Bachelorarbeit

im Studiengang Computerlinguistik

an der Ludwig- Maximilians- Universität München

Fakultät für Sprach- und Literaturwissenschaften

Clustering von Ergebnissen einer automatischen Verschlagwortung

vorgelegt von
Julius Steidl

Betreuer:	Prof. Klaus Schulz
Prüfer:	Prof. Klaus Schulz
Bearbeitungszeitraum:	19. März - 28. Mai 2018

Selbstständigkeitserklärung

Ich erkläre hiermit, dass ich die vorliegende Arbeit selbständig angefertigt, alle Zitate als solche kenntlich gemacht sowie alle benutzten Quellen und Hilfsmittel angegeben habe.

München, den 28. Mai 2018

.....
Julius Steidl

Abstract

In dieser Arbeit wird ein neuartiger Ansatz zur Clusteranalyse von Dokumenten vorgestellt, bei dem die Ergebnisse einer automatischen Verschlagwortung dieser Dokumente zur Analyse genutzt werden. Für die automatische Verschlagwortung wurde der Dientst „TopicZoom Webtags“ genutzt, der zu den erzeugten Schlagwörtern auch Werte über ihre Gewichtung und Angaben zu ihreren Eingeschaften liefert. Zur Umsetzung und Evalution dieses Ansatzes wurde ein Programm entwickelt, dass diese unterschiedlichen „Feature-Werte“ nutzt und verschiedene Cluster-Algorithmen darauf anwendet. Damit fungiert eine solche Verschlagwortung auch als ein neuer Ansatz zur „Feature-Extraction“, die zur Durchführung eines Clusterings nötig ist. Durch die Einschränkung auf bestimmte Schlagwörter anhand ihrer Feature-Werte, sowie der Nutzung ihrer Gewichtungs-Werte, wurden unterschiedliche Ergebnisse bei der Cluster-Analyse erzeugt und evaluiert um die beste Kombination für Nutzung zum Clustering zu ermitteln. Außerdem wurden verschiedene Clustering-Algorithmen dafür angewandt und ihre Ergebnisse ebenfalls evaluiert. Schließlich wurde dieser Ansatz zur Nutzung der Verschlagwortung zur Feature-Extration mit dem gängigen Verfahren der Gewichtung durch Tf-idf verglichen.

Inhaltsverzeichnis

1 Hintergrund & Motivation

1.1 Einordnung und Hintergrund des Themas

Die Clusteranalyse bzw. das „Clustering“ bildet eine wichtige Disziplin im Bereich des „Data-Minings“, der statistischen Analyse meist großer und unstrukturierter Datenbestände, mit dem Ziel der Erkennung unbekannter Strukturen und Relationen darin. Allgemein beschreibt das Clustering die Entdeckung von Clustern bzw. Themengruppen in einem Datensatz, sowie die Zuordnung der Elemente des Datensatzes zu diesen Clustern. Damit bietet es, in Gegensatz zur, bei vielen Anwendungen der Natürlichen Sprachverarbeitung (NLP) eingesetzten Klassifikation von Daten, also der Zuordnung der Datenelemente zu bereits festgelegten Klassen, die Möglichkeit, diese Strukturen durch die Bildung von Gruppenstrukturen selbst zu entdecken. Im Bereich von „Machine Learning“ entspricht das Clustering also dem unüberwachten Lernen („unsupervised learning“), wohingegen die Klassifikation zu überwachten Lernprozessen („supervised learning“) gezählt wird.

Das Clustering wird für eine Vielzahl von Anwendungen im Kontext von Data-Mining und Machine Learning verwendet. So lässt sich die Erkennung und Zuordnung von Strukturen auf unterschiedlichen Datentypen einsetzen, darunter hoch strukturierte Daten in Form von Datenbanken, Daten mit grundlegenden Strukturen wie Textdaten, die durch Syntax und Grammatik strukturiert sind, aber auch unstrukturierte Daten mit einer hohen Entropie, wie beispielsweise visuelle Daten in Form von Bildern oder Messergebnisse wissenschaftlicher Experimente. So findet das Clustering auch Anwendung in den populären Bereichen von „Computervision“ und „Deep Learning“.

Im Feld der Computerlinguistik und des NLP liegen die zu analysierenden Daten zumeist in Textform vor, als einzelner Text oder mehrere Texte in Form von Dokumenten. Dieser Anwendungsbereich der Clusteranalyse auf einer Sammlung unterschiedlicher Dokumente wird meist als Dokumenten-Clustering bezeichnet. Auch das Thema dieser Arbeit ist, mit dem Ziel, die zu verschiedenen Dokumenten erzeugten Schlagwörter zu clustern, dem Anwendungsbereich des Dokumenten-Clusterings zuzuordnen.

Abgeleitet von der Clusteranalyse im allgemeinen Sinn, besteht die Grundidee des Dokumenten-Clusterings darin, die Inhalte verschiedener Dokumente miteinander zu Vergleichen, Ähnlichkeitsstrukturen der Inhalte mit Hilfe der Clusteranalyse zu erkennen und auf dieser Grundlage die Dokumente mehreren Ähnlichkeitsgruppen bzw. Clustern zuzuordnen, die, inhaltlich gesehen, meist Themengruppen entsprechen.

1.2 Thema der Arbeit

Wie im Thema der Arbeit „Clustering von Ergebnissen einer automatischen Verschlagwortung" bereits als Aufgabenstellung definiert wurde, soll es hier um die Anwendung eines Dokumenten-Clusterings auf den durch den Service „TopicZoom Webtags" automatisch erzeugten Schlagwörtern zu den Dokumenten der Eingabedaten gehen.
Die Besonderheit dieser Anwendung des Dokumenten-Clusterings besteht darin, dass die von TopicZoom erzeugten Schlagwörter über Werte zu ihrer Gewichtung und Angaben zu ihren Eigenschaften verfügen. Mit der Aufgabenstellung wurde vorgesehen, diese Angaben zur Auswahl oder zur Einschränkung auf bestimmte Schlagwörter zu nutzen und ein Clustering anhand der unterschiedlichen Gewichtungs-Werte vorzunehmen. Damit entspricht diese Form der Verschlagwortung auch einem Ansatz zur „Feature-Extraction", also der Extraktion spezifischer Elemente, den sogenannten „Features" aus einem Datensatz zur Verwendung für die Analyse. Dadurch ist auch ein Vergleich dieses Ansatzes mit anderen Verfahren der Feature-Extraction, wie dem „Tf-idf-Maß", möglich. Zusätzlich sollen im Zuge der „Feature-Selection", also der Auswahl bestimmter Features nach ihren Feature-Werten, verschiedene Kombinationen der Beschränkungen und Gewichtung umgesetzt, evaluiert und untersucht werden. Mit den ausgewählten Daten soll schließlich eine Clusteranalyse mithilfe verschiedener Cluster-Algorithmen durchgeführt wurden. Abschließend sollen alle Ergebnisse evaluiert und diskutiert werden.
Die Arbeit findet auch vor dem Hintergrund des von der TopicZoom GmbH entwickelten Web-Service „TopicZoom Webtags" statt. Die TopicZoom GmbH ist ein 2008 gegründetes SpinOff des Centrums für Informations- und Sprachverarbeitung der Ludwig-Maximilians-Universität München. Prof. Dr. Klaus U. Schulz, der Betreuer und Prüfer dieser Arbeit, war dabei als Chef-Architekt der TopicZoom Ontologie maßgeblich an dessen Entwicklung beteiligt. Vor diesem Hintergrund könnte diese Arbeit auch als Beitrag zur Grundlagenforschung zur Nutzung und möglichen Weiterentwicklung dieses Services gesehen werden. [1]

1.3 Motivation

Die Motivation des Dokumenten-Clusterings besteht konkret in der Anwendung zur Untersuchung von Dokumentenbeständen. So wird dadurch eine effiziente Ermittlung der Zusammensetzung eines inhaltlich unbekannten Dokumentenbestandes ermöglicht, durch das Aufzeigen der darin beinhalteten Themengruppen und deren Benennung bzw. Bezeichnung anhand der relevantesten Begriffen zu diesen. Eine Implementierung dafür wäre die automatische Bestands- und Inhaltsanalyse eines unbekannten und uneinheitlichen Archivbestandes. Eine weitere Motivation besteht in der Anwendung des Clusterings zur Dokumentenklassifikation. Dabei sollen die einzelnen Dokumente eines Dokumentenbestandes vordefinierten Gruppen zugeordnet werden. Dies ist beispielsweise bei der Sortierung medizinischer Berichte wichtig. Eine weitere zu nennende Motivation ist die Anwendung zum „Document-Retrieval", bei dem auf das Stellen einer (Such-) Anfrage Dokumente, die dieser Anfrage inhaltlich entsprechen, vom System zurückgeliefert werden. Dazu ist es nötig, den Inhalt der vom System gefundenen Dokumente zu prüfen, um die Genauigkeit der Ergebnisse für den Benutzer zu verbessern. Schließlich ermöglicht das Dokumenten-Clustering durch seine analytisch-quantitative Funktionsweise einen objektiveren Blick auf den Inhalt von Dokumenten.

2 Theorie und Grundlagen

Dieser Abschnitt der Arbeit befasst sich mit den theoretischen Grundlagen des Dokumenten-Clusterings. Dazu werden die in der Umsetzung der Aufgabenstellung durchgeführten Schritte und angewandten Verfahren allgemein und in ihrer Funktionsweise erläutert. Das Ziel hierbei ist es, zunächst ein umfassendes theoretisches Verständnis für den Ablauf und die Verfahren zu vermitteln, um das Vorgehen bei der Konzeption und Umsetzung der Aufgabenstellung im Abschnitt 4. und 5. der Arbeit nachvollziehbar zu machen. Dies ist insbesondere nötig, da die zur Programmierung genutzte Sprache „Python“ zwar einerseits eine sehr intuitive und einfache Umsetzung ermöglicht, andererseits aber nur einen unzureichenden Einblick in die zugrundelegenden Funktionsweisen gewährt. Vor allem gilt dies für die Nutzung vordefinierter Funktionen aus den verwendeten Bibliotheken, wie beispielsweise den Clustering-Funktionen der verwendeten „Scikit-learn“- Bibliothek. Der nächste Unterpunkt 2.1. beschreibt zunächst die Grundlagen des Dokumenten-Clusterings. In den darauffolgenden Unterpunkten 2.2 und 2.3. werden dann die wichtigsten Schritte und Verfahren ausführlicher erklärt.

2.1 Allgemeines Vorgehen beim Dokumenten-Clustering

In diesem Unterpunkt werden der allgemeine Ablauf des Dokumenten-Clusterings beschrieben und die einzelnen Schritte aufgezählt.
Als Eingabedaten für das Dokumenten-Clustering dienen mehrere Textdokumente mit unterschiedlichen Inhalten, die in ihrer Gesamtheit oft als Dokumentenkollektion bezeichnet werden. Eine Voraussetzung für ein sinnvolles Clustering der Dokumente nach ihrem Inhalt ist, dass sich Inhalt und verwendete Begriffe in den Dokumenten thematisch zwei oder mehr Ähnlichkeitsgruppen zuordnen lassen. Je eindeutiger die Übereinstimmung der Dokumente mit einer der vorher festgelegten oder im Zuge des Clustering entdeckten Gruppen ist, desto besser lässt sich eine Clusteranalyse durchführen.
Der Ablauf des Dokumenten-Clusterings lässt sich in 5 grundlegende Schritte aufteilen:

1. Feature-Extraction: Als erster Schritt zur Vorbereitung des Dokumenten-Clusterings wird in der Regel eine Form von „Feature-Extraction“ durchgeführt, bei dem bestimmte Begriffe aus dem Text extrahiert und nach ihrer Relevanz für den Inhalt des Dokumentes gewichtet werden. Ein gängiges Verfahren, dass dafür eingesetzt wird, ist das Preprocessing des Textes mit anschließender Gewichtung der Relevanz anhand der Häufigkeit der Begriffe durch das Tf-idf-Maß. Als neuer Ansatz zur Feature-Extraction wird in dieser Arbeit die Verwendung der Ergebnisse einer automatischen Verschlagwortung vorgestellt. Im Unterpunkt 2.2. wird die Feature-Extraction anhand dieser Verfahren genauer erläutert.

2. Vector Space Model: Im zweiten Schritt sollen die extrahierten Begriffe bzw. Features in eine Datenstruktur transformiert werden, die den gesamten Datensatz repräsentiert und auf Grundlage dieser sich erst eine Cluster-Analyse vornehmen lässt. Dazu wird für jedes Dokument ein Vektor erstellt, der „Dokumenten-Vektor“, der für alle in der Dokumentenkollektion vorkommenden Begriffe bzw. Feautes einen Wert erhält, der angibt, wie Häufig oder mit welcher Gewichtung der Begriff in konkreten Dokument auftritt. Analog dazu wird für jeden Begriff der Dokumentenkollenktion ein „Begriffs-“ oder „Term-Vektor“ erstellt, in dem die entsprechen Häufigkeit oder Gewichtung des Begriffes für jedes Dokument angegeben werden.

Dieses Verfahren wird als „Vektorraum-Retrieval“, engl.: „Vector Space Model“ (VSM) bezeichnet wird in vielen Bereichen zur Informationsgewinnung eingesetzt. Die so erhalten Vektoren lassen sich bei Bedarf auch normieren um dafür Einheitsvektoren zu bilden, die eine Länge gleich 1 ist. Diese Maßnahme kann später beim Clustering zu besseren Ergebnissen führen. Nun bilden die Dokumenten- und Term-Vektoren zusammen eine Matrix die einen direkten Vergleich zwischen den einzelnen Dokumenten oder auch Begriffen zulässt und eine Anwendung von Clustering-Algorithmen erstmalig ermöglicht. [2]

3. Abstandsmessung: Anhand dieser Datenstruktur lassen sich im dritten Schritt die inhaltlichen Ähnlichkeiten zwischen den Dokumenten bestimmen. Die zwischen allen Dokumenten errechneten Werte werden als Ähnlichkeits- bzw. Abstandsmaße angeben. Zu den gängigen Maßen zählen der Kosinus-Abstand, der Euklidischer Abstand sowie der „Jaccard Koeffizient“.

4. Clustering: Im vierten Schritt können nun erstmals verschiedene Arten von Clustering-Algorithmen angewandt, wie der„Affinity Propagation-Algorithmus“, dem „Agglomerativ-Hierarchischen Clustering“ oder dem „K-Means-Algorithmus“, die die Dokumenten einer vorgegebenen oder freien Anzahl von Ähnlichkeitsgruppen bzw. Clustern zuordnen. Die Funktionsweise und Eigenschaften dieser Algorithmen wird im Unterpunkt 2.3 ausführlicher erläutert.

5. Bezeichnung der Cluster: Im fünften Schritt können zu jedem der gebildeten Clustern die inhaltlich relevantesten bzw. die am höchsten gewichteten Begriffe extrahiert werden. Dazu werden die finalen Positionen der Zentroiden, die den Mittelpunkt eines Clusters bezeichnen, genutzt. So lassen sich die, an den finalen Zentroiden-Positionen befindlichen Begriffe ausgeben. Diese lassen sich schließlich in Form eines Rankings für jeden Clustern angeben und ermöglichen die Bezeichnung und das Verstänis über den Inhalt des Clusters bzw. der Themengruppe.

[6] [9]

2.2 Feature-Extraction

Die Einheiten aus den Eingabedaten, die zur Analyse verwendet werden, bezeichnet man als „Features“ bzw. „Statistische Variablen“. Sie können einen oder mehrere sog. „Feature-Werte“ enthalten, die für die Analyse genutzt werden. Beim Dokumenten-Clustering handelt es sich bei diesen Features in der Regel um Begriffe aus den Texten und als Feature-Wert wird die Gewichtung ihrer Relevanz für den Inhalt angegeben. Da der Eingabedatensatz viele dieser Features enthalten kann, ist es sinnvoll, zunächst festzulegen, welche der Begriffe als Features genutzt werden sollen und welche Informationen über sie, als ihre Feature-Werte, festgelegt werden sollen. Dieser Schritt der Extraktion von Features aus einem Datensatz und die Ermittlung bzw. Festlegung ihrer Werte wird als „Feature-Extraction“ bezeichnet. Auch beim Dokumenten-Clustering steht in der Regel zu Beginn eine Form von Feature-Extraction. In diesem Fall handelt es sich im Allgemeinen um eine Extraktion relevanter Begriffe und die Gewichtung ihrer Relevanz für den Inhalt des Dokumentes. Darüber hinaus lassen sich weitere Eigenschaften der Begriffe, wie z.B. die Art ihres Typs oder ihre Funktion, als zusätzliche Feature-Werte festlegen.
Die Feature-Extraction lässt sich mit vielen unterschiedlichen Verfahren durchführen. Der in dieser Arbeit vorgestellte neue Ansatz nutzt eine automatische Verschlagwortung als Feature-Extraction und die Schlagworte mit ihren Werten, die sie als Ergebnisse liefert, als die Features für das Dokumenten-Clustering. Vergleichend dazu wird auch der gängigste Ansatz zur Feature-Extraction für Texte vorgestellt, bei dem zuerst ein Preprocessing des Textes durchgeführt wird und dann die Features mithilfe von Tf-idf, einer Technik zur Gewichtung auf dem Verhältnis ihrer Häufigkeit, gewichtet werden.

Das Verfahren der automatischen Verschlagwortung mit TopicZoom zeichnet sich im Besonderen dadurch aus, dass es, im Gegensatz zu vielen anderen Feature-Extraction-Verfahren, neben der Gewichtung auch noch zusätzliche Informationen zu den Schlagwörtern liefert. Diese Informationen lassen eine noch differenziertere Nutzung durch Beschränkung auf bestimmter Features zu. So liefert TopicZoom beispielsweise Angaben zur Allgemeinheit der Schlagwörter oder ihrer Zugehörigkeit zu einem bestimmten Typ, z.B. Zeit, Ort, Person, etc. Der Vorgang der Auswahl bzw. die Beschränkung auf bestimmte Features, die für die Analyse verwendet werden, nennt sich „Feature-Selection“. Die Auswahl, Beschränkung und Kombination von bestimmten Features resultiert meist in unterschiedlichen Ergebnissen ihrer Analyse.

2.2.1 Ergebnisse der Verschlagwortung mit TopicZoom als Ansatz zur Feature Extraction

In diesem Unterpunkt werden die Ergebnisse, die eine Verschlagwortung durch TopicZoom liefert, erklärt. Der Service TopicZoom Webtags liefert für die Eingabe von Texten oder einzelnen Begriffen als Ergebnis eine Auflistung von Schlagwörtern bzw. Begriffen, die den Inhalt des Textes repräsentieren sollen, zurück. Zur Generierung dieser Schlagworte und insbesondere ihrer Zusatzinformationen nutzt TopicZoom eine sehr umfangreiche Ontologie. Dabei handelt es sich um ein breites Themennetz, in dem Wissen über die Zugehörigkeit von Begriffen zu Oberthemen bzw. Oberkategorien, die Relationen zwischen den Themen untereinander und die Einordnung von Begriffen in Themenhierarchien.
TopicZoom bietet also zunächst eine Verschlagwortung im klassischen Sinne, auch als „übliche“ oder „traditionelle“ Verschlagwortung bezeichnet. Die Verschlagwortung, auch Indexierung genannt, ist, nach allgemeiner Definition, die „Zuordnung von Metainformationen zu Dokumenten, mit dem Ziel der Inhaltserschließung und der gezielten Wiederauffindung“ [H. Nohr, S.24]. Als Metainformationen zu diesen Dokumente beschränkt sich die übliche Verschlagwortung dabei auf Begriffe, die wörtlich im Eingabetext auftreten.
Die folgende Zeile ist aus der Ausgabe einer Verschlagwortung mit TopicZoom-Webtags entnommen. Sie enthält neben dem Namen des Schlagwortes, in diesem Fall „München“, auch die unterschiedlichen Feature-Werte, die dafür ermittelt wurden:

```
<TZTopic txt="München" weight="6" DoG="14" Sig="0.436258318551459"
direct="5" RDFID="21996617" Diversity="2" TSCCS="1DFG-DE-TBC" TZTYPE="Geo" />
```

- In der Form dieser Ergebnisse wurde die Angabe über das wörtliche Vorkommen bzw. die „direkten Treffer“ des Schlagwortes mit dem Feature-Wert `direct=1` oder allgemein einem anderen direkt-Wert von $n > 0$ beschrieben. Zudem gibt der direkte Wert auch immer die Anzahl direkter Vorkommen des Begriffs im Text an. Der `direct=0` gibt an, dass es keine wörtlichen Vorkommen des Schlagwortes gibt und es sich um eine „erweiterte Verschlagwortung“ handelt (Erklärung unten).

- Das Gewicht bzw. die Vorkommenszahl wird mit dem ‘weight’-Wert angeben. Dieser gibt die Gesamtzahl von Vorkommen von Begriffen im Text an, die dem betreffenden Thema zu- oder untergeordnet sind. Im Beispiel gibt diese an, dass Begriffe, die dem Thema „München“ zu- oder untergeordnet sind, an sechs Stellen im Text vorkamen. Die Gewichtung lässt sich außerdem zum Ranking oder auch zur Bewertung der Relevanz eines Wortes für den inhaltlichen Kontext nutzen.

- Die Signifikanz des Schlagwortes wird im „Sig-Wert“ angeben. Dies Wert gibt die Auffälligkeit des Wortes in seinem Kontext an, wobei dafür auch die Gewichtung miteinfließt. Darduch erhalten speziellere Themen oder Begriffe bei gleicher Vorkommenszahl eine höhere Signifikanz. Dieser Wert misst, wie „auffällig“ das Thema in seinem Kontext ist. In einem Kontext, in dem ein häufiges Auftreten eines Themas erwarten ist, wird es so erst auffällig, wenn es eine sehr hohe Trefferzahl (weight) besitzt, der Wert wird also in Relation zur üblichen Häufigkeit im Kontext ermittelt. Üblicherweise wird dieser Wert auch zum Ranking bzw. der Bewertung der speziellen Relevanz genutzt.

- Die Tiefe bzw. Allgmeinheit des Themas wird im „DoG-Wert“ angegeben, dem „degree of generality“. Dieser gibt an, wie speziell das Thema aus Sicht der Hintergrund-Ontologie ist. Je höher der DoG-Wert ist, desto spezieller ist das Thema. Dieser Wert ist außerdem vom Kontext unabhängig und bezieht sich nur auf die Ontologie. Er kann von den allgemeinsten Themen mit einer Tiefe von 2 bis zu extrem spezifischen Themen mit einer Tiefe von über 20 reichen. Im Beispiel „München“ mit einem `DoG=14`, handelt es sich schon um ein relativ spezifisches Thema. Für „Schwabing“ wäre der DoG-Wert so auch noch höher.

- Der Typ des Schlagwort wird mit dem „TZTYPE-Wert“ angeben. Dieser gibt die Zugehörigkeit des Schlagwortes zu einer bestimmten Kategorie an, wie „person“, „Geo“, „org“, „time“ oder „Event“ an, mit dem Subtyp „Veranst“.

Mit dem in der umfangreichen Ontologie enthaltenen Weltwissen ist TopicZoom in der Lage, neben einer klassischen Verschlagwortung auch eine erweiterte Verschlagwortung zu erzeugen. Während bei der klassischen Verschlagwortung Begriffe aus dem Text extrahiert werden, um den Inhalt zu repräsentieren, erlaubt die erweiterte Verschlagwortung stattdessen, Oberbegriffe dafür anzugeben, die möglicherweise gar nicht wörtlich im Text auftauchen, aber durch die umfangreiche Ontologie von TopicZoom dafür ermittelt werden können. So würden beispielsweise zu „München“, „Augsburg“ und „Rosenheim“ der Oberbegriff „Stadt“ angeben werden.

- Die Angabe darüber, dass es sich bei einem Schlagwort um ein spezifisches Oberthema bzw. eine Oberkategorie handelt, kann dem „Diversity-Wert“ entnommen werden. Entspricht das Schlagwort einer solchen spezifischen Oberkategorie, wird ein ganzzahliger Diversity-Wert angeben. Ist das Schlagwort unspezifischer, wird ein ungerader Diversity-Wert angeben. So würde den Schlagwörtern „Roggen“, „Hafer“ und „Gerste“ einen ungerader Diversity-Wert von z.B. `Diversity="2.5"` zugeordnet. Zusätzlich würde die Ontologie von TopicZoom aber auch ein Schlagwort „Getreide“ einführen, auch wenn dieses nicht wörtlich im Text vorkommt, und ihm einen Diversity-Wert von 3.0 zuweisen. Falls das Wort „Getreide“ im Text dennoch wörtlich auftritt, würde der Diversity-Wert für „Getreide“ um die Anzahl der direkten Vorkommen erhöht werden, z.B. auf `Diversity=4` bei einem direkten Vorkommen und drei untergeordneten Wörtern. Durch Filtern der Schlagwörter mit ganzzahligen Diversity-Werten lassen sich alle diese spezifischen Oberkategorien zu einem Text extrahieren. Mit den Diversity-Werten lassen sich so auch hierarchische Strukturen abbilden und entnehmen.

[1]

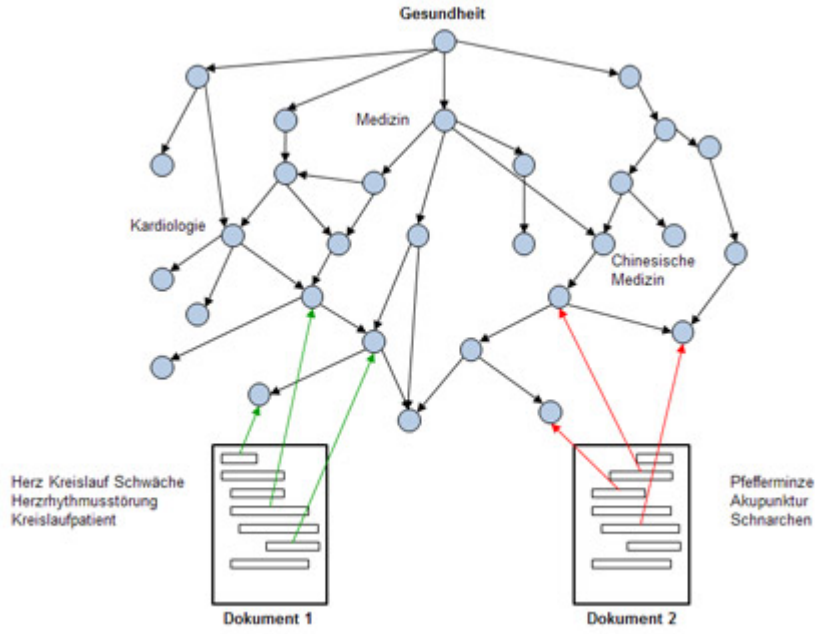

Abbildung 2.1: Intelligente Verknüpfung von Informationen mithilfe einer Ontologie. (Quelle: `http://www.topiczoom.de/wp-content/uploads/2011/09/intelligente-verknuepfung-von-informationen.jpg`)

2.2.2 Preprocessing & Gewichtung mit Tf-idf als klassischer Ansatz zur Feature Extraktion

Ein naiver Ansatz zur Durchführung eines Dokumenten-Clusterings wäre, alle Token in den Dokumente zum Vergleich und Clustering zu benutzen. Ein solcher Ansatz führt in der Praxis meist nicht zu dem gewünschten Ergebnis und bedarf zudem hoher Rechenleistungen, da die Anzahl der zu vergleichenden Token mit zunehmender Größe der Dokumentenkollektion stark ansteigt. Zudem führt die Berücksichtigung aller Wörter zu Fehlannahmen bezüglich der Ähnlichkeit der Dokumente. Dem liegt eine falsche Einschätzung der Relevanz der Begriffe zugrunde. Insbesondere bei den häufig vorkommenden „Stoppwörtern“ zu denen Artikel, Konjunktionen und Präpositionen zählen, die jedoch keinen Inhalt repräsentieren, lässt sich dieser Effekt beobachten. Um dieses Problem zu umgehen und eine gute Repräsentation des Inhaltes zu bekommen, werden zwei grundlegende Schritte durchgeführt: Die Aufbereitung durch „Preprocessing“ und die Beurteilung der Relevanz anhand eines Häufigkeitsmaßes.

Zur Aufbereitung des Inhaltes der Dokumente werden mehrere Verfahren durchgeführt, die unter dem Begriff „Preprocessing“ oder auch der „NLP-Pipeline“ zusammengefasst werden. Dies beinhaltet die Durchführung einer Tokenisierung im ersten Schritt, wodurch der Text in seine einzelnen Token bzw. Wörter aufgeteilt wird. Der zweite Schritt dient dem Herausfiltern der bereits erwähnten Stoppwörter. Im letzten Schritt werden die Tokens durch Stemming, Lemmatisierung, und Normalisierung auf ihren Wortstamm zurückgeführt. Das Ergebnis ist eine Liste aller gefilterter, vorkommenden Wörter in Form ihres Wortstammes, womit sich im nächsten Schritt die Häufigkeit der Wörter in Abhängigkeit zu ihrem Wortstamm ermitteln lässt. So wird verhindert, dass nur ihre konkrete Form gezählt wird. Der zweite grundlegende Schritt ist die Durchführung einer Worthäufigkeitsanalyse. Das Ziel hierbei ist es, für jeden in einem Dokument vorkommenden Wortstamm alle konkreten Formen zu zählen. Anhand dieser Häufigkeiten lässt sich für jeden Wortstamm seine Relevanz für den Inhalt des Dokumenten statistisch schätzen. Vielen Systemen zur Schätzung der Relevanz von Termen für den Inhalt des Textes anhand der Häufigkeit ihres Vorkommens liegt das sog. „Bag-of-words-Modell“ zugrunde. Dieses geht von der vereinfachten Annahme aus, dass nur die Vorkommenshäufigkeit der Terme über deren Relevanz für den Textinhalt entscheidet und die Reihenfolge der Wörter irrelevant ist. Anwendung findet dieses Modell in verschiedenen Verfahren zur Gewichtung von Termen. Ein sehr primitiver Ansatz wäre eine binäre Gewichtung aller Terme über alle Dokumente mit den Werten 0 und 1, wobei ein Term, welcher mindestens einmal in einem Dokument vorkommt, mit 1 gewichtet wird oder mit 0, falls er im Dokument nicht auftritt. Da bei diesem naiven Ansatz die Anzahl der Vorkommen des Termes, welche aber in der Regel Aussagen über die Relevanz des Termes macht, außer Acht gelassen wird, wird ein solches System nur ein ungenaues Ergebnis im weiteren Verlauf liefern. Eine exaktere Gewichtung der Term-Relevanz liefert das sog. „Tf-idf-Maß“ (term frequency – inverse document frequency). Ein Bestandteil davon ist die Term- bzw. Vorkommenshäufigkeit #(t, D) (term frequency bzw. tf), die die lokale Häufigkeit des Terms t im Dokument D angibt. [Formel: TF(t, D) = (Anzahl der Vorkommen des Terms t im Dokument D) / (Anzahl der Terme t im Dokumente D insgesamt)]. Den anderen Bestandteil bildet die inverse Dokumenthäufigkeit idf(t) eines Terms t, die seine Häufigkeit über die gesamte Dokumentenkollektion misst [Formel: IDF(t, D) = log(Gesamtzahl der Dokumente / Anzahl der Dokumente, die t enthalten)]. Das Tf-idf-Maß ist die Kombination einer sublinearen Termhäufigkeit und einer inversen Dokumentenhäufigkeit. Damit setzt das Tf-idf Maß also die Anzahl der lokalen Vorkommen mit der der globalen Vorkommen ins Verhältnis. Der daraus errechnete Wert beschreibt somit die Auffälligkeit des Wortes im Kontext eines Dokumentes in Relation zu seiner allgemeinen Häufigkeit. Der Gewichtung von Begriffen mit Tf-idf liegt der Annahme zugrunde, dass Begriffe, die diese Aufälligkeit in ihrem lokalen Kontext besitzen, relevant für den besonderen Inhalt dieses Dokumentes sind und somit auch seinen Inhalt repräsentieren bzw. beschreiben.
Aufgrund seiner Eigenschaften bildet Tf-idf das gängigste Verfahren zur Gewichtung der Relevanz von Begriffen in Bezug auf ihren Kontext. Trotz seiner Einfachheit liefert es gute Ergebnisse und konnte sich in vielen Anwendungen qualifizieren. [3]

2.3 Funktionsweise der Clustering-Algorithmen

In diesem Unterpunkt werden die drei Clustering-Algorithmen, die für das Dokumenten-Clustering im Zuge dieser Arbeit ausgewählt und angewendet wurden, nacheinander vorgestellt und in ihrer Funktionsweise erklärt. Dabei handelt es sich wie bereits erwähnt um den Affinity Propagation-Algorithmus, das agglomerativ-hierarchische Clustering sowie den K-Means-Algorithmus.

2.3.1 Affinity Propagation-Algorithmus

Der „Affinity Propagation-Algorithmus“ basiert auf der Idee des „Austausch-“ oder „Nachrichten-Prinzips“. Die Cluster durch einen Austausch von Informationen zwischen Elementen-Paaren erzeugt bis diese konvergieren. Ein Datensatz wird anhand einer geringen Anzahl von Exempeln, d.h. als „Ideal-Beispiel“ beschrieben, welche als die repräsentativsten für die anderen Elemente identifiziert wurden. Die zwischen den Paaren ausgetauschten Nachrichten repräsentieren die Eignung eines Elementes als Teil eines Exempels das die anderen repräsentiert. Dieses wird dann hinsichtlich der Werte der anderen Paaren aktualisiert. Diese Aktualisierung wird iterativ durchgeführt bis es konvergiert. An diesem Punkt werden schließlich die finalen Exempel ausgewählt.
Ein Vorteil von Affinity Propagagtion ist, dass hier, im Gegensatz zu den meisten anderen Clustering-Algorithmen keine Festlegung der Anzahl der Cluster vorrausgesetzt wird, sondern diese während der Durchführung selbst ermittelt werden. Affinity Propagation schätzt also die Anzahl der Cluster, womit er sich auch ausschließlich für den Zweck der Ermittlung der Cluster Anzahlen eignet. Der zentrale Nachteil von Affinity Propagation ist seine Komplexität. Der Algorithmus besitzt eine Zeit-Komplexität von $O(N^2T)$, wobei N die Anzahl der Elemente ist und T die Nummer der Iterationen. Darüber hinaus entspricht die Speicher-Komplexität $O(N2)$ bei der Nutzung einer vollständigen Matrix. Am besten lässt sich Affinity Propagation für kleine bis mittelgroße Datensätze nutzen. [7]

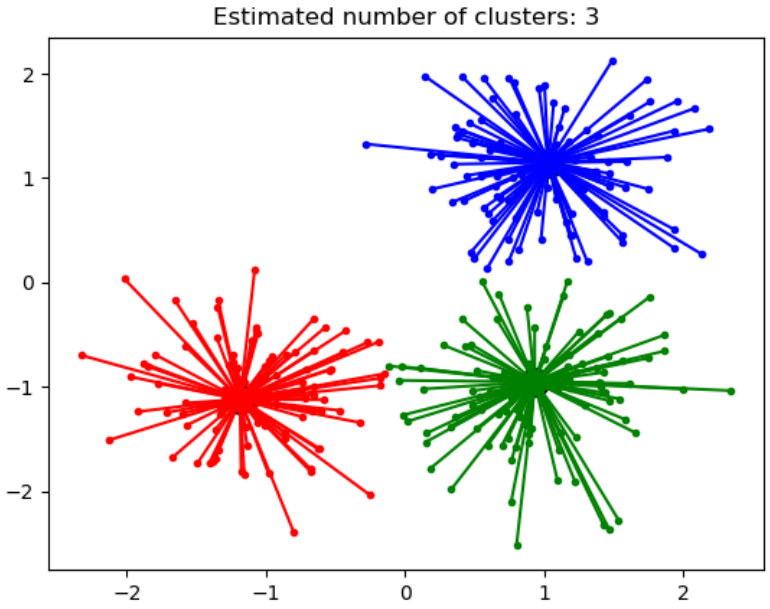

Abbildung 2.2: Affinity Propagation-Clusteranalyse auf einem Datensatz mit drei ermittelten Clustern. (Quelle: `http://scikit-learn.org/stable/auto_examples/cluster/plot_affinity_propagation.html`)

2.3.2 Agglomerativ-Hierarchisches Clustering

Das agglomerativ-hierarchische Clustering lässt sich der Gruppe der hierarchischen Clustering-Algorithmen zuordnen. Agglomerativ beschreibt dabei das Vorgehen, zuerst jedem Element einen eigenen Cluster zuzuordnen und diese dann mit jeder Iteration zu größeren Clustern zusammenzufassen. Dabei spricht man von einem „Bottom-up-Verfahren“. Analog dazu existiert auch ein „divisives-hierarchisches“ Clustering, das nach einem „Top-Down-Verfahren“ arbeitet und so zunächst alle Elemente einen einzigen Supercluster zuordnet. Neben der Unterscheidung dieser allgemeinen Ansätze, wird auch nach „linkage-criteria“, also dem Verknüpfungs-Kriterium unterschieden, dass das Maß für die Strategie zur Verknüpfung der Einzel-Cluster festlegt. Man unterscheidet zwischen drei Strategien dabei:

- Bei der „Ward“-Strategie wird die die Summe der quadratischen Unterschiede zwischen allen Clustern minimiert.
- Die „Maximum“ oder „complete linkage“-Strategie minimiert die maximale Distanz zwischen den beobachteten Cluster-Paaren.
- Bei „Average linkage“ wird die durch durchschnittliche Distanz zwischen allen beobachteten Cluster-Paaren minimiert.

Agglomerativ-hierarchische Clustering lässt sich auch auf große Datensätze skalieren, sofern es zusammen mit einer Verknüpfungs-Matrix verwendet wird. Werden keine Verbindungs-Beschränkung zwischen den Elementen hinzugefügt, sind hohe Rechenleistungen nötig. [7] [4]

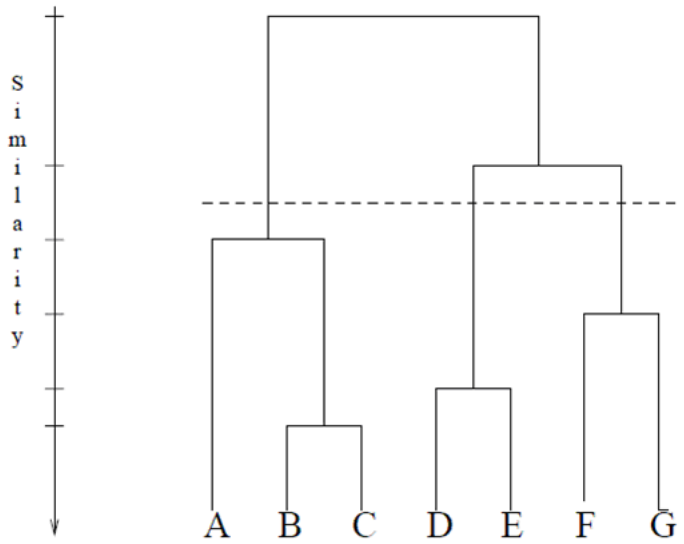

Abbildung 2.3: Agglomerativ-hierarchische-Clusteranalyse als Dendogram dargestellt. (Quelle: `https://commons.wikimedia.org/wiki/File:Agglomerative_clustering_dendogram.png`)

2.3.3 K-Means-Algorithmus

Der K-Means-Algorithmus, ist der der populärste Clustering-Algorithmus. Seine Funktionsweise ist sehr ähnlich zum „Expectation-Maximization-Algorithmus“. Er speichert k Zentroiden zur Definition der Cluster. Ein Element wird als einem Cluster zugehörig gesehen, wenn es zu dessen Zentroiden, einem Punkt in der Mitte eines Clusters, näher ist, als zum Zentroiden eines anderen Clusters. K-Means findet die besten Zentroiden- Positionen durch die abwechselnde Zuordnung von Elementen zu den Zentroiden auf Grundlage derer aktuellen Position (1) und der Wahl neuer Zentroiden-Positonen auf Basis der aktuellen Zuordnung der Elemente zu den Clustern.
Die Vorteile des K-Means Algorithmus liegen in seinem ausgeglichenen Verhältnis von hoher Genauigkeit und geringen Leistungsanforderung, sowie seiner universellen Einsetzbarkeit, begründet. Die initialen Positionen der Zentroiden werden meist zufällig gesetzt, können aber auch vorgegeben werden. Diese Startpositionen sind meist entscheidend für die Qualität des Clusterings sowie Anzahl der benötigten Iterationen. Werden die Startpositionen durch Zufall ungünstig gesetzt, entsteht womöglich in diesem Durchgang ein anderes und schlechteres Clustering. Dies ist auch der größte Nachteil des Clusterings mit K-Means. Um diesen Fehler entgegen zu wirken und die Genauigkeit des Clusterings zu verbessern, ist es in der Regel sinnvoll, mehrere Durchläufe vorzunehmen, jedes mal eine andere zufällige oder geschätzte initiale Zentroidenposition zu wählen und im Anschluss die Ergebnisse miteinander vergleichen und so Fehlerhafte Durchläufe auszuschließen. Mit dieser Maßnahme lassen sich mit dem K-Means Algorithmus sehr gute Ergebnisse erzielen. Eine erweiterte Variante ist der „K-Means++“-Algorithmus, bei dem die Startpositionen nicht zufällig sonder, nach festgelegten Vorschriften gewählt werden, um dem oben genannten Problem entgegenzuwirken. Weitere Varianten sind der halbierende „bisecting K-means“, „K-Median-Algorithmus“, sowie der „K-Medoids-Algorithmus“. [7] [8]

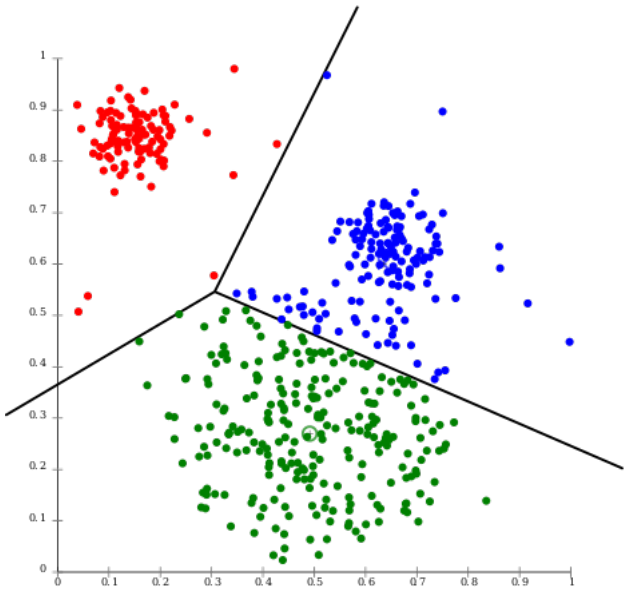

Abbildung 2.4: k-Means-Algorithmus-Clusteranalyse auf einem Datensatz mit Gauss-verteilten Clustern. (Quelle: `https://de.wikipedia.org/wiki/Datei:KMeans-Gaussian-data.svg`)

3 Ziele der Arbeit

In diesem Abschnitt werden die Ziele formuliert, die im Rahmen dieser Arbeit durch die Umsetzung eines Systems und die Untersuchung der Verfahren erreicht werden sollen. Die Grundlage dieser Arbeit bilden Konzeption, Umsetzung und Optimierung eines Systems zum Clustering von Dokumenten. Dieses System soll speziell für die Nutzung von Ergebnisse einer automatischen Verschlagwortung durch TopicZoom konzipiert sein. Das bedeutet im Konkreten, dass das System in der Lage sein soll, zunächst die mit der Verschlagwortung erzeugten Werte zu nutzen und danach die Auswahl an genutzten Features einzuschränken oder unterschiedliche Werte zur Gewichtung zu verwenden.

Das erste Ziel besteht deshalb in der erstmaligen Entwicklung eines solchen Systems und dessen Zurverfügungstellung für verschiedene Anwendungen. Es gibt unterschiedliche Anforderungen an ein solches System. Zuerst soll es viele Nutzungsmöglichkeiten anbieten, d.h. es soll eine vielseitige und differenzierte Steuerung zulassen. Eine weitere Anforderung wird an die Ausgabe des Systems gestellt, es soll die zu ermittelnden Ergebnisse aussagekräftig und übersichtlich dem Benutzer präsentieren. Ein Einblick in die Zwischenschritte und deren Ergebnisse muss geboten werden, um diese nachvollziehbar zu machen und ein Verständnis über stattfindenden Prozesse zu ermöglichen. Die dritte Anforderung besteht in der Flexibilität des Systems. Es soll so entwickelt sein, dass es nicht zu spezifisch auf eine bestimmte Anwendung bzw. auf ein bestimmtes Anwendungsszenario festgelegt bzw. beschränkt ist. Es soll zudem stabil sein, d.h., dass auch Fehlern und Unregelmäßigkeiten in den Eingabedaten oder bei unvorhergesehener Benutzung erkannt werden und sinnvoll darauf reagiert wird, z.B. in Form von Fehlermeldungen oder durch Ignorieren fehlerhafter Daten. Zudem sollen erzeugte Ergebnisse extern abgespeichert werden, um einen Datenverlust zu vermeiden oder anderen Programmen diese Daten zur Verfügung zu stellen. Darüber hinaus, sollte das System auch möglichst effizient programmiert werden. Es sollte keine unnötige Operationen vorgenommen werden, sodass die Anforderungen seine Laufzeit und seinen Speicherverbrauch möglichst gering bleiben. Zuletzt muss das System in seiner Bedienung bzw. „Usability“ einfach, intuitiv und bequem sein. Alle genannten Anforderungen wurden im Zuge der Entwicklung des Systems berücksichtigt.
Die Entwicklung einer solchen Infrastruktur schloss auch die Anforderung bzw. das Ziel mit ein, den gesamten Prozess von der Auswahl der Eingabedaten über deren Extraktion, Verschlagwortung und Clusteranalyse bis hin zur Darstellung und Evaluation der Ergebnisse möglichst effizient, einfach zugänglich und intuitiv zu gestalten. Deshalb wurde das System zum Clustering durch die Entwicklung eines Tools zur einfachen Generierung der Eingabedaten sinnvoll ergänzt, mit dem Ziel, den gesamten Prozess der Durchführung zu vervollständigen und zu automatisieren. Hierfür wurde das Programm „tagfiles_creator.py“ entwickelt, das Extraktion und Verschlagwortung übernimmt.

Da der Zweck des Systems auch in der Schaffung einer Infrastruktur liegt, auf der sich unterschiedliche Kombination dieser Features, unterschiedliche Algorithmen zum Clustering und weitere Techniken und Optimierungen testen und evaluieren lassen, lässt sich die Untersuchung der unterschiedlichen Möglichkeiten als ein zweites Ziel dieser Arbeit formulieren. Dahinter steht die Ermittlung des Ansatzes, der sich am besten für diese Anwendung eignet, also die besten Ergebnisse liefert. Dies soll der Grundlagenforschung und den weiteren Entwicklungen auf diesem Gebiet dienen.

Das dritte Ziel ist die Beurteilung, ob der, im Zuge dieser Arbeit vorgestellte Ansatz zur Nutzung von Ergebnissen einer automatischen Verschlagwortung zum Clustering und damit die Verwendung dieser Verschlagwortung als Verfahren zur Feature-Extraction sich grundsätzlich als sinnvoller Ansatz bewähren kann. Um diese Vermutung beurteilen zu können, wurde ein anderer gängiger Ansatz zur Feature-Extraction mit Preprocessing und Tf-idf durchgeführt und die daraus erhaltenen Ergebnisse vergleichend gegenübergestellt.

Da diese Arbeit auch im Rahmen des Services „TopicZoom“ entstanden ist, soll das entwickelte System sowie die erhaltenen Erkenntnisse ein neue Nutzungsmöglichkeit den Service TopicZoom Webtags darstellen und möglicherweise auch Grundlagen zur möglichen Erweiterung oder Verbesserung dieses Services durch ein solches Verfahren bieten.

4 Konzeption der Umsetzung

Dieser Abschnitt der Arbeit befasst sich mit den konzeptionellen Arbeit zur Umsetzung eines Programms zum Dokumenten-Clustering beschrieben. Dazu werden die Überlegungen und Motivationen hinter bestimmten Entscheidungen zur konkreten Umsetzung, beschrieben.

4.1 Zusammenstellung der Dokumentenkollektion

In diesem Unterpunkt wird erklärt, nach welchen Kriterien die Dokumente für die Dokumentenkollektion ausgewählt wurden und wie sich der so erstellte Eingabedatensatz zusammensetzt.
Als Quelle der Dokumente für die Eingabedatensätze dieser Arbeit wurde die deutsche Seite der freien Enzyklopädie Wikipedia gewählt. Theoretisch wäre jedoch auch die Nutzung von Dokumenten einer anderen Quelle oder aus mehreren unterschiedlichen Quellen möglich gewesen [Beispiel-Anwendungen], da die einzige Voraussetzung für ein Dokumenten-Clustering im Inhalt der einzelnen Dokumente liegt. Der Anlass zur Wahl von Wikipedia für diese Arbeit war das breite Angebot an enzyklopädischen Artikeln, die sich inhaltlich meist auf ein bestimmtes Subjekt beschränken, sowie die Möglichkeit, die Artikel nach gemeinsamen Themen oder Überthemen zu wählen, die später den Clustern entsprechen. So sind die Artikel, meist abhängig vom Thema oder der Kategorie, inhaltlich entweder relativ spezifisch und eindeutig oder eher unspezifisch und variabel. Das Gleiche gilt für die inhaltliche Homogenität oder Heterogenität des Inhaltes verschiedener Artikel einer Kategorie oder eines Themas untereinander. Damit ist gemeint, dass beispielsweise Artikel zu Ländern, wie auch der Artikel zu Deutschland, inhaltlich relativ unspezifisch sind, da sie alle unterschiedlichen Aspekte wie Geographie, Geschichte und Kultur im Inhalt vereinen, im Gegensatz zu z.B. Artikeln über astronomische Objekte, die meist ähnliche Eigenschaften besitzen. Im Hinblick auf die Evaluierung der Auflösung bzw. Genauigkeit der Erfassung des Systems wurde diese Eigenschaft enzyklopädischer Artikel bei ihrer Auswahl für die Evaluierungsdatensätze genutzt. Zur Überprüfung der Funktionstüchtigkeit des Systems und grundlegenden Einschätzung der Genauigkeit wurden zunächst Datensätze mit nur zwei bis drei eindeutig voneinander abgegrenzte Themen bzw. Clustern gewählt. So wurden die Cluster „Elemente des Periodensystems“ mit „Seen in Kanada“ und „Medaillengewinner der Olympischen Spiele 2016“ ausgewählt. Anhand dieser ersten Einschätzung über die vom System zu erwartende Auflösung wurde ein großer, komplexerer Datensatz für die Evaluation zusammengestellt. Dieser umfasst 200 Wikipedia Artikel, die sich zu gleichen Teilen von je 25 Dokumenten zu acht verschiedenen Themen bzw. Clustern eindeutig zuordnen lassen. Die Themen sind dabei so gewählt, dass sie eine thematische Hierarchie abbilden und sich somit Supercluster auf zwei Ebenen bilden lassen [Grafik der Cluster]. Die Themen „Seen in Deutschland“ (A) und „Flüsse in Deutschland“ (B) bilden den Supercluster „Gewässer in Deutschland“ (AB) und sind gleichzeitig ähnlich zu „Seen in Kanada“ (C) sowie zu dem zweiten Supercluster „Städte in Deutschland“ (DE), bestehend aus „Städte in Bayern“ (D) und „Städte in Baden-Württemberg“ (E). Zusammen bilden diese fünf einzelnen Cluster (A,B,C,D,E) zwei Supercluster (AB, DE) und einen weiteren Cluster (C), den noch allgemeineren Supercluster „Geographische Objekte“ (ABCDE). Der dritte Supercluster „Beste Musiker“ (FGH) umfasst die Themen „Beste Gitarristen“ (F), „Beste Schlagzeuger“ (G) und „Beste Sänger“ (H). Diese Zusammensetzung des Datensatzes ermöglicht also ein sehr allgemeines bzw. rudimentäres Clustering mit zwei Superclustern wie „Geographische Objekte“ und „Musiker“ genauso,

wie ein sehr differenziertes bzw. feines Clustering mit einer maximalen Aufteilung des Datensatzes in bis zu acht Subcluster. Bewusst wurden auch Cluster wie „Beste Gitarristen“, „Beste Schlagzeuger“ und „Beste Sänger“ gewählt, da sie als „Musiker“ thematisch sehr nah beieinander angesiedelt sind, bis hin zu Überschneidungen wie z.B. bei „Jimi Hendrix“, der auch „Sängern“ zuzuordnen wäre. Damit bildet dieses Cluster-Tripel auch zusammen mit „Städte in Deutschland“ die uneindeutigste bzw. schwierigste vermutete Clusterzuordnung für ein System. Schließlich wäre auch die Anwendung des hierarchischen Clusterings auf der dreiebenigen Struktur („Flüsse in Deutschland“ $->$ „Gewässer in Deutschland“ $->$ „Geographische Objekte“) möglich. Da die Wikipedia-Artikel eines Themas inhaltlich nicht immer homogen sind, kann es vorkommen, dass einzelne Artikel sich inhaltlich stark von den anderen des gleichen Themas absetzen. Dies führt in der Regel zu einer fehlerhaften Zuordnung durch das System und verzerrt die gemessene Genauigkeit bei der Evaluierung. Um den Einfluss dieser Fehler gering zu halten, wurde die hohe Anzahl von 200 Artikeln und acht Clustern gewählt.
Der beschriebene Datensatz wurde also so konzipiert, dass er eine exakte Messung der Genauigkeit des Systems zulässt und sich anhand der steigerbaren Zahl von Clustern (2-8) mit eindeutig abgrenzbaren, wie auch sehr ähnlichen Clustern, die Auflösung des Systems stufenweise abbilden lässt.

4.2 Auswahl der Clustering-Algorithmen

Dieser Unterpunkt erläutert die Auswahl der verwendeten Clustering-Algorithmen und die Durchführung des Clusterings im Programm „tag_clustering.py“.
Aus der breiten Auswahl gängiger Clustering-Verfahren und ihrer entsprechenden Derivate wurden im Rahmen dieser Arbeit drei unterschiedliche Verfahren zur Anwendung und Evaluation gewählt: 1. der Affinity Propagation-Algorithmus, 2. das agglomerativ-hierachrchische Clustering und 3. der K-Means-Algorithmus, die bereits im Punkt 2.3 der Arbeit in ihrer theoretischen Funktionsweise und ihren Eigenschaften erklärt wurden. Das Motiv zur Wahl dieser drei Algorithmen aus der Menge der gängigen Verfahren ist unter anderem darin zu finden, dass sich jedes dieser Verfahren im Konzept seiner Funktionsweise grundlegend von den anderen unterscheidet. Für jedes der drei Verfahren existieren zudem Erweiterungen bzw. Derivate, z.B. der k-Median-Algorithmus, k-Means++ Algorithmus, die in ihrer Funktionsweise und ihren Ergebnissen meist der Grundvariante ähneln. Deshalb ist anzunehmen, dass die Anwendung und Evaluation dieser Grundvarianten auch repräsentativ für diese Derivate sein könnten und damit eine Einschätzung ihrer möglicher Eignung zulässt. Die Beschränkung auf nur drei Verfahren ergibt sich außerdem daraus, dass der Schwerpunkt dieser Arbeit auf Nutzung und Evaluation verschiedener Feature-Kombinationen sowie den Vergleich zu anderen Feature-Extractions-Verfahren, wie Tf-idf, gelegt wurde. So ergibt sich durch die Anwendung aller drei Algorithmen auf alle Kombinationen der Features bereits eine hohe Anzahl von Durchführungen, deren Ergebnisse manuell evaluiert werden müssen, weswegen es im Rahmen dieser Arbeit nicht möglich ist, alle der Verfahren anzuwenden.
In den vergleichenden Untersuchungen von Michael Steinbach, George Karypis und Vipin Kumar zur Anwendung verschiedener Clustering-Algorithmen zum Dokumenten-Clustering konnten sowohl mit dem K-Means-Algorithmus als auch mit einem agglomerativ-hierarchischem-Clustering gute Ergebnisse erzielen. Weswegen die Auswahl für diese Arbeit auch auf sie viel. [10]

5 Umsetzung & Programmierung

Dieses Kapitel der Arbeit beschreibt das Vorgehen bei der Umsetzung und Programmierung des Dokumenten-Clusterings. In den folgenden drei Unterpunkten, wird der Aufbau und die Funktionsweise des Python-Programms „tag_clustering.py“, das eine Clusteranalyse auf den Eingabedaten durchführt, schrittweise erklärt.

5.1 Erzeugung und Verschlagwortung der Dokumentenkollektion

In diesem Unterpunkt wird dargelegt, wie die Texte für die Dokumentenkollektion erzeugt wurden und wie bei der anschließenden Verschlagwortung der Dokumente vorgegangen wurde.
Zur effizienten Erstellung großer Dokumentensammlungen und ihrer anschließenden Verschlagwortung wurde im Zuge dieser Arbeit ein Python-Programm entwickelt, das diese Aufgabe automatisch erledigt. Das Programm „tagfiles_creator.py“ fasst mit der Option -l als Argument entweder eine Liste mit Namen bzw. Bezeichnungen deutscher Wikipedia-Artikel, durch Zeilenumbrüche voneinander getrennt in einer .txt-Datei zusammen (`$ tagfiles_creator.py -l A titles.txt`) oder mit der Option -n nur den Namen eines einzelnen Artikels direkt als Argument (`$ tagfiles_creator.py -n A München`). Die Artikel sollten einem vorgesehenen gemeinsamen Cluster zuzuordnen sein, für den ein einzelner Buchstabe oder eine Zahl als Kennzeichnung im Argument angegeben wird (`$ tagfiles_creator.py -l A`). Dieser Wert wird in den erzeugten Text- und Schlagwortdateien vermerkt, um später bei der Evaluation die Zuordnung durch das Clustering mit dem vorgesehenen Wert vergleichen zu können. Damit bilden diese vordefinierten Cluster-Zugehörigkeiten zusammen den Goldstandard für das Clustering und ermöglichen später einen direkten Vergleich und somit eine effiziente Evaluierung. Neben der Angabe der Artikel besteht auch die Möglichkeit, über die Option -r zufällige Artikel von Wikipedia auszuwählen und zu verschlagworten, wobei die gewünschte Anzahl der Artikel als Argument angegeben wird (`$ tagfiles_creator.py -r 5`). Für die eingegebenen Artikelbezeichnungen extrahiert das Programm den Textinhalt der Artikel mittels der Wikipedia-Bibliothek für Python, entfernt die Leerzeilen und alle nicht-textuellen Inhalte und speichert den Text als String zusammen mit der entsprechen Artikelbezeichnung in einer eigenen .txt-Datei im Unterordner „text_collection_new“. Die Textdateien dieses Ordners bilden zusammen die Dokumentenkollektion, die in einem späteren Schritt direkt zur Analyse mit Tf-idf verwendet wird. Danach wird der Textinhalt jedes Artikels über die curl-Schnittstelle an den TopicZoom Webtags-Server gesendet, der die entsprechenden Schlagwörter für den Text erzeugt und diese in Form des auf dem XML basierenden SOAP-markup-Textes zurückliefert. Die so zu jedem Dokument erzeugten Schlagwörter werden in dieser Form, zusammen mit dem Titel des Artikels und der vordefinierten Cluster-Zugehörigkeit, in einzelnen .txt-Dateien im Unterordner ‘tag_collection_new’ abgespeichert. Um die erzeugten Textdateien zum Clustering zu verwenden, müssen sie abschließend aus dem ‘tag_collection_new’ Ordner in den „Arbeits“-Ordner ‘tag_collection’ übertragen werden, analog wird mit dem ‘text_collection_new’ Ordner vorgegangen.
Als Eingabedaten bezieht das Programm „tag_clustering.py“ die erzeugten Schlagwörter als Textdateien aus dem „tag_collection“-Ordner. Die Textdateien werden einzeln eingelesen, der Titel sowie die vordefinierte Clusterzuordnung extrahiert und der Inhalt in SOAP-markup Form wird geparsed. Für jedes Schlagwort eines Dokumentes wird ein Wörterbucheintrag mit seiner Bezeichnung erstellt, unter dem die einzelnen Feature-Werte wie „weight“ oder „diversity“ als Liste abgespeichert werden. Die erzeugten Wörterbücher

werden ebenfalls unter ihrem Dokumententitel und zusammen mit ihrer vordefinierten Cluster-Zugehörigkeit in einem Wörterbuch gespeichert, das nun den gesamten Eingabedatensatz repräsentiert. Um diesen zeitintensiven Vorgang nicht bei jedem Programmstart erneut durchführen zu müssen, lässt sich der vollständige Datensatz einmalig in einer von Python lesbaren pickel-Datei abspeichern und effizient wieder laden. Dazu wird im 'data' Unterordner die Datei 'doc_dict.pkl' erzeugt. Durch Auskommentieren der Funktionen `read_tag_files` [Zeile 37] und `save_dict_to_file` [Zeile 38] innerhalb der Main-Methode lässt sich diese Datei für eine geänderte Dokumentenkollektion neu erzeugen.

5.2 Feature-Selection

In diesem Schritt findet eine Feature-Selection, d.h. eine Beschränkung auf bestimmte Schlagwörter bzw. Werte für das Clustering statt. Wird das Programm „tag_clustering.py“ ohne Argument gestartet, werden standardmäßig alle Schlagwörter verwendet und als Wert ihrer Gewichtung wird die Anzahl ihrer Vorkommen bzw. ‚weight‘ genutzt. Über den Programmstart mit Argumenten bzw. Optionen lässt sich die Feature-Selection definieren. So legt das erste Argument fest, welcher Wert für die Gewichtung der Schlagworte verwendet wird. Es besteht die Möglichkeit, die Schlagwörter entweder nach Anzahl ihrer Vorkommen bzw. dem ‚weight‘-Wert, oder nach ihrer Signifikanz, d.h. die Auffälligkeit des Wortes in seinem Kontext, über das Argument ‚-s‘ oder ‚--sig‘ zu gewichten. Über das zweite Argument lassen sich die für das Clustering weiter verwendeten Schlagwörter auf die wörtlich im Text vorkommenden Schlagwörter, also auf direkte, beschränken, mit dem Argument ‚-d‘ oder. ‚--direct‘. Soll keine Einschränkung vorgenommen werden muss das Argument ‚_‘ angegeben werden. Über das dritte Argument lassen sich die verwendeten Schlagwörter weiter auf Oberthemen bzw. Oberbegriffe einschränken. Mit ‚-d‘, ‚--div‘ oder ‚--diversity‘ als drittes Argument werden ausschließlich alle Schlagwörter, die spezifischen Oberthemen entsprechen, d.h. einen gerade ‚diversity‘-Wert besitzen, verwendet [Funktion?]. Analog zum zweiten Argument wird bei der Angabe von ‚_‘ keine Einschränkung vorgenommen. Schließlich lässt sich über das vierte und fünfte Argument noch eine weitere Einschränkung auf Schlagwörter eines bestimmten Types bzw. vornehmen. Diese Funktion wird über die Angabe von ‚-t‘, ‚--type‘ oder ‚--tztype‘ aktiviert und erwartet die Angabe des Types, auf den eine Einschränkung vorgenommen werden soll, als fünftes Argument. Beispielsweise würde ein Clustering mit der Signifikanz als Gewichtung aller wörtlich vorkommenden Schlagwörter mit Beschränkung auf Oberthemen und auf den ‚Geo‘-Typ mit `tag_clustering.py -s -d -d -d Geo` gestartet werden. Zusätzlich werden in diesem Schritt auch noch einige Wikipedia-spezifische Schlagwörter wie 'Weblinks', 'Literatur', 'ISBN', 'Bilder', usw. herausgefiltert. Da diese Begriffe in vielen Artikeln und meist sehr häufig Auftreten, d.h. einen hohe Gewichtung erfahren, jedoch keinen thematisch relevanten Inhalt darstellen, wird durch ihr Herausfiltern das Fehl-Clustering durch die fälschliche Assoziierung von Dokumenten aufgrund dieser Begriffe zueinander reduziert und die allgemeine Genauigkeit erhöht. Abschließend werden in diesem Schritt die gefilterten Eingabedaten wieder in einem neuen Wörterbuch abgespeichert.

5.3 Vorbereitung und Durchführung des Clusterings

Nun werden die auf die gewünschten Features beschränkten Schlagwörter in eine Dokumente-Term-Matrix transformiert. Dazu werden die Schlagwörter für jedes Dokument aus den Wörterbüchern zunächst in Listen konvertiert, die den Dokumentenvektoren entsprechen. Die durch die Scikit-learn-Bibliothek zur Verfügung gestellte Funktion `DictVectorizer` nimmt diese Listen als Argument und transformiert sie zur Dokumenten-Term-Matrix. Um beim Clustering bessere Ergebnisse zu erhalten, lassen sich die Vektoren der Matrix mit der `normalize` Funktion der Scikit-Learn Bibliothek normieren. Aus dieser Matrix können nun die Kosinus-Ähnlichkeiten bzw. Kosinus-Abstände zwischen jedem der Doku-

mente zueinander berechnet werden. Dazu benutzt man die Funktion `cosine_similarity` aus der Scikit-learn-Bibliothek, die die Ähnlichkeit errechnet. Um den Kosinus-Abstand zu erhalten, werden von 1 die Werte der Kosinus-Ähnlichkeiten subtrahiert. [7]
Die im letzten Schritt erzeugte Dokumente-Term-Matrix sowie die errechneten Kosinus-Abstände bilden die Voraussetzung zur Anwendung eines Clustering-Verfahrens. Darüber hinaus ist die Angabe der Anzahl der zu bildenden Cluster für den K-Means-Algorithmus sowie für das agglomerative Clustering nötig. Dieses wird im ‚tag_clustering.py' Programm aus der Anzahl der vordefinierten Cluster-Zugehörigkeiten der Eingabedaten ermittelt. Alternativ besteht natürlich auch die Möglichkeit, die Anzahl der Cluster selbst festzulegen, um z.B eine allgemeinere bzw. gröbere Clusteranalyse durchzuführen. Da als erstes Verfahren das Clustering mit Affinity Propagation durchgeführt wird, das im Gegensatz zu den anderen Verfahren die Anzahl der zu bildenden Cluster anhand des Datensatzes festlegt, besteht außerdem die dritte Möglichkeit, die dadurch ermittelte Cluster-Anzahl für die anderen beiden Verfahren zu nutzen.

6 Ergebnisse & Evaluation

In diesem Abschnitt der Arbeit werden die Ergebnisse der Clusteranalyse evaluiert und diskutiert.

6.1 Unterscheidung der Ergebnisse & Informationen

Das im Rahmen dieser Arbeit entwickelte Programm zur Cluster-Analyse liefert unterschiedliche Informationen und Ergebnisse zurück. Es wurde so umgesetzt, dass sich der gesamte Prozess der Vorbereitung und Analyse in einzelne Schritte bzw. Funktionen gliedern lässt. Jede dieser Funktionen führt einen bestimmten Schritt durch, wobei sie einen oder mehrere Eingabedaten oder Parameter enthält, mit diesen eine Operation durchführt und die transformierten Daten oder Ergebnisse weiter an die nächste Funktion übergeben werden. Am Ende dieses Prozesses steht dann ein zentrales Ergebnis, das der Nutzer des Systems erwartet. Bei den drei zentralen Ergebnissen, die das zum Dokumenten-Clustering entwickelte Programm liefert, handelt es sich um:

1. Die Angabe für jedes Dokument, welchem Cluster es zugeordnet wurde, also seine Clusterzugehörigkeit, die der im Goldstandard vorgesehenen Zuordnung zur Evaluation gegenübergestellt wurde. Für jedes der Clustering-Verfahren wurden die entsprechenden Ergebnisse der Zuordnung angegeben und in einer Textdatei gespeichert.

2. Die Angabe der für jeden Cluster ermittelten n relevantesten Begriffe, aufgelistet in Form eines Rankings, die über seine Zentroidenposition ermittelt wurden. Diese Ergebnisse wurden für die Cluster des K-Means-Algorithmus erzeugt und in einer Textdatei abgespeichert.

3. Die Anzahl der Cluster, die für den entsprechenden Eingabedatensatz durch das System selbst ermittelt wurden. Dies ist der Wert den der Affinity Propagation-Algorithmus für ein Clustering der Eingabedaten vorsehen würde.

Da ein Ziel dieser Arbeit die Untersuchung der Wirksamkeit unterschiedlicher Feature-Kombinationen sowie der Vergleich mit dem Tf-idf Verfahren zur Feature-Extraction ist, werden diese zentrale Ergebnisse für alle gewählten Feature-Kombination und Verfahren erzeugt. Bei jedem Durchlauf des Programms wurden so die Ergebnisse des Clusterings mit allen der drei angewandten Clustering-Algorithmen erzeugt. Zusätzlich wurde für die Feature Kombination mit den besten Ergebnissen, sowie wie für das Tf-idf Verfahren, jeweils ein Durchlauf ohne Vektornormierung zu Vergleichszwecken durchgeführt. Mit insgesamt elf Durchläufen und dem Clustering mit drei verschiedenen Algorithmen für jeden Durchlauf wurden also insgesamt 33 zentrale Ergebnisse für Clusterzuordnungen generiert und evaluiert.
Die einzelnen Ergebnisse von Punkt 1. (siehe oben) wurden, zusammen mit den Informationen über die verwendete Feature-Kombination, sowohl direkt ausgegeben, als auch jeweils in einer eigenen Textdatei gespeichert, um eine Evaluation übersichtlich zu gestalten. Zusätzlich wurden die Ergebnisse von Punkt 2. (siehe oben) über die relevantesten Begriffe ebenfalls ausgegeben und in einer Textdatei gespeichert. Dieser Ordner repräsentiert damit also die Ergebnisse eines Durchlaufes und dient zur Evaluation der Feature-Kombination. Die Einzelergebnisse von Punkt 1. werden in Form sortierter Liste mit verschiedenen Spalten dargestellt. Die erste Spalte enthält einen Index, der der Übersicht dienen kann, jedoch

bisher keine besondere Funktion erfüllt. Die zweite Spalte enthält den Titel des Dokumentes. In der dritten Spalte wurde die durch den Clustering-Algorithmus zugeordnet Clusterzugehörigkeit in Form einer Zahl von 0 bis n vermerkt. Die letzte Spalte enthält die im Goldstandard vorgesehene Clusterzugehörigkeit des Dokumentes. Die Liste wird aufsteigend nach der Goldstandard-Zuordnung sortiert, bei gleichen Zugehörigkeiten aufsteigend nah der Zuordnung durch das Clustering. Diese Form der Darstellung der Ergebnisse wurde mit dem Ziel konzipiert, eine möglichst übersichtliche Darstellung der erzeugten Clusterzuordnungen zu bieten und hat sich auch für den Abgleich der beiden Werte bei der manuellen Evaluation als am effizientesten bewährt. Dieser Vorgang wird im nächsten Unterpunkt 6.2 genauer erläutert.
Die folgenden Zeilen sind ein Ausschnitt aus den Einzelergebnissen für jeden Cluster-Algorithmus:

```
Index Titel  Cluster Gold
2 2 Bodensee 2 A
2 42 Königssee 2 A
2 48 Ammersee 2 A
...
```

Diese zentralen Ergebnisse erfüllen meist den Informationsbedarf, der durch das Programm geliefert werden soll und bieten sich als übersichtliche und verständliche Zusammenfassung der in den einzelnen Analyseprozessen erzeugten Resultate an. So liefern sie auch in dieser Arbeit die von einem Dokumenten-Clustering zu erwartenden Antworten.
Diese Ergebnisse bilden jedoch nur einen Bruchteil der im Laufe des Programms erzeugten Informationen ab. So liefert jeder einzelne Schritt bzw. jede Funktion auch ein eigenes Ergebnis. Diese Zwischenergebnisse bieten meist informative Einblicke in die einzelnen Schritte und ermöglichen so einen umfassenden Blick auf die durchgeführte Analyse, zudem können sie auch gespeichert oder für eine andere Anwendung genutzt werden. Nicht alle Zwischenergebnisse liefern einen brauchbaren Informationswert für den Benutzer des Systems, deshalb wurden nur die informativen zur Ausgabe ausgewählt und aufbereitet. Der Zweck dieser Zusatzinformationen und Statistiken ist es, dem Benutzer einen besseren Einblick in die Zusammensetzung der Dokumentenkollektion zu gewähren und die häufigsten und relevantesten Begriffe auszugeben, um ein Verständnis für den Inhalt der Dokumente zu erlangen. Diese Informationen erfüllen also einen rein informativen Zweck und wurden deshalb auch nicht evaluiert.
Bei den Zusatzinformation, die das entwickelte Programm liefert, handelt es sich um:

1. Die Angabe der Anzahl und Titel der Dokumente, die sich nicht zum Clustering nutzen lassen, da keines ihrer Schlagwörter in einem anderen Dokument vorkommt. Dies ist meist der Fall, wenn ein Dokument leer oder fehlerhaft ist. Zudem wird der Anteil solcher Dokumente von der Anzahl aller Dokumente angegeben.

2. Für jedes Dokument wird die Anzahl der dafür erzeugten Schlagwörter angegeben und der Anteil derer, die auch in anderen Dokumenten vorkommen, also gemeinsame Schlagwörter sind, sowie der Anteil der eigenen Schlagwörter, die nicht für das Dokumenten-Clustering genutzt werden können.

3. Zu jedem Dokument werden die n relevantesten Schlagwörter in einem Ranking angegeben, also die mit den höchsten Gewichtungs- oder Signifikanzwerten.

4. Die Gesamtzahl der Schlagwörter aller Dokumente und der Anteil derer, die in mehreren Dokumenten vorkommen, sowie der Anteil der Schlagwörter insgesamt, die nur in einem Dokument auftreten.

5. Aus allen Dokumenten werden die n häufigsten gemeinsamen Schlagwörter, also die in den meisten Dokumenten vorkommen, in einem Ranking angegeben, zusammen mit der Anzahl der Dokumente, in denen sie gemeinsam auftreten.

6. Aus allen Dokumenten werden die n relevantesten gemeinsamen Schlagwörter angeben, also die, deren Summe ihrer Werte am größten ist, in einem Ranking angegeben.

Darüber hinaus ist das Programm auch in der Lage, numerische Werte zu den Zwischenergebnissen auszugeben. Diese sind weniger informativ, lassen sich jedoch abspeichern und für andere Anwendungen weiterverwenden. Dazu gehört:

1. Die Form bzw. Größe der Dokumenten-Term-Matrix, also die Anzahl der Dokumente als Zeilen und Schlagwörter als Spalten sowie deren Werte.

2. Die zwischen allen Dokumente der Dokumentenkollektion errechneten Abstands- oder Ähnlichkeitsmaße in Form der Kosinus-Abstände.

3. Die Position der Zentroiden der gebildeten Cluster.

6.2 Überlegungen und Vorgehen bei der Evaluation

In diesem Unterpunkt werden die Überlegungen und das Vorgehen bei der Evaluation der Ergebnisse beschrieben. Die Evaluation in diesem Punkt beschränkt sich auf die Ergebnisse die zentralen Ergebnisse des Systems.
Für die Evaluation eines Systems zum Dokumenten-Clustering war zunächst eine Analyse notwendig, welche Anforderungen an ein solches System überhaupt gestellt werden und wie sich messen lässt, inwieweit diese Anforderungen erfüllt wurden.
Die erste zentrale Anforderung an ein solches System ist die korrekte Zuordnung von Dokumenten zu Clustern bzw. Themengruppen. Diese repräsentiert die Wirksamkeit eines solchen Systems in seiner Anwendung zur korrekten Zuordnung von Dokumenten zu vorgesehenen Themengruppen.
Diese Genauigkeit der Zuordnung von Dokumenten zu Clustern durch das System lässt sich relativ einfach evaluieren, indem die Zuordnung durch das System mit der im Goldstandard vorgesehen Zuordnung verglichen wird. Die Bewertung erscheint trivial: Stimmt die Zuordnung mit der des Goldstandards überein, war die Clusterzuordnung für dieses Dokument korrekt, ist die Zuordnung abweichend, handelt es sich um eine Fehlzuordnung bzw. einen Clusteringfehler. Eine Schwierigkeit bei der Durchführung dieser Evaluation war jedoch, dass die Clustering-Algorithmen die Bezeichnungen der Cluster beliebig festlegen, z.B. durch einfaches Durchnummerieren aller Cluster. So werden zwar bei jeder Durchführung des Clusterings die gleichen Strukturen in den Eingabedaten gebildet, da die Bezeichnung der Cluster jedoch beliebig ist, variiert sie bei jeder Durchführung. Aufgrund der Beliebigkeit der Bezeichnung ist kein direkter Abgleich mit den Werten des Goldstandards möglich. Somit ist auch eine Automatisierung des Evaluationsprozesses nicht einfach umsetzbar. Es gibt jedoch zwei Möglichkeiten, dieses Problem zu lösen, bei beiden wird versucht, die durch das Clustering vergebene Bezeichnung der entsprechenden Bezeichnung im Goldstandard zuzuordnen. Beispielsweise würde das System die „Seen in Deutschland“ zum Cluster 0 zuordnen, im Goldstandard wurde die Bezeichnung für „Seen in Deutschland“ jedoch mit dem Wert 2 oder auch dem Wert A bezeichnet. Die Lösung liegt also darin, zu Erkennen, dass 0 und 2 bzw. A den gleichen Cluster, nämlich „Seen in Deutschland“, beschreiben. Ein intuitiver Ansatz wäre es, die Zugehörigkeit anhand der Übereinstimmung der für den Cluster wichtigsten Begriffe zu ermitteln. Das Problem hierbei ist jedoch, dass solche für den Cluster intuitiv angenommene Begriffe wie z.B. „Seen“ und „Deutschland“, die im Goldstandard für „Seen in Deutschland“ als die den Cluster bezeichnenden Begriffe vermerkt werden würden, nicht unbedingt den tatsächlich wichtigsten Begriffen des Clusters entsprechen, die man vielleicht nicht intuitiv dafür annehmen würde. Die Dokumente zu „Seen in Deutschland“ könnte das System stattdessen möglicherweise einem Cluster „Natur“ zuordnen.
Der zweite, einfacher umzusetzende Lösungsansatz, basiert auf der Annahme, dass es sich

bei der Clusterzuweisung mit den meisten Dokumenten, die einer im Goldstandard festgelegten Clusterzuordnung zugeordnet werden, um die richtige Clusterzuweisung handelt. Wenn beispielsweise für einen im Goldstandard festgelegten Cluster A mit 20 Dokumenten durch die Cluster-Analyse elf der dem Cluster A zugehörigen Dokumente dem Cluster mit der Bezeichnung 0 zugeordnet werden und die restlichen neun einem oder mehreren anderen Clustern, gilt die Annahme, dass es sich bei der erkannten und mit 0 bezeichneten Struktur tatsächlich um die Struktur des Clusters A handelt. Eine Überprüfung der Gültigkeit dieser Annahme würde dann einen Vergleich der für den Cluster relevantesten Begriffe des Clusters, wie im ersten Lösungsansatz beschrieben, voraussetzen. So konnten einzelne Überprüfungen dieser Annahme deren Gültigkeit bisher nicht widerlegen, jedoch ist eine universelle Gültigkeit dieser Annahme nicht anzunehmen.
Zur Evaluation des Systems wurden die in den Durchläufen von jedem Algorithmus erzeugten Einzelergebnisse in Form von Listen verwendet (siehe Unterpunkt 6.1). So wurden für jeden Eintrag dieser Listen der vom Algorithmus zugeordnete Wert mit dem im Goldstandard vorgesehenen Wert zur Clusterzuordnung verglichen. Bei diesem Vergleich wurde von oben genannte Annahme zur Beurteilung der Korrektheit ausgegangen. Stimmen die beiden Werte der Spalten überein, wurde die Clusterzordnung korrekt vorgenommen. Analog handelt es sich bei einer unstimmigen Zuordnung um einen Clustering-Fehler bzw. ein Fehlclustering. Diese Fehler werden für jedes Einzelergebnis gezählt und in einer Ergebnistabelle vermerkt. Da alle Durchläufe eine Dokumentenkollektion mit der gleichen Größe von 200 Dokumenten verwendet haben, lässt dieser Wert der Fehleranzahl eigentlich eine direkte Einschätzung der Genauigkeit zu. Zur Verallgemeinerung dieser Werte, wurde die Genauigkeit, meist als „Accuracy“ angeben, für dieses Einzelergebnis ermittelt. Dazu wurde die Anzahl der Fehler von der Gesamtzahl subtrahiert und der erhaltene Wert der korrekten Clusterzordnungen durch die Gesamtzahl dividiert.
Während vielen Anwendungen im Bereich der natürlichen Sprachverarbeitung (NLP) eine binäre Klassifikation zugrunde liegt, handelt es sich beim Dokumenten-Clustering um einen Prozess, bei dem meist eine Zuordnung zu mehr als zwei Gruppen möglich ist bzw. auch vorgesehen ist. Bei der Evaluation der Ergebnisse solcher Systeme, die sich auf auf eine binäre Klassifikation beschränken, werden oft Maße wie „Precision“, „Recall“ und „F-Score“ zur Beurteilung ihrer Qualität und zu ihrem Vergleich herangezogen. Solche Maße sind jedoch nicht zur Beurteilung der Qualität eines Dokumenten-Clusterings, das mehrere Zuordnungen erlaubt, vorgesehen. Dies liegt auch daran, dass diese gängigen Maße vor allem für Klassifikationen konzipiert wurden, bei denen auch eine „Nicht-Zuordnung“, ein „Nicht-Erkennen“ bzw. ein Ignorieren einzelner Einheiten, z.B. Wörter, möglich ist. Zwar wäre theoretisch auch ein System zum Dokumenten-Clustering denkbar, dass über dieses Funktion verfügt, die Dokumente, die sich nicht eindeutig oder überhaupt nicht zuordnen lassen, entweder zu ignorieren oder einem Cluster zuordnen, der als Obergruppe für diese diversen und nicht-zuordenbaren Dokumenten fungiert. Da jedoch das im Rahmen dieser Arbeit entwickelte System eine solche Funktion bisher nicht benutzt und deshalb jedem Dokument einen durch den Inhalt definierten Cluster zuordnet, entfällt als Folge daraus dieser Anteil an Elementen, die wie bei der Klassifikation nicht zugeordnet werden. Zudem wird keine Aussage über eine Relevanz gemacht, so wie es bei vielen Anwendungen des „Information Retrieval“ der Fall ist. Aufgrund dessen bietet sich das für das Information Retrieval meist ungeeignete Maß der Genauigkeit bzw. Accuracy in diesem Fall zur Beurteilung der durch das entwickelte System erzeugten Ergebnisse sehr gut an. Es überzeugt zudem, als intuitivste, aus der Menge der Maße in seiner Aussage verständlichste und auch in seiner Anwendung einfachste Maß.
Neben der der Clusterzordnung und seiner Genauigkeit liefert das System jedoch auch noch andere Informationen, wie das Ranking der relevantesten Begriffe des Clusters oder der Anzahl der Cluster, die das System für die Dokumentenkollektion ermittelt hat. Vor allem bei den wichtigsten Begriffen zu jedem Cluster handelt es sich um eine für den Benutzer sehr wichtige Information, die durch das System geliefert wird. Der ausschließ-

lichen Zuordnung aller Dokumente zu Cluster- bzw. Gruppenstrukturen, die nicht näher bezeichnet werden, kann der Benutzer eines solchen Systems nur geringen Informationsgehalt entnehmen. Erst die Bezeichnung dieser Cluster bzw. Gruppen durch die Zuordnung der relevantesten Begriffen dazu ermöglicht dem Benutzer ein Verständnis über das Ergebnis des Clusterings. So bildet dieser Aspekt der Ergebnisse vermutlich den wichtigsten Bestandteil der Informationen, die ein solches System zum Dokumenten-Clustering dem Nutzer liefert. Obwohl diese Ergebnisse einen so hohen Informationsgehalt besitzen, ist es ihre Eigenschaft, sehr subjektive Information zu liefern, die sich so eindeutig evaluieren lässt, wie die Genauigkeit des Clusterings.
Ein Ansatz zur Evaluation dieser Information wäre der Abgleich der vom System gelieferten Begriffe zu den Clustern mit vorher dafür festgelegten oder vermuteten Begriffen. Dennoch würde dies eine subjektive Beurteilen voraussetzen. Vermutlich ist eine Evaluation solcher Ergebnisse jedoch auch nicht zielführend oder sinnvoll.
Der letzte wichtige Aspekt, den die Ergebnisse liefern, ist die Anzahl der Cluster, die das System für die Eingabedaten selbst ermittelt hat. Dabei handelt es sich um die Anzahl Cluster, nach der der Affinity Propagation-Algorithmus die Eingabedaten clustern bzw. aufgliedern würde. Diese Anzahl bezieht sich nicht auf die im Goldstandard definierte oder durch den Benutzer eingegebene Zahl.
Zur Evaluation lässt sich dieser erzeugte Wert mit der im Goldstandard vorgesehenen Anzahl an Clustern vergleichen. Eine Annäherung des Wertes, an die vorgesehen Anzahl wird mit einem genaueren Clustering assoziiert.
Abschließend lässt sich feststellen, dass auch der im Goldstandard festgelegten Zugehörigkeiten sowie die dafür vorgesehene Anzahl an Clustern immer eine subjektive Einschätzung zur Einteilung und dem Inhalt der Dokumente zugrunde liegt.

6.3 Evaluationsergebnisse und Diskussion

In diesem Unterpunkt werden die Ergebnisse der Evaluation präsentiert und diskutiert.
Im Folgenden werden zunächst die Evaluationsergebnisse in Form von zwei Tabellen dargestellt.
In der ersten Tabelle sind die Evaluationsergebnisse nach der Kombination der Features und Clustering-Algorithmen sortiert.
Die zweite Tabelle präsentiert die Evaluationsergebnisse als Ranking, also absteigend sortiert nach der ermittelten Genauigkeit. Ein solches Ranking ermöglicht einen direkten Vergleich von Ergebnissen in ihrer Genauigkeit und es lassen sich jeweils die besten oder auch schlechtesten Kombinationen entnehmen.

Tabelle 6.1: Ergebnis- & Evaluationstabelle nach Verfahren geordnet:

Feature-Kombination	**Algorithmus**	**Genauigkeit (Accuracy)**	**ermittelte Cluster (Affinity Propagation)**
1. -w _ _ _	b) Agglomerativ	0,905	
1. -w _ _ _	a) K-Means	0,835	
1. -w _ _ _	c) AffinityProp.	0,705	10
2. -s _ _ _	b) Agglomerativ	0,825	
2. -s _ _ _	a) K-Means	0,8	
2. -s _ _ _	c) AffinityProp.		15
3. -w -d -d _	a) K-Means	0,75	
3. -w -d -d _	b) Agglomerativ	0,695	
3. -w -d -d _	c) AffinityProp.		16
4. -s -d -d _	a) K-Means	0,82	
4. -s -d -d _	b) Agglomerativ	0,775	
4. -s -d -d _	c) AffinityProp.		18
5. -w _ -d _	b) Agglomerativ		
5. -w _ -d _	a) K-Means		
5. -w _ -d _	c) AffinityProp.		
6. -s _ -d _	a) K-Means		
6. -s _ -d _	b) Agglomerativ		
6. -s _ -d _	c) AffinityProp.		
7. -w -d _ _	b) Agglomerativ	0,7	
7. -w -d _ _	a) K-Means	0,69	
7. -w -d _ _	c) AffinityProp.		16
8. -s -d _ _	a) K-Means	0,79	
8. -s -d _ _	b) Agglomerativ	0,765	
8. -s -d _ _	c) AffinityProp.		18
9. -w _ _ (unnormiert)	b) Agglomerativ	0,35	
9. -w _ _ (unnormiert)	a) K-Means	0,33	
9. -w _ _ (unnormiert)	c) AffinityProp.		38
10. Tf-idf	b) Agglomerativ	0,81	
10. Tf-idf	a) K-Means	0,79	
10. Tf-idf	c) AffinityProp.	0,69	12
11. tf-idf (unnormiert)	b) Agglomerativ	0,685	
11. tf-idf (unnormiert)	c) AffinityProp.	0,66	12
11. tf-idf (unnormiert)	a) K-Means	0,515	

Tabelle 6.2: Ergebnis- & Evaluationstabelle nach Genauigkeit absteigend sortiert:

Ranking-Platz	**Feature-Kombination**	**Algorithmus**	**Genauigkeit (Accuracy)**	**ermittelte Cluster (Affinity Propagation)**
1.	1. -w _ _ _	b) Agglomerativ	0,905	
2.	1. -w _ _ _	a) K-Means	0,835	
3.	2. -s _ _ _	b) Agglomerativ	0,825	
4.	4. -s -d -d _	a) K-Means	0,82	
5.	10. Tf-idf	b) Agglomerativ	0,81	
6.	2. -s _ _ _	a) K-Means	0,8	
7.	10. Tf-idf	a) K-Means	0,79	
7.	8. -s -d _ _	a) K-Means	0,79	
8.	4. -s -d -d _	b) Agglomerativ	0,775	
9.	8. -s -d _ _	b) Agglomerativ	0,765	
10.	3. -w -d -d _	a) K-Means	0,75	
11.	1. -w _ _ _	c) AffinityProp.	0,705	10
12.	7. -w -d _ _	b) Agglomerativ	0,7	
13.	3. -w -d -d _	b) Agglomerativ	0,695	
14.	10. Tf-idf	c) AffinityProp.	0,69	12
15.	7. -w -d _ _	a) K-Means	0,69	
16.	11. tf-idf (unnormiert)	b) Agglomerativ	0,685	
17	11. tf-idf (unnormiert)	c) AffinityProp.	0,66	12
18.	11. tf-idf (unnormiert)	a) K-Means	0,515	
19.	9. -w _ _ (unnormiert)	b) Agglomerativ	0,35	
20.	9. -w _ _ (unnormiert)	a) K-Means	0,33	
	2. -s _ _ _	c) AffinityProp.		15
	3. -w -d -d _	c) AffinityProp.		16
	4. -s -d -d _	c) AffinityProp.		18
	5. -w _ -d _	b) Agglomerativ		
	5. -w _ -d _	a) K-Means		
	5. -w _ -d _	c) AffinityProp.		
	6. -s _ -d _	a) K-Means		
	6. -s _ -d _	b) Agglomerativ		
	6. -s _ -d _	c) AffinityProp.		
	7. -w -d _ _	c) AffinityProp.		16
	8. -s -d _ _	c) AffinityProp.		18
	9. -w _ _ (unnormiert)	c) AffinityProp.		38

Zunächst auffällig ist das beste Ergebnisse mit dem Ranking 1 an der Spitze der Tabelle, mit einer ermittelten Genauigkeit von 0,9 und damit weitaus genauer als die folgenden Ergebnisse. Zum Clustering wurde der normierte ‚weight'- Wert verwendet. Zudem wurde keine Einschränkung auf direkte Schlagwörter oder Oberthemen vorgenommen. Ein weitere Eigenschaft ist die Nutzung des agglomerativen Clusterings.

Grundsätzlich lassen sich folgende Eigenschaften beobachten und daraus Schlüsse ziehen:

- Bei der Beschränkung auf bestimmte Schlagwörter gilt mit einzelnen Ausnahmen: Jegliche Form von Einschränkung auf bestimmte Schlagwörter führt zu einem Abfall der Genauigkeit. Dies gilt sowohl für die Beschränkung auf direkte Schlagwörter, wie auch auf Oberthemen. Bei der uneingeschränkten Benutzung aller Schlagwörter werden 100% (=115653) der Schlagwörter ausgewählt, von denen 85% (=99105) in mehreren Dokumenten Dokumenten auftreten und zum Clustering benutzt werden können.
  Daraus lässt sich schließen, dass die Nutzung aller Schlagwörter zu den besten Ergebnissen führt.
- Agglomeratives Clustering liefert zusammen mit K-Means die besten Ergebnisse.
- Tf-idf liefert relativ gute Ergebnisse, jedoch scheint die uneingeschränkte Nutzung der Schlagwörter die besten Resultate zu bringen.
- Eine Verwendung unnormierter Vektoren führt zu einem deutlichen Verlust der Genauigkeit und bildet somit den stärksten Negativ-Faktor beim Clustering beider Verfahren zur Feature-Extraktion.

7 Schluss

7.1 Ausblick und mögliche Weiterentwicklungen

Aufgrund des breiten Feldes, das das Dokumenten-Clustering und seine zahlreichen Anwendungen abdeckt, wären eine Vielzahl von möglichen Weiterentwicklungen und Anwendungsszenarien dafür denkbar.
[Visualisierung der Ergebnisse]
Für das Programm zum Dokumenten-Clustering existieren einige vielversprechende Ansätze zur Verbesserung und Erweiterung. Grundsätzlich umfassen diese eine Kombination seiner Komponenten und Features. Konkret wäre eine Nutzung mehrdimensionaler Features ein zentrales Ziel künftiger Entwicklungen, dies würde sich ebenfalls für ein mehrdimensionales Clustering eignen. Eine erste Implementierung wäre die Kombination von Gewichtungswert ‚weight' und Signifikanz ‚Sig' jeweils als Feature-Dimension. Aber auch die Miteinbeziehung des TZ-Types, der Diversity oder der anderen Angaben als einzelne Feature-Werte wäre eine Erforschung wert.
Neben den drei verwendeten Clustering-Algorithmen gibt es eine Vielzahl weiterer Algorithmen sowie ihrer Derivate und Erweiterungen. So werden allein in der Scikit-learn-Bibliothek viele weitere Algorithmen angeboten, wie „Mean-shift", Spektrales Clustering, „DBSCAN", „Gaussian mixtures" und „Birch". Wie bereits erwähnt, existieren zu diesen Grundtypen auch noch Derivate und Erweiterungen, die meist ähnliche, aber dennoch unterschiedliche Ergebnisse liefern und oft für eine bestimmte Anwendungen entwickelt oder optimiert wurden. Michael Steinbach et. All haben in ihren vergleichenden Untersuchungen zur Eignung von Clustering-Algorithmen für das Document-Clustering beispielsweise konnte der „bisecting K-means", eine halbierenden Variante des K-Means-Algorithmus als das am besten für diese Anwendung geeignete Verfahren ermittelt. Zum Hierarchischen Clustering wurde zudem eine Variante des agglomerativen Clusterings eingesetzt, dem „Unweighted Pair Group Method with Arithmetic mean"(kurz UPGMA), der ebenfalls überzeugenede Ergebnisse leistete.[10]
Auf Basis dieser Erkenntnisse wäre also eine Verbesserung der Genauigkeit auch im Bereich des Clusterings von Schlagwörtern zu erwarten.
Ein weiterer vielversprechender Ansatz könnte die Kombination der eingesetzten Clustering-Algorithmen sein. So wäre zunächst eine Kombination der Ergebnisse zur Bildung eines einzigen Ergebniswertes für das Clustering denkbar. Aber auch andere ergänzende Verwendungen sind denkbar, wie die Ermittlung der Anzahl der möglichen Cluster durch Affinity Propagation und der Nutzung dieser Information für andere Algorithmen, die eine Angabe oder zumindest Vermutung der Clusteranzahl benötigen.
Darüber hinaus erscheint der Ausschluss von Dokumenten mit uneindeutigem oder nicht zuordenbarem Inhalt aus der Clusterzordnung als sinnvoll. Alternativ dazu ließe sich ein spezieller Cluster einführen, dem diese variablen Inhalte zugeordnet würden. Dies würde wohl zu einer geringeren Erfassung bzw. geringeren Recall-Werten führen, jedoch wäre eine gesteigerte Genauigkeit mit Sicherheit zu erwarten.
Schließlich wäre auch ein Ansatz denkbar, bei dem die beiden im Zuge dieser Arbeit gegenübergestellten Verfahren des Schlagwort-Clusterings und und der Gewichtung durch Tf-idf kombiniert werden.

Literaturverzeichnis

[1] TopicZoom GmbH. Topiczoom. `http://www.topiczoom.de/`, 2018. abgerufen am 27. Mai 2018.

[2] Alexey Grigorev. Cluster analysis. `http://mlwiki.org/index.php?title=Vector_Space_Models&oldid=655`, 2015. abgerufen am 27. Mai 2018.

[3] Alexey Grigorev. Cluster analysis. `http://mlwiki.org/index.php?title=TF-IDF&oldid=659`, 2015. abgerufen am 27. Mai 2018.

[4] Alexey Grigorev. Cluster analysis. `http://mlwiki.org/index.php?title=Agglomerative_Clustering&oldid=606`, 2015. abgerufen am 27. Mai 2018.

[5] Alexey Grigorev. Cluster analysis. `http://mlwiki.org/index.php?title=K-Means&oldid=140`, 2015. abgerufen am 27. Mai 2018.

[6] Alexey Grigorev. Document clustering. `http://mlwiki.org/index.php?title=Document_Clustering&oldid=764`, 2017. abgerufen am 27. Mai 2018.

[7] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[8] Chris Piech. Cluster analysis. `http://stanford.edu/~cpiech/cs221/handouts/kmeans.html`, 2013. abgerufen am 27. Mai 2018.

[9] Brandon Rose. Document clustering with python. `http://brandonrose.org/clustering`, 2015. abgerufen am 27. Mai 2018.

[10] Michael Steinbach, George Karypis, Vipin Kumar, et al. A comparison of document clustering techniques. In *KDD workshop on text mining*, volume 400, pages 525–526. Boston, 2000. `http://glaros.dtc.umn.edu/gkhome/fetch/papers/docclusterKDDTMW00.pdf`.

Abbildungsverzeichnis

Tabellenverzeichnis

Inhalt der beigelegten CD

1.
2.
3.