden sich zwei disjunkte Cluster, die untereinander nur durch den Term „mitversichert" zusammenhängen. Die Bildung der Cluster ist die Folge von ähnlichen Formulierungen und der damit verbundenen Begriffauswahl.

Für die Umsetzung des Prototyps zur automatisierten Extraktion von Risikomerkmalen werden die Begriffe „Faltenunterspritzung", „Piercing", „dauerhafte Haarentfernung", „Permanentmakeup" und „Tattoo" gewählt, da diese neben anderen mit dem Term „mitversichert" kookkurrieren und die meistvertretenden Risikomerkmale im verwendeten Datenbestand sind. Dies wurde bereits in der Wortzählung festgestellt und durch die Kookkurrenzanalyse bestätigt. Die stichprobenartige Betrachtung von Freitexten, die diese Terme enthalten, bestätigt die Annahme, dass diese Leistungen vom Versicherungsschutz ausgenommen sind. Zwei Beispielwortlaute solcher Texte sind nachfolgend dargestellt.

> Nicht versichert sind Ansprüche wegen Schäden durch Tattoo, Piercing, Faltenunterspritzung, Permanent-Make-Up und dauerhafte Haarentfernung.

> Kein Versicherungsschutz besteht für Ansprüche im Zusammenhang mit Permanent-Make-Up, Tattoo, Piercing und Faltenunterspritzung.

In beiden Fällen sind die genannten Dienstleitungen vom Versicherungsschutz explizit ausgenommen. Im ersten Fall wird dies durch den Zusatz des Terms „nicht“ dargestellt, welches die häufiger vertretene Variante ist. Im zweiten Fall besteht die Formulierung aus dem Term „kein“ in Verbindung mit dem Term „Versicherungsschutz“.

Als weiterer Ansatz werden auch die kookkurrierenden Terme zu den beiden Wörtern „versichert“ und „mitversichert“ betrachtet. Darunter befinden sich neben den oben genannten noch die Terme „Deckungskonzept“, „Risik[en]“, „gesetz[liche]“, „Haftpflicht“, „Versicherungsnehm[er]“. Da die Kookkurrenzanalyse nur auf Wortstämme statt auf ganze Wörter angewendet wurde, befinden sich in der Liste nur die Wortstämme. In den eckigen Klammern wurde deshalb die jeweilige Form des Wortes ergänzt. Die folgenden drei Boxen geben Einblicke in typische Formulierungen, bei denen in den Freitexten der Term „Haftpflicht“ vorkommt. Aus Datenschutzgründen wurden die Personendaten durch Platzhalter in geschweiften Klammern ersetzt.

> Versichert ist die gesetzliche Haftpflicht von Herrn {Vorname} {Nachname} als Pächter und Betreiber der Gaststätte.

> Nicht versichert ist die gesetzliche Haftpflicht von Herrn {Vorname} {Nachname} als Eigentümer des Haus- und Grundbesitzes.

> Mitversichert gilt im Umfang und Rahmen des Vertrages die Bauherrenhaftpflichtversicherung für eigene Bauvorhaben (Sanierung/Modernisierung/Anbauten des

In den ersten beiden Fällen wird die gesetzliche Haftpflichtversicherung von zwei Personen einmal in den Versicherungsschutz eingeschlossen und einmal davon ausgeschlossen. Im dritten Fall wird eine Bauherrenhaftpflichtversicherung zum normalen Vertragsumfang und Versicherungsschutz ergänzt. Die Angabe von Personen, insbesondere in

Bezug auf gesetzliche Haftpflichtversicherungen, tritt vermehrt auf. Deshalb wird in diesem Zusammenhang auch der Personenname durch den Prototyp extrahiert, um Abläufe des Sachbearbeiters zu vereinfachen und präzisere Prüfungen durchzuführen. Die Person steht in diesem Zusammenhang als Ergänzung zu einem Merkmal und kann selbst kein Merkmal sein. Es kommt vor, dass neben dem Namen der Person die Beziehung des Versicherungsnehmers weiter spezifiziert wird, indem beispielsweise die Schlüsselbegriffe „Ehepartner“, „Tochter“ o.ä. verwendet werden. Die Beziehung ist, soweit vorhanden, ebenfalls durch den Prototyp zu ermitteln. Eine Möglichkeit der Ablage ist die Speicherung der Person in dem Partnersystem des Unternehmens, indem alle zu dem Unternehmen stehenden Partner stehen. Dies sind beispielsweise Versicherungsnehmer, Vertriebler, Kunden oder Lieferanten. Es kann zudem eine Recherche gemacht werden, ob die erwähnte Person bereits im System enthalten ist, um eine bessere Verknüpfung herzustellen. Da die Angabe eines Personennamens jedoch nicht eindeutig ist, wird eine konkrete Zuweisung kein triviales Problem darstellen. Im Rahmen dieser Arbeit geschieht keine solche Zuweisung und Verknüpfung mit dem bestehenden Partnersystem.

Neben den Risikomerkmalen und Personen sind in manchen Freitexten zudem Geldbeträge angegeben. Dabei handelt es sich um Summen, bis zu denen ein Versicherungsschutz gegeben ist. Zwei solcher Fälle sind in den folgenden Boxen dargestellt.

Mitversichert gilt ein Lagerraum in der Tiefgarage mit einer Versicherungssumme von 1.500,00 EUR.

Mitversichert gelten im Rahmen und Umfang des Vertrages Schäden an von beherbergten Gästen eingebrachten Sachen bis 2.500,- EUR.

Die Schreibweise solcher mitgelieferten Beträge ist sehr heterogen und wird im Kapitel 3.4.3 genauer beschrieben. Für die Regulierung eines Schadens, also die Bewertung und letztendlich Auszahlung und Begleichung der entstandenen Schäden, sind hinterlegte Beträge relevant, da diese von den normalerweise vertraglich geregelten Beträgen abweichen können. Die Informationen zu diesen Beträgen lassen sich zur Überprüfung und

Festlegung von Auszahlungsgrenzen oder Beschränkungen heranziehen und sind ebenfalls durch den Prototyp zu verarbeiten.

Anhand dieser drei Anforderungen wird sich die Erstellung des Prototyps orientieren und es werden alle nötigen Schritte umgesetzt. In den nächsten Unterkapiteln sind für jeden Anwendungsfall eine Ausformulierung und Definition der zu erledigenden Schritte beschrieben. Mit dieser Ausprägung kann nur ein kleiner Teil aller abgelegten und sehr verschiedenen Informationen automatisiert verarbeitet werden. Die Informationen lassen sich aufgrund der Vielfältigkeit nicht durch einen einzelnen Algorithmus gänzlich verarbeiten. Auch stellt die Ablage und damit die Bereitstellung zur computergestützten Verarbeitung ein Problem dar. Es lassen sich immer nur einzelne Anwendungsfälle betrachten und für sich umsetzen.

3.4 Festlegung signifikanter Merkmale

Zunächst werden nur Sätze verwendet, die einen der folgenden Terme enthalten: „versichert“, „mitversichert“ oder „mitvers.“. Anschließend wird das Objekt selbst aus dem Satz extrahiert. Hierbei kann es sich um Dienstleitungen oder Objekte handeln.

Es wird geprüft, ob Personennamen vorkommen. Für den Fall wird ferner geprüft, ob eine der bekannten Personenkategorien (z.B.: Ehepartner, Kind, Tochter, Sohn, o.ä.) vorkommt, um die Person näher zu beschreiben. In dem Fall, dass kein Personenname in dem Satz enthalten ist, wird geprüft, ob eine der Personenkategorien vorkommt. So kann sichergestellt werden, dass auch namentlich nicht genannte Personen identifiziert werden.

3.4.1 Extraktion von Risikomerkmalen

Um Risikomerkmale zu identifizieren, die in den Versicherungsschutz ein- oder ausgeschlossen sind, wird nach definierten Wörtern gesucht, die auf einen Einschluss oder Ausschluss eines Merkmals hinweisen. Dies sind Terme wie „mitversichert“ oder „versichert“, aber auch Termkombinationen wie „[…] vom Versicherungsschutz ausgeschlossen […]“. Der entsprechende Satz wird auf die Existenz eines Merkmals überprüft und das Merkmal ermittelt. Mögliche Merkmale werden in einer Liste geführt, die neben den

zugehörigen Termen auch Termabhängigkeiten enthält. Die Abhängigkeiten helfen dabei komplexe Strukturen, wie die obige, darzustellen.

Für die Ermittlung, ob es sich um einen Ein- oder Ausschluss handelt, wird nach bestimmten Begriffen gesucht. Die Existenz des Wortes „nicht“ oder „ausgeschlossen“ weisen auf einen Ausschluss hin. Das Wort „Ausschluss“ selbst kommt in den Freitexten nicht.

3.4.2 Extrahieren von Beträgen

Neben den Informationen zu den im Versicherungsschutz ein- oder ausgeschlossenen Objekten werden optional Deckungswerte zu den Objekten selbst angegeben. Diese tauchen in Verbindung mit den Wörtern „in Höhe von“ oder Ähnlichen und einer Dezimalzahl im deutschen Format auf, was bedeutet, dass als Tausendertrennzeichen der Punkt (.) und als Dezimaltrennzeichen das Komma (,) verwendet wird (beispielsweise *12.345,67*). Die zusätzlichen Werte werden ebenfalls ermittelt und gespeichert. In der Kookkurrenzmatrix tauchen zudem die Terme Deckungserweiter[ung] (117 Mal) und Versicherungssumm[e] (67 Mal) in Kombination mit dem Term Euro auf.

Die Extraktion von angegebenen Versicherungsgrenzen und der Wert des mitversicherten Objektes lassen sich durch die Verwendung von dem Konstrukt „in Höhe von“, „i. H. v.“ oder durch Angabe des Währungskennzeichens identifizieren. Die Währung wird dabei in Euro oder Deutsche Mark angegeben, letztere lediglich in der Kurzschreibweise DM. Das Währungszeichen Euro wird in der standardisierten Form EUR, vereinzelt als Euro-Zeichen „€“ oder ausgeschrieben dargestellt. Eine Suche nach der ebenfalls gängigen Schreibweise mit dem Halbgeviertstrich hinter dem Komma brachte keine Ergebnisse hervor. Stattdessen ist der Halbgeviertstrich durch die vereinfachte Schreibweise mit dem Minus-Zeichen ersetzt. Insgesamt lassen sich die in der nachfolgenden Tabelle aufgeführten Konstellationen feststellen.

Art	Darstellung	Eindeutige Identifizierung
Schreibweise mit Tausendertrennzeichen	XX.XXX,YY	Ja
Schreibweise ohne Tausendertrennzeichen	XXXXX,YY	Nein
Schreibweise mit Tausendertrennzeichen mit Halbgeviertstrich (oder Minus)	XX.XXX,–	Ja

Schreibweise ohne Tausendertrennzeichen mit Halbgeviertstrich (oder Minus)	XXXXX,–	Ja
Schreibweise mit Währungskennzeichen	XXX,YY EUR XXX,YY € XXX,YY Euro XXX,YY DM	Ja
Schreibweise ohne Dezimalstellen, Tausendertrennzeichen und Währungskennzeichen	XXXXX	Nein

Tabelle 10: Unterschiedliche Darstellungen von Geldbeträgen

Bei der Suche nach Beträgen ist zu berücksichtigen, dass auch andere Zahlen, beispielsweise Prozentwerte, in den Texten auftauchen. Ist im Text unmittelbar davor die Terminologie „in Höhe von“, „im Wert von“ oder „bis zu [einem Betrag von]“ zu finden, handelt es sich definitiv um einen Betrag. Ein weiterer Indikator dafür ist die Zusammensetzung der Zahl: Wenn entweder die Währung, ein oder mehrere Tausendertrennzeichen oder der Halbgeviertstrich (oder Minus) angegeben sind, kann ebenfalls davon ausgegangen werden, dass der betrachtete Fall einen Betrag darstellt. Schwieriger ist es bei der Darstellung lediglich mit Nachkommastellen, beispielsweise der Zahl 1,23 oder 12345,67, da es sich hier beispielsweise um Prozentangaben handeln kann. Sind keine näheren Angaben zu finden, kann nicht mit ausreichender Wahrscheinlichkeit davon ausgegangen werden, dass die Zahl einen Betrag darstellt. In dieser Konstellation kann eine Sicherstellung nur durch eine darauf hinweisende Terminologie geschehen. Für alle anderen Fälle ist die Erfüllung mindestens einer der drei oben beschriebenen Bedingungen notwendig.

3.4.3 Extraktion von Personen und Beziehungen

Um Personen und deren Beziehung zum Versicherungsnehmer aus den Freitexten zu erhalten, wird vor dem Prozess der Stammformreduktion ein POS-Tagging durchgeführt. Dieser Schritt ist zwingend vor der Stammformreduktion auszuführen, da sonst für den Algorithmus wichtige und benötigte Informationen verloren gehen. Für die Bestimmung der Wortform reicht die auf den Stamm reduzierte Form nicht aus. Es wird erwartet, dass Personennamen mit eigenen Tags, beispielsweise *NAME* oder ähnliches gekennzeichnet werden. Diese Namen werden zur späteren Verarbeitung separat gespeichert.

Für die Ermittlung der Beziehung zu den Personen wird neben dem Namen nach bestimmten Schlüsselbegriffen wie Tochter, Sohn, Lebenspartner, Ehefrau, Ehemann etc. gesucht. Bei einem Fund kann die identifizierte Person direkt in eine Beziehung mit dem Versicherungsnehmer gesetzt werden.

4 Implementierung eines Prototyps

In diesem Kapitel werden die gewonnenen Kenntnisse eingesetzt, um daraus einen Prototyp zu erstellen, der die dieser Arbeit zugrundeliegende Aufgabenstellung löst, um daraus Erkenntnisse für spätere Projekte innerhalb dieses Themenbereiches zu gewinnen. Für Verträge sind relevante Risikomerkmale aus den Freitextfeldern zu extrahieren. Dafür wird zunächst beschrieben, wie die Daten aus der Quelle in das verarbeitende System transferiert werden. Die Aufbereitung und Vorbereitung der Daten zur Verarbeitung werden im zweiten Schritt erläutert. Anschließend werden die Schritte, welche bei der Erstellung des Prototyps durchgeführt werden, dargestellt und die Möglichkeiten einer Rückführung der angereicherten Daten in das ursprüngliche Quellsystem beleuchtet. Außerdem wird die korrekte Arbeitsweise des Prototyps mit Hilfe von Tests überprüft. Abschließend werden Erweiterungs- und Anpassungsmöglichkeiten für spätere Versionen aufgezeigt.

4.1 Datenbeschaffung und Integration

Die Verarbeitung der Daten wird nach dem Konzernstandard in Java implementiert, weshalb für die Ausführung eine virtuelle Java-Maschine Voraussetzung ist. Die Entwicklung geschieht auf einem Desktop-PC, auf dem das Java Development Kit (JDK) in der Version 8, Update 31 installiert ist.

Die Daten stammen aus dem Datenbanksystem DB2, welches durch die Firma IBM entwickelt und vertrieben wird. Die Instanz dieses Datenbanksystems befindet sich auf einem z/OS-System, welches auf einem Großrechner, dem Mainframe, läuft. Beide Komponenten werden ebenfalls durch IBM entwickelt.

Die Datenextraktion und das Einlesen in das Programm werden im Normalfall direkt über den JDBC-Treiber geschehen, mit dem die Java Virtual Machine (JVM) direkt eine Datenbankverbindung herstellen kann. Aus Gründen der Datensicherheit jedoch werden diese Daten zunächst in eine CSV-Datei (Comma Separated Values) geladen, um von

dort aus weitere Verarbeitungen vorzunehmen. Für die Extraktion wird das ETL[20]-Tool IBM InfoSphere DataStage in der Version 9.1 verwendet, welches sowohl Konnektoren für Datenbanken, als auch für das Schreiben in CSV-Dateien zur Verfügung stellt. Mithilfe dieses Tools lassen sich zudem die Daten bereits filtern und einige Bereinigungen, welche im nächsten Kapitel besprochen werden, durchführen. IBM InfoSphere DataStage wird auf einem AIX-Server betrieben. Dieser und der Mainframe befinden sich im eigens durch die GDIS betriebenen Rechenzentrum in Aachen.

Es werden zunächst nur Freitextfelder zu aktuellen Vertragsständen von Kleinunternehmen der Branchen Friseur, Kosmetik und Handwerk verwendet, während historische Datensätze vernachlässigt werden.

Die erzeugte CSV-Datei wird mittels FTP vom Server auf den Client transferiert und dort von dem Java-Programm verarbeitet. Während der Entwicklung und Verarbeitung sind verschiedene serialisierte Zwischenergebnisse erzeugt worden, die dazu dienen, wiederholte Berechnungen zu vermeiden.

4.2 Aufbereitung der Daten

Bevor die Datenverarbeitung durchgeführt wird, sind die Daten so vorzubereiten, dass die Text Mining-Algorithmen besser funktionieren. Im Folgenden werden die Schritte der Vorbereitung dargestellt und erläutert.

4.2.1 Datenglättung

Die Qualität der Daten spielt eine entscheidende Rolle und deshalb sind bestimmte Dinge zu bereinigen. So befinden sich in den vorliegenden Daten nicht zu gebrauchende Zeichen wie Whitespaces (mehrere aufeinanderfolgende Leerzeichen) oder es fehlen Leerzeichen. Dies tritt durch die, in Kapitel 3.1 beschriebene, Darstellung der Freitexte in den Masken des Sachbearbeiters auf.

Es gilt, solche zu entfernen bzw. durch einzelne Leerzeichen zu ersetzen, um einen möglichst glatten Textfluss zu erzeugen. Für die Analyse von Texten werden häufig zunächst

[20] ETL-Tools sind darauf ausgelegt, Daten aus verschiedenen Quellen zu extrahieren, diese zu transformieren und anschließend wieder in ein Ziel zu laden. Der Prozess wird als Extraction Transforming Loading bezeichnet.

Sätze anhand der Interpunktion voneinander getrennt und anschließend in einzelne Wörter separiert. Trennzeichen dafür ist das einfache Leerzeichen. Für das Entfernen von Whitespaces bietet Java keine direkte Funktion an. Durch die Verwendung der Funktion `replaceAll()` in Kombination mit dem regulären Ausdruck `\s+` ist es möglich, mehrere aufeinanderfolgende Leerzeichen zu eliminieren.

Die virtuelle Maschine von Java ist grundsätzlich Case-Sensitive, was bedeutet, dass Groß- und Kleinschreibung unterschieden werden. Zur Vereinfachung der Anwendung von Funktionen und Algorithmen werden alle Zeichen durch das kleingeschriebene Pendant ersetzt. So sind auch am Satzanfang stehende und großgeschriebene Wörter, die normalerweise klein geschrieben werden, einheitlich und lassen sich beispielsweise bei der Wortzählung einfach zusammenfassen. Die Konvertierung wird mit der integrierten Java-Funktion `toLower()` der String-Klasse erreicht. Einige Algorithmen, wie die Satzdetektion verwenden die ursprüngliche, nicht kleingeschriebene, Version des Satzes.

Je nach verwendetem Zeichensatz haben Algorithmen mit den deutschen Umlauten und dem Buchstaben „ß“ Probleme, weshalb diese als weitere Vereinheitlichung durch die jeweilige Langschreibweise (z.B. ä=ae, ß=ss) ersetzt werden. In den Texten tauchen Wörter sowohl nach alter, als auch nach neuer deutscher Rechtschreibung auf.

4.2.2 Stammformreduktion

Die Stammformreduktion wird lediglich zu Analysezwecken durchgeführt, da bei der Extraktion der Risikomerkmale nicht die auf ihren Wortstamm reduzierte Version verwendet wird. Zu Wortzählungszwecken oder zur Ausführung einer Kookkurrenzanalyse sind die vereinfachten Terme jedoch hilfreich bei der Reduzierung der Komplexität.

Für dieses Projekt wird der Porter-Stemmer-Algorithmus herangezogen, welcher ursprünglich von Martin Porter für die englische Sprache entwickelt wurde. Die Portierung in andere Sprachen, unter anderem für die deutsche, ist frei im Internet erhältlich[21]. Die in Java implementierte deutsche Version wird in diesem Projekt verwendet.

[21] Vgl. Miles, P.: Germanic language stemmers - Snowball

4.2.3 Entfernen von Stopp- und Füllwörtern

Das Entfernen von Stopp- und Füllwörtern dient ebenfalls zur Vereinfachung der Datenbasis. Wörter, die keine Bedeutung für die Analyse haben, da sie keinen Inhalt mit sich führen und eine Auswertung erschweren, sind aus den Sätzen zu entfernen. Die Entfernung geschieht durch ein einfaches Ersetzen aller Stoppwörter durch eine leere Zeichenkette.

Eine Liste der durch das Modul entfernten Stoppwörter wird im Anhang aufgeführt und basiert auf der Ausführung von Tim Ehling[22]. Die erste Version basiert auf der durch das Programm „R“ zur Verfügung gestellten Funktion *stopwords("german")*. Diese ist jedoch unvollständig, was im Laufe der Arbeit zu Verzerrungen der Ergebnisse führte. Die von Tim Ehling zur Verfügung gestellte erweiterte Sammlung enthält außerdem die Version von Wörtern, bei denen die Umlaute durch ihre Langschreibweise ersetzt wurden (z.B. ö = oe, ß = ss).

4.3 Erstellung eines Prototyps

Zunächst werden die Daten aus einer CSV-Datei eingelesen. Diese sind bereits gefilterte Sätze und enthalten nur noch die Freitextdaten mit den Vertragsnummern. Die ersten drei Spalten enthalten die Prüfziffer, Vertragsnummer und das Unternehmenskennzeichen des aktuellen Vertrages. Die vierte Spalte enthält den Inhalt eines Freitextfeldes zu einem Vertrag und kann mehrere Sätze enthalten. Nach der Bereinigung und Aufbereitung des Datensatzes wird dieser in die einzelnen Sätze aufgeteilt, um die Daten satzweise zu verarbeiten und Wörter in einen Kontext zu setzen. Für die Aufteilung wird aus dem Open-NLP-Package der Apache Software Foundation[23] die Klasse *SentenceDetector* verwendet, welche mit der Bibliothek *de-sent.bin* bereits auf die deutsche Sprache ausgelegt ist. Aus dem Package werden auch noch weitere Konfigurationen verwendet, die in der folgenden Tabelle dargestellt sind.

[22] Vgl. Ehling, T.: Deutsche Stoppwort-Liste
[23] Vgl. The Apache Software Foundation: Apache Download Mirrors

Modul / Beschreibung	Konfigurationsdatei	Beschreibung
SentenceDetector	de-sent.bin	Konfiguration für deutsche Sätze
Trainiertes Modell zum Erkennen von Sätzen.		
Tokenizer	de-token.bin	Konfiguration für deutsche Wörter
Modell zum Aufteilen von einzelnen Sätzen nach den Satzelementen wie Wörter oder Zahlen.		
POSTagger	de-pos-maxent.bin de-pos-perceptron.bin	Konfiguration für deutsche Wörter
Modelle zur Erkennung von Wortarten, um ermittelte Token in einem Satz mit dem jeweiligen Tag zu versehen.		

Tabelle 11: Module und Konfigurationen des OpenNLP-Packages

Der Satzdetektor übernimmt die Arbeit des Aufteilens eines Textes in einzelne Sätze. Die ermittelten Sätze werden unabhängig voneinander betrachtet und verarbeitet. Der Satz wird mit Hilfe des *Tokenizers* in seine einzelnen Teile (Token) aufgeteilt. Die Token sind Wörter, Zahlen oder sonstige Elemente und die Rückgabe erfolgt als Array, in dem jeder Token einen Eintrag darstellt. Auf Basis dieses Arrays arbeitet der *POS-Tagger*, der die Wortarten und Typen der Elemente ermittelt. Es stehen zwei mögliche Konfigurationen für den POS-Tagger zur Auswahl, die Maxent-Konfiguration und die Perceptron-Konfiguration. Beide wurden unterschiedlich trainiert und in dieser Arbeit verglichen. Beide Algorithmen liefern keine hundertprozentige Genauigkeit, letzterer liefert im Vergleich zur Maxent-Konfiguration sogar deutlich schlechtere und unbrauchbare Ergebnisse. Die Algorithmen liefern neben der Vorhersage auch einen Wahrscheinlichkeitswert für die Aussage zur Wortart und damit einen Indikator zur Überprüfung der Tauglichkeit. Dieser Wert wurde neben Stichproben für den Vergleich der beiden herangezogen. Während die Perceptron-Konfiguration in vielen Fällen eine Wahrscheinlichkeit von bis zu 96%

und im Durchschnitt 87,8% erzielt, liegt die der Maxent-Konfiguration mit 36,2% lediglich knapp über einem Drittel. Dazu wurden über 26.000 Wörter aus ungefähr 1.200 Sätzen analysiert, die den Term „mitversichert“ oder ähnliche enthalten.

4.3.1 Umsetzung der Extraktion von Risikomerkmalen

Die Implementierung der Risikomerkmalsfindung wird anhand regelbasierter Abfragen umgesetzt, welche auf den durchsuchten Satz angewendet werden. Es wird für jeden Satz, der Risikomerkmale enthält, alle Nomen mit einer Merkmalstabelle abgeglichen. Bei einem Treffer wird geprüft, ob es sich um einen Ausschluss oder Einschluss des Merkmals handelt. Dazu wird nach einer Negation gesucht, welche durch die Terme „nicht“, „ausgeschlossen“ oder „kein“/“keine“ dargestellt wird. Die Informationen werden in einer Tabelle abgelegt.

Um einen Satz nach bestimmten Terminologien zu durchsuchen wird ein Konstrukt erzeugt, in dem Wörter, deren Synonyme und Abhängigkeiten abgelegt werden, nach denen gesucht werden kann. Das Konstrukt ist in zwei Java-Klassen aufgeteilt, eine Klasse *Term* und eine Klasse *Term_Dependency*. Die erste Klasse enthält neben dem Term selbst eine Liste mit Synonymen, welche als Zeichenkette abgelegt sind, und eine Liste mit weiteren abhängigen Termen. Eine Abhängigkeit wird über den abhängigen Term, die maximale Distanz zum abhängigen Wort und die Richtung, in der die Abhängigkeit gilt, definiert. In der folgenden Abbildung sind die Eigenschaften der maximalen Distanz und der Abhängigkeitsrichtung dargestellt.

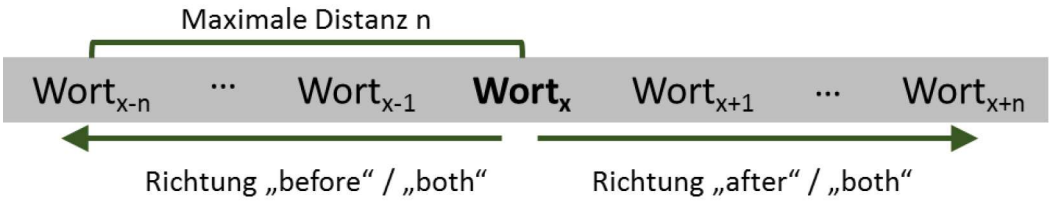

Abbildung 5: Eigenschaften der Term-Abhängigkeiten

Ein Beispiel für dieses Konstrukt ist die Suche nach den zusammengehörenden Termen „in“, „Höhe“ und „von“. Wenn die drei Wörter in Kombination mit einer jeweiligen Distanz von Eins im Satz vorkommen, enthält der Satz einen Betrag.

4.3.2 Implementierung der Extraktion von Personen

Bei der Umsetzung hat sich gezeigt, dass der verwendete POS-Tagger keine Namen erkennt, sondern diese als Nomen klassifiziert. Aus diesem Grund kommt neben dem Tagger ein zusätzliches Modul zur Named-Entity-Recognation (NER) zur Verwendung. Ein in Java implementierter Algorithmus wird von der Universität Stanford unter dem Namen „Stanford Named Entity Recognizer“ bereitgestellt[24]. Hierbei handelt es sich um ein trainierbares Modell, welches anhand von Gelerntem Namen erkennt. Für das Training des Algorithmus werden bereits klassifizierte Datensätze benötigt. In dieser Arbeit wurden jedoch zwei bereits trainierte Modelle verwendet, welche durch Sebastian Padó zur Verfügung gestellt werden[25]. Die Modelle sind nur für akademische Zwecke und vor der produktiven oder kommerziellen Nutzung auszutauschen. Die Erkennungsrate der beiden Modelle weicht leicht voneinander ab, was in einem Test festgestellt wurde. Fünf der acht Namen und Namenskombinationen aus Vor- und Nachname werden jeweils nur von einem der beiden Modelle erkannt. Die folgende Tabelle gibt Übersicht über die verschiedene Erkennungsrate. Als Probetext wurde ein Artikel von der Nachrichtenredaktion „Der Spiegel“[26] verwendet. Der verwendete Text besteht aus 384 Wörtern, davon sind 10 Wörter oder Wortpaare zu taggende Namen.

#	Name	Englisch	Deutsch 1 (HGC)	Deutsch 2 (DEWAG)
1	Angela Merkel	ja	ja	ja
2	Beata Szydlo	ja	ja	ja
3	Merkels	nein	ja	nein
4	Merkel	ja	ja	nein
5	Witold Waszczykowski	ja	ja	ja
6	Waszykowski	ja	nein	ja
7	Frank-Walter Steinmeier	nein	teils	ja

[24] Vgl. Stanford University: The Stanford Natural Language Processing Group
[25] Vgl. Padó, S. / Faruqui, M.: Training and Evaluating a German Named Entity Recognizer with Semantic Generalization
[26] Vgl. Spiegel Online: Polen: Witold Waszczykowski wirft Berlin egoistische Politik vor - SPIEGEL ONLINE

8	Waszczykowski	ja	nein	ja

Tabelle 12: Namenserkennungsrate der verschiedenen Modelle

Gruppe	Englisch	Deutsch 1 (HGC)	Deutsch 2 (DE-WAG)
True Positiv richtig erkannte Namen	6	5	6
False Positiv falsch als Namen klassifiziert	27	0	0
True Negativ richtig erkannte andere Wörter	371	394	394
False Negativ nicht erkannte Namen	2	3	2

Tabelle 13: Konfidenzmatrix zur Namenserkennung

Zum Vergleich neben den beiden deutschen Modellen wurde auch ein Modell, welches mit englischen Texten trainiert und direkt von der Stanford University zur Verfügung gestellt wurde, verwendet. Das englische Modell liefert zusammen mit dem deutschen Modell DEWAG ein gutes Ergebnis für die Gruppe *True Positiv*, welche die korrekte Namensidentifizierung darstellt. In der Gruppe *False Positiv* klassifizierte der englische Algorithmus 27 Terme fälschlicherweise als Namen, ein nicht akzeptables Ergebnis. Die Gruppe *False Negativ* beinhaltet Terme, die fälschlicherweise nicht als Namen erkannt wurden. Dort liegt das deutsche Modell HGC aufgrund der schlechten Identifizierung mit 3 nicht identifizierten Namen hinten.

Eine ausreichende Korrektheit konnte mit keinem Algorithmus erreicht werden. Die *True Positiv* Gruppe des zweiten deutschen Algorithmus liegt gerade einmal bei 62,5% (5 von 8 Namen identifiziert). Das englische Modell kann aufgrund seiner hohen Fehlerquote bei der Gruppe *False Positiv* nicht verwendet werden. Um eine hinreichende Abdeckung zu erlangen, wird die Summe beider deutschen Modelle verwendet. Bei den Testdaten kann so eine 100% Trefferquote erzielt werden. Für die Kombination werden die Freitexte durch beide Modelle verarbeitet und alle Treffer in der Ergebnissumme vereint.

Beim POS-Tagging und der Namensextraktion kommen zwei verschiedene Algorithmen zum Einsatz, die verschiedene Parameter erwarten. Während der Algorithmus zum POS-Tagging ein Array erwartet, welches alle Wörter des Satzes enthält, implementiert der

Algorithmus zur Namensextraktion die Tokenisierung selbst und erwartet entsprechend als Eingabe einen kompletten Satz. Beide Algorithmen liefern ein Array zurück, welches für jedes Wort einen Tag bzw. Kennzeichen, ob es ein Name ist, enthält.

Alg.	Der	Hausrat	von	Max	Mustermann	ist	mit	versichert
POS	ADJ	NN	ADV	NN	NN	V	PRP	V
NER	O	O	O	Name	Name	O	O	O
Ergebnis	ADJ	NN	ADV	Name	Name	V	PRP	V

Tabelle 14: Kombination der Tagging-Arrays

Diese beiden Arrays werden übereinandergelegt, sodass die Ausgaben der Namensextraktion den getaggten Wert des POS-Taggings überschreiben. Jedoch arbeitet der Algorithmus für die NER bei der Tokenisierung anders als der POS-Tagging-Algorithmus, sodass es vorkommen kann, dass die Arrays unterschiedliche Längen aufweisen. Dieser Fall tritt in verschiedenen Szenarien ein und liegt meist an einer verschiedenen Behandlung des Minus-Zeichens. Der NER-Algorithmus behandelt das Minus-Zeichen, wie es für die Vervollständigung von Worten genutzt wird (zum Beispiel bei „Haus- und Gartenpflege“), als eigenständigen Term. Deshalb werden bei dem Vergleich nur Elemente mit einer Länge größer Eins betrachtet.

4.3.3 Suche nach Beträgen

Der letzte Aspekt, nach dem in den Sätzen gesucht wird, ist die Angabe von Versicherungssummen, bis zu denen ein Objekt mitversichert ist. Der Indikator für die Angabe einer solchen Summe ist eine der nachfolgenden Terminologien: „in Höhe von“, die Abkürzung „i. H. v.“ oder „im Wert“. Für das Ermitteln von Beträgen wird der nachfolgend aufgeführte reguläre Ausdruck verwendet.

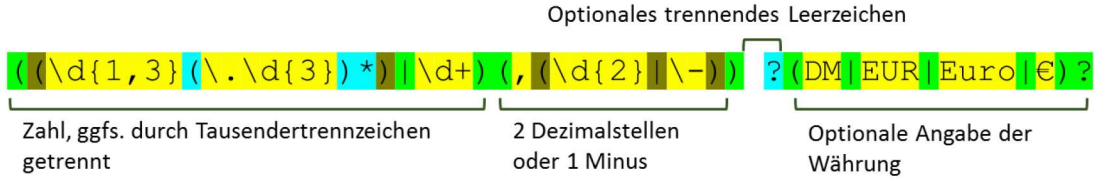

Abbildung 6: Regulärer Ausdruck zur Extraktion von Beträgen

Der reguläre Ausdruck setzt sich aus drei Teilen zusammen. Zunächst wird nach ganzen Zahlen gesucht, die entweder durch das Tausendertrennzeichen, den Punkt, getrennt sind oder ohne ein solches angegeben werden. Bei der Angabe mit Tausendertrennzeichen enthält die erste Zahlenfolge entweder eine, zwei oder drei Ziffern. Die nächsten Tausenderblöcke enthalten dann immer drei Ziffern. Nachfolgend wird der Dezimalbetrag ermittelt, welcher allerdings nicht immer mit angegeben, sondern auch durch die Schreibweise mit Komma und Bindestrich statt Komma mit zwei Nullen dargestellt wird. Abschließend kann nach einem optionalen Leerzeichen die Währung mit extrahiert werden, sofern diese angegeben wurde. Werte, die in der nicht mehr verwendeten Währung Deutsche Mark angegeben sind, werden für die Speicherung mit dem unwiderruflichen Umrechnungskurs 1,95583 in Euro umgerechnet[27].

4.4 Rückführung der Daten

Letztendlich sind die ergänzten Daten in die verarbeitenden Systeme zurückzuspielen und die dort vorliegenden Vertragsdaten zu ergänzen. Wie ein solcher Prozess aussehen kann, wird in diesem Unterkapitel beschrieben.

Zunächst ist die Datenbank um eine Tabelle zu erweitern, in der zu Verträgen inkludierte oder exkludierte Merkmale abzulegen sind. Dazu werden die zur Identifikation des Vertrages benötigten Attribute Vertragsprüfziffer, Vertragsnummer und das Unternehmenskennzeichen abgelegt. Des Weiteren werden die Daten fachlich historisiert abgelegt, was bedeutet, dass zu einem mitversicherten Merkmal der Beginn und das Ende des Gültigkeit gespeichert wird. Ein Historienkonzept wird bereits in vielen existierenden Systemen und Tabellen betrieben und spielt im Versicherungswesen eine wichtige Rolle. Die Dauer des Versicherungsschutzes, bzw. der Gültigkeit erstreckt sich in den meisten Fällen über die komplette Vertragsdauer, jedoch kann sie auch variieren und nur einen bestimmten Zeitraum gültig sein.

Ein weiteres benötigtes Attribut ist ein Kennzeichen darüber, ob das Merkmal bzw. das Objekt im Versicherungsschutz mit inbegriffen oder davon ausgenommen wurde. Beide

[27] Vgl. Euro-Informationen: Umrechnung zwischen DM und Euro - EU-Info.de

Konstellationen sind vorzufinden. Das Feld *Einschluss_KZ* kann entweder eine 1 für eine Inklusion oder eine 0 für eine Exklusion beinhalten.

Das Feld *Objekt* enthält die Bezeichnung des Objektes oder der Dienstleistung, also das Risikomerkmal, welches identifiziert wurde. Sind mehrere Risikomerkmale in einem Satz angegeben, werden mehrere Einträge in dieser Tabelle erzeugt, sodass jedes Risikomerkmal separat aufgelistet wird.

Optional kann zu einem Merkmal eine hinterlegte Deckungssumme angegeben werden, bis zu der ein Versicherungsschutz gewährleistet ist. Dieses Feld kann nur bei einem Einschluss sinnvoll gefüllt werden und auch nur dann, wenn in dem ursprünglichen Freitext eine solche Deckungssumme mit angegeben wurde.

Für die nicht zwingende Angabe von Personen, auf die sich ein Risikomerkmal bezieht, ist das Feld *Ref_Person* vorgesehen. Es wird, wenn angegeben, der Vor- und Zuname gespeichert und zur näheren Spezifizierung die Beziehung, in welcher die Person zu dem Versicherungsnehmer steht. Diese wird in dem Feld *Ref_Person_Relation* gespeichert.

Mit dem Anlegen der Tabelle lassen sich mitversicherte und von der Versicherung ausgeschlossene Objekte festhalten. Damit stehen sie für weitere Abfragen, z.B. das maschinelle Auslesen zur Verfügung.

Spaltenname	Datentyp	Optionen PK = Primary Key	Erklärung
VKPZ	Smallint	PK, Not null	Prüfziffer des Vertrages
VKNR	Decimal(15)	PK, Not Null	Vertragsnummer
VKUKZ	Smallint	PK, Not Null	Unternehmenskennzeichen
Gueltig_Von	Timestamp	PK, Not Null	Gültig Von
Gueltig_Bis	Timestamp	Not Null	Gültig Bis
Einschluss_Kz	Tinyint	Not Null	1 = Einschluss eines Objektes 0 = Ausschluss eines Objektes
Objekt	Varchar(255)	PK, Not Null	Beschreibung des Objektes

Deckungs-summe	Decimal(8,2)	Null	Optional angegebene Deckungs-summe
Ref_Person	Varchar(255)	Null	Optional, Personenname für die das Merkmal gilt
Ref_Per-son_Relation	Varchar(255)	Null	Optional, die Beziehung zu dem Ver-sicherungsnehmer

Tabelle 15: Darstellung der Tabellenstruktur zur Speicherung der Risikomerkmale

Um den Datenbestand jedoch aktuell zu halten, reicht nicht nur die Vorhaltung von historischen Daten in der Tabelle, sondern auch die Daten von aktuellen Verträgen sind zu speichern. Insbesondere bei Neuerfassung von Verträgen oder Vertragsänderungen, bei denen bislang auf die Nutzung der Freitextfelder für das Eintragen von Risikomerkmalen zurückgegriffen wurde, ist die Verwendung der Felder zukünftig zu vermeiden. Die Informationen sind direkt in die neuen Tabellen aufzunehmen, wodurch die Nutzung der Freitextfelder reduziert wird. Eine dauerhafte und regelmäßige Bewirtschaftung der neuen Tabellen aus den Freitextfeldern heraus ist mit diesem Konzept schwer möglich, da die Extraktion und das Zurückführen einen wesentlichen manuellen Eingriff benötigen. Deshalb ist die Überführung mit dem Prototyp nur rückwirkend durchzuführen.

Für aktuelle Daten wird die Benutzung der Freitextfelder durch neue Felder in der Eingabemaske überflüssig gemacht. Dazu wird dem Sachbearbeiter ein zusätzlicher und ergänzender Dialog angeboten, über den Risikomerkmale eingetragen werden. Die Liste der Objekte ist bereits vorgegeben, sie kann jedoch durch den Sachbearbeiter bei Bedarf durch einen Neueintrag erweitert werden. Die Umsetzung des neuen Dialoges ist nicht Teil dieser Arbeit.

4.5 Überprüfen der Ergebnisse

Für die Überprüfung der Arbeitsweise und Präzision des Prototyps wurde eine Reihe von gemischten Datensätzen herangezogen, die einen repräsentativen Bestand abbildet. Es sind Fälle für alle umzusetzenden Anforderungen, sowie die Kombinationen aus diesen, enthalten.

Für jeden Testdatensatz wird vor der Verarbeitung ein erwartetes Soll-Ergebnis definiert, mit dem das tatsächliche Resultat zu vergleichen ist. Für den ersten Test werden schematisch Testdaten generiert, mit denen die Kombinationen von den zu extrahierenden Daten getestet werden. Im zweiten Schritt werden Daten aus dem Datenbestand herangezogen, manuell die erwarteten Ergebnisse hinterlegt und anschließend mit den vom Prototyp ermittelten Ergebnissen abgeglichen.

Die Ergebnisse sind in der nachfolgenden Tabelle festgehalten, die Konfigurationen der generischen Testfälle einschließlich der Soll- und Istwerte sind im Anhang A VII zu finden. Die echten Testfälle sind aus Datenschutzgründen nicht aufgeführt.

Testfall	Art	Ergebnis	Testfall	Art	Ergebnis	Testfall	Art	Ergebnis
#1	generisch	ok	**#17**	echt	fehlt	**#33**	echt	fehlt
#2	generisch	ok	**#18**	echt	fehlt	**#34**	echt	ok
#3	generisch	n. ok	**#19**	echt	ok	**#35**	echt	fehlt
#4	generisch	ok	**#20**	echt	fehlt	**#36**	echt	ok
#5	generisch	ok	**#21**	echt	ok	**#37**	echt	ok
#6	generisch	ok	**#22**	echt	ok	**#38**	echt	fehlt
#7	generisch	ok	**#23**	echt	ok	**#39**	echt	fehlt
#8	generisch	n. ok	**#24**	echt	ok	**#40**	echt	fehlt
#9	generisch	ok	**#25**	echt	ok	**#41**	echt	fehlt
#10	generisch	ok	**#26**	echt	ok	**#42**	echt	fehlt
#11	generisch	ok	**#27**	echt	ok	**#43**	echt	fehlt
#12	generisch	ok	**#28**	echt	fehlt	**#44**	echt	fehlt
#13	generisch	ok	**#29**	echt	n. ok	**#45**	echt	fehlt
#14	generisch	ok	**#30**	echt	fehlt	**#46**	echt	fehlt
#15	echt	fehlt	**#31**	echt	fehlt	**#47**	echt	fehlt
#16	echt	fehlt	**#32**	echt	fehlt			

Tabelle 16: Übersicht über alle Testfälle

Wie in der Tabelle zu sehen ist, arbeitet der Prototyp auf den vorhandenen generischen Testdaten nahezu fehlerfrei und hat alle zu extrahierenden Informationen gefunden.

Falsche Ergebnisse, also das Erkennen von Informationen, die im Text entweder nicht existieren, oder anders stehen, wurden nicht produziert. Die zwei fehlerhaften Fälle sind auf eine falsche Satzerkennung zurückzuführen, welche die gängige Abkürzung „i. H. v." für „in Höhe von" als Satzende interpretiert. Das verwendete Modul zur Satzerkennung teilt einen solchen Satz in zwei einzelne Sätze auf, wodurch es zu einer fehlerhaften Verarbeitung kommt.

Anders sieht es bei der Auswahl der echten Testdaten aus, für die bereits existierende Daten herangezogen und manuell im Vorfeld Risikomerkmale, Beträge und Personen extrahiert wurden. Die Implementierung liefert bei 45 Prozent der Datensätze richtige und vollständige Ergebnisse. Von 33 Datensätzen wurden bei 15 alle relevanten Informationen gefunden, bei einem Satz wurden lediglich Teilinformationen extrahiert. Die restlichen 17 Datensätze wurden um keine Informationen ergänzt. Bei der Suche nach dem Grund ist, wie auch bei den generischen Datensätzen, eine fehlerhafte Satzaufteilung aufgefallen. Das Aufteilen eines Textes, der Punkte nicht nur zur Satztrennung, sondern auch an Stellen wie Abkürzungen (z.B. „geb." für „geboren", oder „i. H. v.") enthält, wird durch den Algorithmus zur Satzdetektion in keiner hinreichenden Präzision erledigt. Punkte, die als Tausendertrennzeichen vorkommen stellen kein Problem dar.

4.6 Erweiterungsmöglichkeiten

Der in dieser Arbeit entwickelte Prototyp besitzt rudimentäre Funktionen und es ist sinnvoll, ihn vor einer produktiven Nutzung anzupassen und zu erweitern. Es wird eine mögliche Erweiterung beschrieben, welche den Prototyp neue Risikomerkmale erlernen lässt. Zudem wird auf die Möglichkeit der Korrektur von Rechtschreibfehlern eingegangen und die Anpassung für die massentaugliche Ausführung auf großen Rechenclustern beleuchtet.

4.6.1 Automatisiertes Erlernen von neuen Risikomerkmalen

Damit die Erkennung von Risikomerkmalen nicht für alle Merkmalsarten und Ausprägungen manuell erstellt werden muss, ist es denkbar, diesen Prozess zu automatisieren. Die Umsetzung lässt sich unterschiedlich gestalten und auf zwei Varianten wird hier eingegangen.

Durch die Verwendung zweier Listen, einer Positiv- und einer Negativliste, kann dem Programm durch deren dynamische Erweiterung beigebracht werden, Risikomerkmale zu erkennen. Dazu werden dem Sachbearbeiter hintereinander in den Texten vorkommende Nomen angezeigt, die weder auf der Positiv- noch auf der Negativliste stehen. Der Sachbearbeiter entscheidet dann, ob sich dieser Term als Merkmal eignet oder nicht. Je nach Entscheidung wird eine der beiden Listen um den Eintrag ergänzt und bei dem nächsten Vorkommen dieses Wortes nicht mehr nachgefragt. Dem Prototyp wird vor der Ausführung mit Hilfe der Positivliste beigebracht, welche Terme als Risikomerkmale zu identifizieren sind. Bei diesem Vorgehen handelt es sich um einen semiautomatischen Prozess, da für alle Ausprägungen die Entscheidung über die Eignung durchzuführen ist.

Eine weitere Möglichkeit ist die Verwendung eines Machine-Learning-Algorithmus, der auf Basis von Freitexte, zu denen die Merkmale bereits ermittelt wurden, trainiert wird. Durch die Vielzahl der Terme und deren Ausprägungen ist der Umfang der Trainingsdaten hinreichend groß zu gestalten, was ebenfalls einen manuellen Eingriff erfordert. Für die Überprüfung eines solchen intransparenten Algorithmus ist neben dem Trainingskorpus ein Testkorpus von Nöten, mit dessen Hilfe die Qualität der erzeugten Ausgaben überprüft wird. Intransparent bedeutet, dass nicht zu erkennen ist, auf Grundlage welcher Eigenschaften oder Strukturen der Algorithmus seine Entscheidung fällt. Bei einem verhältnismäßig kleinen Datenbestand von ungefähr 5.000 Datensätzen lässt sich kein Trainings- bzw. Testkorpus erzeugen, der groß genug ist, ohne dadurch bereits Großteile der eigentlichen Daten zu verwerten.

4.6.2 Umgang mit Rechtschreibfehlern

Da Rechtschreibfehler oder andere Unsauberkeiten zu Falschinterpretationen oder unvollständigen Extraktionen führen können, sind diese zu beseitigen. Ein möglicher Ansatz ist die Korrektur mithilfe der Kombination von Hidden Markov Ketten zur Berechnung der Wahrscheinlichkeit eines Wortes und der Abgleich des Vorhandenen unter Verwendung einer sogenannten *Fuzzylogik.* Durch die Zuhilfenahme der *Fuzzylogik* lassen sich Wörter vergleichen, auch wenn diese keine volle Übereinstimmung der Buchstaben aufweisen.

4.6.3 Parallelisierte Ausführung im Cluster

Damit sich bei dem späteren, weiterentwickelten Einsatz auch bei größeren Datenmengen schnell Ergebnisse erzeugen lassen, kann das Programm nach ein paar Anpassungen auch im parallelen Betrieb auf anderen Plattformen ausgeführt werden. Durch die Implementierung in Java ist eine Plattformunabhängigkeit gegeben, die lediglich die Installation JVM voraussetzt. Hinzu kommt, dass der parallele Betrieb in großen Rechenclustern wie Hadoop oder Spark, deren Implementierung zu großen Teil ebenfalls in Java erfolgt ist, ohne großen Aufwand realisierbar ist. Spark ermöglicht das parallele Verarbeiten von großen Datenmengen auf verschiedene Art und Weise und erfreut sich, besonders im Vergleich zum klassischen Map-Reduce-Ansatz, wie er im Hadoop-Umfeld Anwendung findet, im Big Data Umfeld immer größerer Beliebtheit. Für die Ausführung von Java-Code im Cluster ist die Verwendung des Spark-Frameworks notwendig, welches direkte Schnittstellen für die parallele Ausführung von Code zur Verfügung stellt. Die Implementierung des Frameworks zur einfachen Wortzählung, wie sie auch zur Analyse der Datensätze in dieser Arbeit durchgeführt wurde, ist im nachfolgenden Listing zu sehen.

```
JavaRDD<String> textFile = sc.textFile("hdfs://SOURCE_FILE");
JavaRDD<String> words = textFile.flatMap(
        new FlatMapFunction<String, String>() {
                public Iterable<String> call(String s) {
                        return Arrays.asList(s.split(" "));
                }
        }
);
JavaPairRDD<String, Integer> pairs = words.mapToPair(
        new PairFunction<String, String, Integer>() {
                public Tuple2<String, Integer> call(String s) {
                        return new Tuple2<String, Integer>(s, 1);
                }
        }
);
JavaPairRDD<String, Integer> counts = pairs.reduceByKey(
        new Function2<Integer, Integer, Integer>() {
                public Integer call(Integer a, Integer b) {
                        return a + b;
                }
        }
);
counts.saveAsTextFile("hdfs://TARGET_FILE");
```

Listing 1: WordCount Beispiel in Apache Spark[28]

[28] Apache Software Foundation: Examples | Apache Spark

Im ersten Schritt wird eine Datei eingelesen und durch die Aufteilung der Zeichenkette bei jedem Leerzeichen ein Wort-Array erzeugt. Im zweiten Schritt werden Wertpaare gebildet, bei denen der erste Wert (nicht eindeutiger Schlüssel) das Wort und der zweite Wert der Zahl Eins entspricht. Der letzte Teil der Verarbeitung fasst alle gleichen Wörter zusammen und summiert den zugehörigen Wert auf, wodurch ein eindeutiger Schlüssel mit dem zugehörigen Wert erzeugt wird.

5 Fazit

In diesem letzten Kapitel werden die produzierten Ergebnisse zusammenfassend und rückblickend betrachtet. Neben dem Hervorheben der wichtigsten Erkenntnisse und den Punkten der Implementierung wird kritisch geprüft, ob und wie die Ziele erreicht wurden. Zu guter Letzt erfolgt ein Ausblick auf mögliche Weiterentwicklungen sowohl hinsichtlich des Prototyps als auch der Forschung im Bereich des Text Minings.

5.1 Zusammenfassung

Im Rahmen dieser Arbeit wurden grundlegende Aspekte und Methoden des Text Minings beleuchtet. Es existieren viele weitere Ansätze für verschiedene Problemstellungen, jeweils mit eigenen Stärken und Schwächen. Die für die in dieser Arbeit betrachtete Problemstellung eingesetzten Methoden sind vor allem die Differenz- und Kookkurrenzanalyse. Mit deren Hilfe lassen sich mögliche relevante Risikomerkmale identifizieren und die Satzstrukturen untersuchen. Probalistische Sprachmodelle zum Taggen von Wortarten und das Pattern-Matching für die regelbasierte Extraktion von Informationen aus Texten haben ebenfalls ihre Anwendung gefunden.

Bei der Analyse des Datenbestandes sind zunächst einige Unstimmigkeiten aufgefallen, die sich später begründen und korrigieren ließen. Weitere Hilfsmittel der Analyse und Vorbereitung der Daten waren neben der simplen Datenglättung und dem Entfernen von Stoppwörtern die Stammformreduktion. Durch sie lässt sich die Komplexität der Sätze reduzieren und Analysen aussagekräftiger und einfacher durchführen. Bei der Analyse, aber vor allem bei der Umsetzung hat sich gezeigt, dass scheinbar triviale Problemstellungen, wie die Extraktion von Namen oder die Ermittlung von Geldbeträgen, häufig ungeplante und im ersten Schritt nicht erkennbare Probleme mit sich bringen.

Bereits in den ersten Phasen der Arbeit hat sich gezeigt, dass anhand des regelbasierten Suchens und Sammelns von Informationen aus völlig unstrukturierten und teilweise kontextfreien Daten nur ein kleiner Teil der Informationen mit ausreichender Qualität erreicht werden kann. Diese Erkenntnis wurde durch die nachträgliche Betrachtung und Validierung der Ergebnisse bestätigt.

5.2 Kritische Würdigung

Während der Recherche nach möglichen Ansätzen für die Lösung des Problems sind besonders die Möglichkeiten des Text Minings in den Bereichen Kategorisierung und Sentiments Analyse (Meinungen und Emotionen in beispielsweise Tweets oder Kundenbriefen) aufgefallen. Diese Bereiche sind bereits auf vielfältige Art und Weise, ob mit Machine Learning, K-Means Algorithmen oder Vektorverfahren, umgesetzt und ausgiebig beschrieben. Auch in den Bereichen Fraud Detection (Betrugserkennung) werden große Fortschritte erzielt indem durch Algorithmen nach nicht aufgedeckten Zusammenhängen und Unregelmäßigkeiten gesucht wird. Das Problem der Vielfältigkeit und Heterogenität der Inhalte von den Freitextfeldern wurde aber nach aktuellem Kenntnisstand nur unzureichend gelöst. Hauptsächlich ist eine harmonisierte Ablage der Informationen für die bessere, automatische Auswertbarkeit eine Schwierigkeit. Die eben genannten Verfahren und Algorithmen eignen sich nicht für die Extraktion von Risikomerkmalen aus Feldern in denen eine Fülle an ungeordneten Informationen vorliegt und die vorgestellten Algorithmen setzen syntaktisch einwandfreie und vollständige Sätze voraus. Mit der Wahl eines Entscheidungsbaumes bzw. Pattern-Matching Algorithmus werden relevante Sätze herausgefiltert, nach bestimmten Kriterien untersucht und letztendlich die wesentlichen Informationen extrahiert. Neben dem hohen initialen manuellen Aufwand besteht die Möglichkeit, dass Informationen, welche nicht den Suchkriterien entsprechen, nicht durch den Algorithmus gefunden oder aus anderen Kontexten heraus falsch interpretiert werden.

Für die Vor- und Aufbereitung der Texte, welche auch in dieser Arbeit in gewissen Teilen bearbeitet wurde, existieren bereits viele Erkenntnisse. Insbesondere die hier nicht betrachtete Fuzzylogik kann gewinnbringend zur Korrektur von Rechtschreibfehlern eingesetzt werden. Die automatische Ermittlung von Wortarten anhand von trainierbaren Modellen, wie das POS-Tagging oder Markov-Ketten, ermöglichen eine kontextabhängige Betrachtung von Wörtern und Wortgruppen. So hat die Überprüfung gezeigt, dass der gewählte Ansatz des regelbasierten Sammelns sehr zuverlässige Ergebnisse erzeugt, jedoch nicht alle Konstellationen und Risikomerkmale automatisch erkennt. Insgesamt kommt es bei einer automatischen Datenextraktion teilweise zu Informationsverlusten,

wenn beispielsweise die Risikomerkmale durch Eigenschaften wie die örtliche Lage oder das Fassungsvermögen näher spezifiziert sind. Insgesamt übertrafen die gesetzten Erwartungen die Möglichkeiten in diesem Bereich und das Ziel wurde aufgrund der Komplexität angepasst.

5.3 Implikation für Wissenschaft und Praxis

Der entwickelte Prototyp hat eine Möglichkeit der Datenverarbeitung aufgezeigt und an einem Beispiel vorzeigbar gemacht. Für einen sinnvollen wirtschaftlichen Einsatz sind jedoch weitere Anpassungen am Prototyp selbst und vor allem an der Arbeitsweise und dem Handlungsspielraum des Sachbearbeiters durchzuführen. Die Informationsbreite und Heterogenität der Daten sind zukünftig zu reduzieren, indem dem Sachbearbeiter neue Eingabemöglichkeiten für verschiedene Ausprägungen angeboten werden. Die Migration der bestehenden Daten in eine bessere Struktur kann nach der Weiterentwicklung des Prototyps automatisiert erfolgen.

Aufgrund der Heterogenität von Sprachen ist die Anwendung von Algorithmen, die für eine andere Sprache als die untersuchte entwickelt wurden, nicht immer gewinnbringend. Durch die Andersartigkeit liefern sie ungenaue oder gar falsche Ergebnisse und lassen sich nicht immer auf die untersuchte Sprache anpassen. Die größten Fortschritte und die meisten Algorithmen sind für die englische Sprache vorhanden, während es für die deutsche Sprache entweder keine trainierten Modelle oder schlichtweg keine passenden Algorithmen gibt. Es ist jedoch zu erwarten, dass es aufgrund der immer höheren Anforderungen an das maschinelle Auswerten von Texten auch in anderen Sprachen zunehmend Material geben wird.

Insgesamt ist die Auswertung sowohl von strukturierten als auch von unstrukturierten Daten in der heutigen Zeit immer wichtiger. Bei einer angenommenen Verdopplung aller Daten alle zwei Jahre[29] nehmen Verfahren zum automatischen Auswerten einen immer größeren Stellenwert ein. Im Textmining sind bereits viele ausgereifte Verfahren im Umlauf, für andere Teilgebiete wiederum fehlen diese oder sind noch im Prototypenstadium.

[29] Vgl. WeltN24 GmbH: Zettabytes: Datenvolumen verdoppelt sich alle zwei Jahre - WELT

Anhang

A I. Vollständige Liste mit Stoppwörtern[30]

ab, aber, abgesehen, alle, allein, aller, alles, als, am, an, andere, anderen, anderenfalls, anderer, anderes, anstatt, auch, auf, aus, aussen, außen, ausser, außer, ausserdem, außerdem, außerhalb, ausserhalb, behalten, bei, beide, beiden, beider, beides, beinahe, bevor, bin, bis, bist, bitte, da, daher, danach, dann, darueber, darüber, darueberhinaus, darüberhinaus, darum, das, dass, daß, dem, den, der, des, deshalb, die, diese, diesem, diesen, dieser, dieses, dort, duerfte, duerften, duerftest, duerftet, dürfte, dürften, dürftest, dürftet, durch, durfte, durften, durftest, durftet, ein, eine, einem, einen, einer, eines, einige, einiger, einiges, entgegen, entweder, erscheinen, es, etwas, fast, fertig, fort, fuer, für, gegen, gegenueber, gegenüber, gehalten, geht, gemacht, gemaess, gemäß, genug, getan, getrennt, gewesen, gruendlich, gründlich, habe, haben, habt, haeufig, häufig, hast, hat, hatte, hatten, hattest, hattet, hier, hindurch, hintendran, hinter, hinunter, ich, ihm, ihnen, ihr, ihre, ihrem, ihren, ihrer, ihres, ihrige, ihrigen, ihriges, immer, in, indem, innerhalb, innerlich, irgendetwas, irgendwelche, irgendwann, irgendwo, irgendwohin, ist, jede, jedem, jeden, jeder, jedes, jedoch, jemals, jemand, jemandem, jemanden, jemandes, jene, jung, junge, jungem, jungen, junger, junges, kann, kannst, kaum, koennen, koennt, koennte, koennten, koenntest, koenntet, können, könnt, könnte, könnten, könntest, könntet, konnte, konnten, konntest, konntet, machen, macht, machte, mehr, mehrere, mein, meine, meinem, meinen, meiner, meines, meistens, mich, mir, mit, muessen, müssen, muesst, müßt, muß, muss, musst, mußt, nach, nachdem, naechste, nächste, nebenan, nein, nichts, niemand, niemandem, niemanden, niemandes, nirgendwo, nur, oben, obwohl, oder, oft, ohne, pro, sagte, sagten, sagtest, sagtet, scheinen, sehr, sei, seid, seien, seiest, seiet, sein, seine, seinem, seinen, seiner, seines, seit, selbst, sich, sie, sind, so, sogar, solche, solchem, solchen, solcher, solches, sollte, sollten, solltest, solltet, sondern, statt, stets, tatsächlich, tatsaechlich, tief, tun, tut, ueber, über, ueberall, überall, um, und, uns, unser, unsere, unserem, unseren, un-

[30] Vgl. Ehling, T.: Deutsche Stoppwort-Liste

serer, unseres, unten, unter, unterhalb, usw, viel, viele, vielleicht, von, vor, vorbei, vorher, vorueber, vorüber, waehrend, während, wann, war, waren, warst, wart, was, weder, wegen, weil, weit, weiter, weitere, weiterem, weiteren, weiterer, weiteres, welche, welchem, welchen, welcher, welches, wem, wen, wenige, wenn, wer, werde, werden, werdet, wessen, wie, wieder, wir, wird, wirklich, wirst, wo, wohin, wuerde, wuerden, wuerdest, wuerdet, würde, würden, würdest, würdet, wurde, wurden, wurdest, wurdet, ziemlich, zu, zum, zur, zusammen, zwischen

A II. Beispielsätze aus dem Datenbestand

Beispielsätze
Im Rahmen der eingeschlossenen Haus- und Grundbesitzer-Haftpflichtversicherung, gilt das Risiko in der [Adresse entfernt] mitversichert. Mitversichert sind über die eingeschlossene Gewässerschaden-Haftpflicht drei Heizöltanks mit jeweils 2.000 Liter Fassungsvermögen.
Fremdenzimmer
Vermittlung von Veranstaltungen. Das Veranstalterrisiko ist ausgeschlossen. Auf den eingeschränkten Versicherungsschutz bei Vermögensschäden wird hingewiesen (vgl. Ausschlusstatbestände).
[Bezeichnung entfernt] Besondere Bedingungen und Risikobeschreibungen für die Betriebshaftpflichtversicherung für Betriebe des Bauhaupt- und Baunebengewerbes
Der 10 %- ige Dauernachlass ist in dem Beitrag bereits berücksichtigt.
Der Dauernachlass von 10% ist im ausgewiesenen Beitrag enthalten.
Im ausgewiesenen Beitrag ist der Dauernachlass bereits eingerechnet.
Salons und Shops in: Würzburg, [Adresse entfernt] (Shop), Würzburg, [Adresse entfernt] (Salon), Frankfurt, [Adresse entfernt] (Shop), Lager mit Friseurbedarf in: [Adresse entfernt].
Die Betriebshaftpflichversicherung gemäß Deckungskonzept gilt nur für die Betriebsart Gastwirtschaft. Die Deckung für das "[Name entfernt] " gilt nicht versichert.
Der Dauernachlass ist bereits im Beitrag enthalten.
In den genannten Beiträgen ist ein 10 %iger Dauernachlass für 3 Jahre Vertragsdauer enthalten.
Der Dauernachlass ist in dem oben genannten Beitrag bereits enthalten.
Mitversichert gilt ein vermietetes Zweifamilienhaus ([Adresse entfernt]) sowie eine auf dem Betriebsgrundstück betriebene Photovoltaikanlage.
Der tarifliche Dauerrabatt ist bereits berücksichtigt.
Der 10%-ige Dauernachlass für die Vertragslaufzeit von 3 Jahren ist im Beitrag bereits eingerechnet.
Außer den mitversicherten Risiken gemäß Deckungskonzept sind keine weiteren Umweltrisiken vorhanden.
Ein Laufzeitnachlass in Höhe von 10 % ist in der Beitragsberechnung bereits berücksichtigt.
Vereinbarungsgemäß beginnt der Versicherungsschutz für den Hotelbetrieb erst am 01.01.2012, 12 Uhr.
Im Jahresbeitrag sind 10% Dauernachlass berücksichtigt.
Im ausgewiesnen Jahresbeitrag ist ein Dauernachlass von 10 Prozent enthalten.
Der Jahresbeitrag ist bereits um den 10 %igen Dauernachlass reduziert.

Tabelle A 1: Beispielsätze aus dem Datenbestand

A III. Ausschnitt aus der Kookkurrenzmatrix

	dies	fur	schad	vereinbar	weg	rahm	gilt	nicht	besond	haarentfern	dauerhaft	haftpflicht	faltenunter-	tattoo	vertrag	euro	mitversichert	permanent-	anspruch	gesetz	piercing	versiche-	gelt	risik	umweltris	gemass	auss	deckungskon-	vorhand
fur	201																												
schad	0	0																											
vereinbar	0	0	0	201																									
weg	0	0	0	921	203																								
rahm	0	209	0	0	0	0																							
gilt	0	289	0	0	0	0	317																						
nicht	0	211	991	0	907	0	211																						
besond	0	0	301	267	301	0	0	313																					
haarentfern	0	0	773	0	737	0	0	771	217																				
dauerhaft	0	0	773	0	737	0	0	771	217	777																			
haftpflicht	0	219	0	0	0	235	383	303	0	0	0																		
faltenunter-spritz	0	0	795	0	755	0	0	799	223	773	773	0																	
tattoo	0	0	745	0	741	0	0	749	223	729	729	0	751																
vertrag	0	0	0	0	0	373	277	0	0	0	0	231	0	0															
euro	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0														
mitversi-chert	293	597	1081	279	961	533	831	1297	451	781	779	697	803	753	431	231													
permanent-make-up	0	0	761	0	757	0	0	759	241	715	713	0	729	723	0	0	763												
anspruch	0	0	789	0	795	0	0	905	243	737	737	0	759	745	0	0	979	725											
gesetz	0	249	0	0	0	255	393	305	0	0	0	671	0	0	255	0	731	0	0										
piercing	0	0	823	0	783	0	0	827	255	763	763	0	793	741	0	0	833	751	749	0									
versiche-rungsnehm	0	0	0	0	0	0	261	0	0	0	0	297	0	0	0	0	423	0	0	327	0								
gelt	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	341	0	0	0	0	0							
risik	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1233	0	0	0	0	0	0						
umweltris	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1209	0	0	0	0	0	0	1207					
gemass	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1377	0	0	0	0	0	0	1215	1209				
auss	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1223	0	0	0	0	0	0	1215	1209	1217			
deckungs-konzept	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1229	0	0	0	0	0	0	1215	1209	1227	1217		
vorhand	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1233	0	0	0	0	0	0	1215	1209	1225	1217	1225	
kein	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1267	0	0	0	0	0	0	1217	1209	1221	1219	1217	1217

Tabelle A 2: Ausschnitt der Kookkurrenzmatrix

A IV. Grafische Darstellung der Kookkurrenz

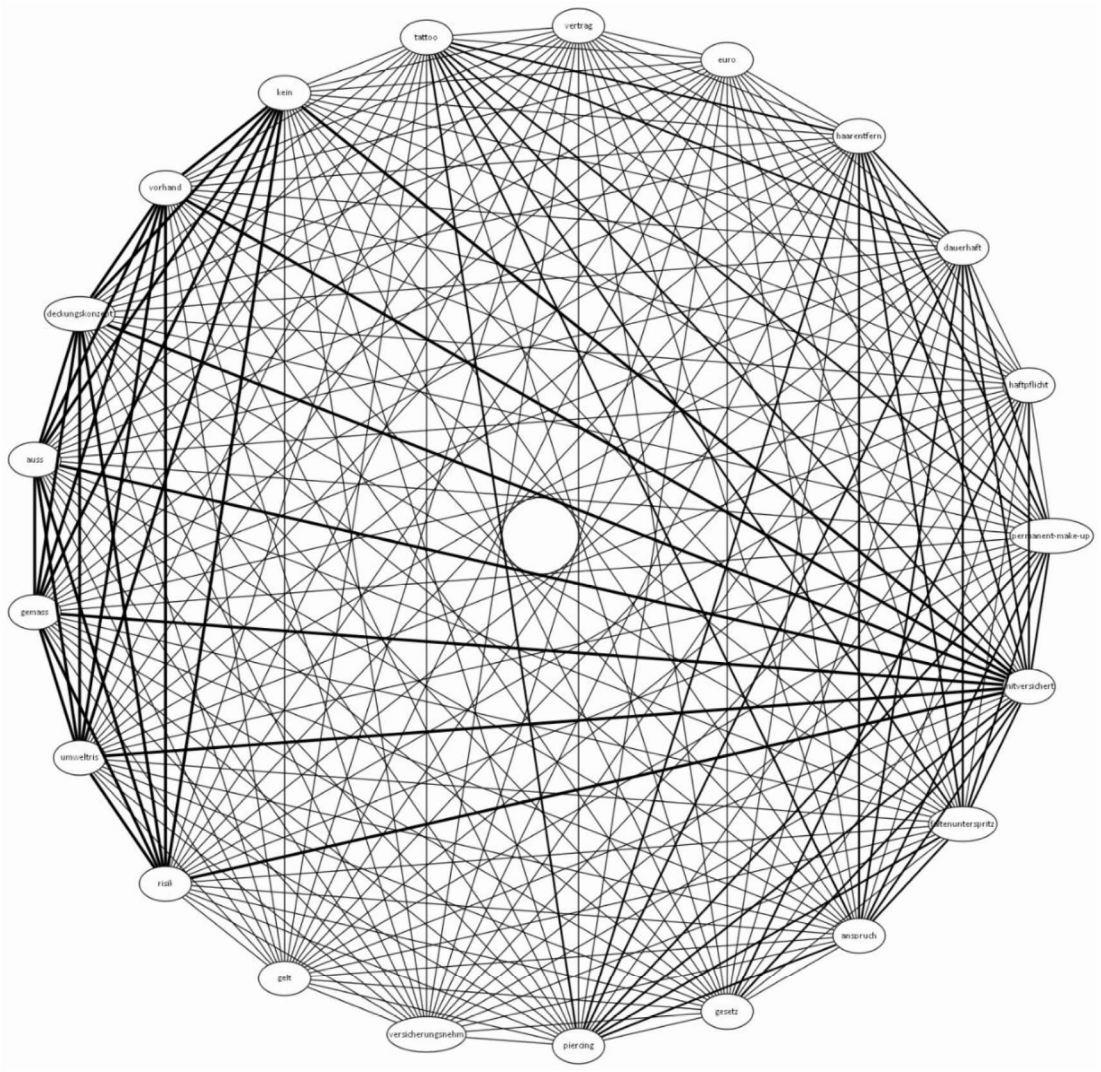

Abbildung A 1: Grafische Darstellung der Kookkurrenzmatrix

A V. Liste der Wortformen nach Penn Treebank

Tag	Description	Tag	Description
CC	Coordinating conjunction	**PRP$**	Possessive pronoun
CD	Cardinal number	**RB**	Adverb
DT	Determiner	**RBR**	Adverb, comparative
EX	Existential *there*	**RBS**	Adverb, superlative
FW	Foreign word	**RP**	Particle
IN	Preposition or subordinating conjunction	**SYM**	Symbol
JJ	Adjective	**TO**	to
JJR	Adjective, comparative	**UH**	Interjection
JJS	Adjective, superlative	**VB**	Verb, base form
LS	List item marker	**VBD**	Verb, past tense
MD	Modal	**VBG**	Verb, gerund or present participle
NN	Noun, singular or mass	**VBN**	Verb, past participle
NNS	Noun, plural	**VBP**	Verb, non-3rd person singular present
NNP	Proper noun, singular	**VBZ**	Verb, 3rd person singular present
NNPS	Proper noun, plural	**WDT**	Wh-determiner
PDT	Predeterminer	**WP**	Wh-pronoun
POS	Possessive ending	**WP$**	Possessive wh-pronoun
PRP	Personal pronoun	**WRB**	Wh-adverb

Tabelle A 3: Wortformen nach Penn Treebank

A VI. SQL-DDL für die neue Tabelle

```
CREATE TABLE `risikomerkmal` (
  `VKPZ` smallint(6) NOT NULL,
  `VKNR` decimal(15,0) NOT NULL,
  `VKUKZ` smallint(6) NOT NULL,
  `Gueltig Von` timestamp NOT NULL,
  `Gueltig Bis` timestamp NOT NULL DEFAULT '9999-12-31 23:59:59',
  `Einschluss Kz` tinyint(4) NOT NULL,
  `Kategorie ID` int(11) NOT NULL,
  `Objekt` varchar(255) NOT NULL,
  `Deckungssumme` decimal(8,2) DEFAULT NULL,
  `Ref_Person` varchar(255) DEFAULT NULL,
  `Ref Peron Relation` varchar(255) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf-8;

ALTER TABLE `risikomerkmal`
  ADD PRIMARY KEY (`VKPZ`,`VKNR`,`VKUKZ`,`Gueltig_Von`);
```

A VII. Konfiguration der generischen Testfälle

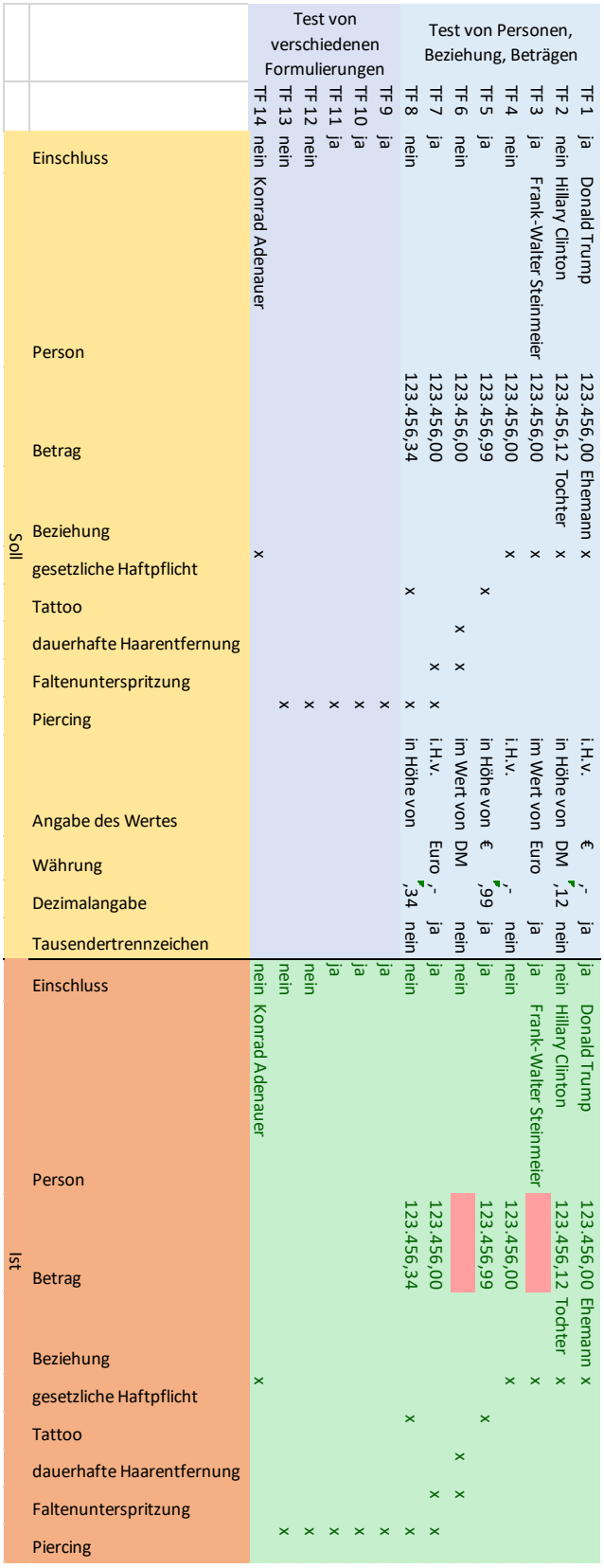

Abbildung A 2: Konfiguration der generischen Testfälle

Literaturverzeichnis

Bibliographisches Institut GmbH (2009) *Duden | Zum Umfang des deutschen Wortschatzes*, Berlin, URL: http://www.duden.de/sprachwissen/sprachratgeber/zum-umfang-des-deutschen-wortschatzes (Abruf: 15.10.2016)

Brants, Thorsten (1999) *Cascaded Markov Models*, Saarbrücken, URL: http://www.coli.uni-saarland.de/~thorsten/publications/Brants-EACL99.pdf (Abruf: 15.10.2016)

Bußmann, Hadumod (1990) *Lexikon der Sprachwissenschaft*, 2., völlig neu bearb. Aufl., Kröner, Stuttgart

Ehling, Tim (2016) *Deutsche Stoppwort-Liste*, Hanau, URL: https://phoenix-vierpunkt-null.de/deutsche-stoppwort-liste-400 (Abruf: 15.10.2016)

Euro-Informationen (2001) *Umrechnung zwischen DM und Euro - EU-Info.de*, URL: http://www.eu-info.de/euro-waehrungsunion/5007/5221/5178/ (Abruf: 15.10.2016)

Foata, Dominique/ Fuchs, Aime (1999) *Wahrscheinlichkeitsrechnung*, 1. Aufl., Springer, Basel

Generali Versicherung AG (o. J.) *Über uns*, München, URL: https://www.generali.de/ueber-generali/ueber-uns (Abruf: 15.10.2016)

Georg Dehio (1914) *Kunsthistorische Aufsätze*, München/Berlin, URL: http://www.deutsches-textarchiv.de/book/show/dehio_aufsaetze_1914 (Abruf: 15.10.2016)

Heyer, Gerhard/ Quasthoff, Uwe/ Witting, Thomas (2006) *Text Mining: Wissensrohstoff Text*, 1. Aufl., W3L-Verlag, Herdecke

Manning, Christopher/ Schütze, Hinrich (1999) *Foundations of Statistical Natural Language Processing*, 1. Aufl., MIT Press, Cambridge, Mass.

Marinschek, M./ Daume, H. (2001) *Text Mining: Verwandlung von unstrukturiertem Text in wertvolle Informationen*

Miles, Patrick (o. J.) *Germanic language stemmers - Snowball*, URL: http://snowballs-tem.org/algorithms/germanic.html (Abruf: 15.10.2016)

Padó, Sebastian/ Faruqui, Manaal (2010) *Training and Evaluating a German Named Entity Recognizer with Semantic Generalization*, Saarbrücken, URL: http://www.nlpado.de/~sebastian/software/ner_german.shtml (Abruf: 15.10.2016)

Pieper, Ursula (1979) *Über die Aussagekraft statistischer Methoden für die linguistische Stilanalyse*, 1. Aufl., Tübingen, Narr

Scheffer, Tobias/ Vanck, Thomas (2010) *Hidden Markov Models - HMMs*, Potsdam, URL: http://www.cs.uni-potsdam.de/ml/teaching/ss10/st/HMMs.pdf (Abruf: 15.10.2016)

Spiegel Online (2016) *Polen: Witold Waszczykowski wirft Berlin egoistische Politik vor - SPIEGEL ONLINE*, Spiegel Online, Hamburg, URL: http://www.spiegel.de/politik/ausland/polen-witold-waszczykowski-wirft-berlin-egoistische-politik-vor-a-1109546.html (Abruf: 15.10.2016)

Stanford University (o. J.) *The Stanford Natural Language Processing Group*, Stanford, USA, URL: http://www-nlp.stanford.edu/software/CRF-NER.html (Abruf: 15.10.2016)

Szymanski, Grzegorz/ Ciota, Zygmunt (2002) *Hidden Markov Models Suitable for Text Generation*, Lodz, Polen, URL: http://www.wseas.us/e-library/conferences/skiathos2002/papers/447-308.pdf (Abruf: 15.10.2016)

The Apache Software Foundation (2010) *Apache Download Mirrors*, Los Angeles, USA, URL: https://opennlp.apache.org/cgi-bin/download.cgi (Abruf: 15.10.2016)

The Apache Software Foundation (o. J.) *Examples | Apache Spark*, Los Angeles, USA, URL: http://spark.apache.org/examples.html (Abruf: 15.10.2016)

Uni Leipzig (o. J.) *Wortschatz - International Portal*, Leipzig, URL: http://corpora2.informatik.uni-leipzig.de/download.html (Abruf: 15.10.2016)

University of Pennsylvania (2003) *Penn Treebank P.O.S Tags*, Philadelphia, USA, URL: https://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html (Abruf 15.10.2016)

WeltN24 GmbH (2013) *Zettabytes: Datenvolumen verdoppelt sich alle zwei Jahre - WELT*, Berlin, URL: https://www.welt.de/wirtschaft/webwelt/article118099520/Datenvolumen-verdoppelt-sich-alle-zwei-Jahre.html (Abruf: 15.10.2016)

Zipf, George Kingsley (1935) *The Psycho-Biology of Language. An Introduction to Dynamic Philology*, 1. Aufl., George Routledge & Sons Ltd., London

Versicherung über Selbstständigkeit

Hiermit versichere ich, dass ich die vorliegende Arbeit ohne fremde Hilfe selbstständig verfasst und nur die angegebenen Hilfsmittel benutzt habe.

Hamburg, den _____ _____