

This is a post-peer-review, pre-copyedit version of an article published in Springer Iran Journal of Computer Science. The final authenticated version is available online at: <https://doi.org/10.1007/s42044-018-0019-0>".

© 2018 Springer Nature Switzerland AG. Part of Springer Nature.

Knowledge Discovery in Multidimensional Knowledge Representation Framework

An integrative approach for the visualization of text analytics results

Johannes Zenkert · André Klahold · Madjid Fathi

Received: date / Accepted: date

Abstract Visualization of results is one of the central challenges in big data analytics and integrative text mining. With a growing amount of unstructured data and different perspectives on big data, knowledge graphs have difficulties to simultaneously represent and visualize all analyzed dimensions of knowledge. This paper proposes integrative text mining as a solution to combine results from different dimensional analysis in a multidimensional knowledge representation (MKR) for knowledge discovery and visualization purpose. Analysis results from named entity recognition, topic detection, sentiment analysis and the extraction of semantic relationships are therefore integrated into a common representation structure. In the implementation part of this research, an application is introduced which utilizes MKR based on the results of stated text mining methods applied on a German and English news data set. State-of-the-art visualizations are used in the application and MKR adaptively transforms the visualization type of the knowledge graph according to the selected context for knowledge discovery.

Keywords Knowledge Representation · Knowledge Graphs · Text Mining · Natural Language Processing · Information Visualization

1 Introduction

Today's time can be characterized as the information age, with advances towards digitalization and connec-

tive networking provided by the Internet of Things. Technological changes are causing constant information growth, which means the production, collection and analysis of unstructured, semi-structured and structured big data. Especially for unstructured data, the volume of user-generated content has enormously increased over recent years as a result of the strong attraction of social media in the World Wide Web (WWW). Consequently, the variety of social media produced data (e.g. text, image, audio, video) and the velocity of the heterogeneous content, makes it difficult to maintain an overview of current news, events, facts, contexts, relationships and the connections between all of them. Nowadays, social media users are even facing a challenge from being influenced by doubtful information. The information overload and stated trends are likely to further increase over the next years.

Semantic technologies facilitate the connection of web data sources and enable the structured representation of relations between pieces of different information [1]. In the semantic context, information is considered to be a set of objects and meaningful relationships which can display a larger, complex picture in a simple and structured representation. To this aim, the visualization of semantic structures is typically a graph which includes graphical mappings of topics, concepts, and, or entities. In addition, these structures, such as semantic networks, contain relationships between these objects. A typical representation and visualization for large semantic networks are knowledge graphs - a graphical representation of a knowledge base. The information in the knowledge base can be organized in various forms, although the typical form of the knowledge base representation is an ontology, a collection of semantic triples or dimensional information which is stored in a multidimensional knowledge base [2]. The analysis and rep-

J. Zenkert, A. Klahold and M. Fathi
Institute of Knowledge Management and Knowledge Based Systems, Department of Electrical Engineering and Computer Science, University of Siegen, Germany
Tel.: (+49) 271 740-2142
E-mail: johannes.zenkert@uni-siegen.de

resentation of semantic relationships between entities, the information extraction, or the inference on knowledge graphs, are only a few examples for the utilization and application of knowledge graphs. Large knowledge graphs have been established in the past years. Freebase [3], WikiData¹, Yago [4], NELL², Microsoft Concept Graph³, and Google Knowledge Graph⁴ are typical examples.

Integrative text mining is the process of information extraction from textual resources in various analysis perspectives and the combination of the results for further analysis potentials. In general, text mining provides a set of different statistical methods for computers to understand textual unstructured content. Typical applications are text classification or clustering, topic detection, entity recognition, sentiment analysis, automatic document summarization or knowledge fact extraction. With the possibilities of different text mining analysis, topic, sentiment, entities and facts can be determined and represented in separate visualizations. However, in the approach of integrative text mining, different analysis perspectives are combined in order to gain additional information or to exploit further analysis potentials.

Simitsis et al. used a multidimensional structure for analysis and exploration of content by combining keyword search with Online Analytical Processing (OLAP) aggregation, navigation, and reporting [5]. Zhang proposed the usage of topic modeling for OLAP on Multidimensional Text Databases (MTD) [6]. Lin et al. define Text-Cube model on multidimensional text database [7]. Similarly, the proposed Multidimensional Knowledge Representation (MKR) as a framework for text analytics has been conceptualized by the authors [2]. This article is an extension of the approach. It shows how MKR can be applied in practice as a means of integrating various results into a common representation structure and how the benefits of MKR affect visualization and knowledge discovery. With MKR rich information is captured and put into the correct context. Moreover, in this paper we show how to utilize MKR in implementation, and for example how results from other analysis dimensions are used for dimensional selection or filtering in knowledge graphs. Therefore, the representation structure allows the adjustment on knowledge graph visualizations without any further analysis or calculation. Thus, by the selection or filtering of dimensions, additional analysis results are achieved, which were initially not considered. Both, MKR as framework based

on integrative text mining results and the visualization of MKR by knowledge graphs are discussed in this research.

The article is structured in five sections. Section 2 is a literature review in the area of knowledge graphs, big data analytics, integrative text mining as well as the related methods. Section 3 discusses the MKR and provides further information about the visualization possibilities of analyzed textual content. Section 4 discusses the preparation of the data set and the implementation of this research. Moreover, the section discusses the results which have been produced by the application. Section 5 provides the conclusion and future work, which also summarizes the results.

2 Literature Review

This article introduces MKR as a knowledge base representation structure for multidimensional information from unstructured data in large-scale knowledge graphs. Therefore, the state of the art in the fields of knowledge graphs, big data analytics, integrative text mining and related methods have been considered and reviewed.

2.1 Knowledge Graphs and Visualization

Knowledge graphs which illustrate and represent semantic knowledge are mostly based on the Resource Description Framework (RDF) [8]. Therefore, this section explains what RDF is and how it is used to semantically organize knowledge.

The internet offers a wealth of information, which has drastically increased by the widespread use of the WWW and information sharing across the private sector. In fact, a very large part of the web data has been mostly created by human end users and is therefore designed to meet human readability requirements. It is easily understandable and associable with other information created by the human mind. Computers, however, cannot easily find interconnections between different web resources. More specifically, the task to model the cognitive process of finding relationships is extremely complex.

A common framework to describe interconnections and relations between web resources is the Semantic Web. It is based on the idea of mapping the information in a machine-readable form. To this aim, the World Wide Web Consortium (W3C) has defined several standards in recent years to facilitate this process. This efforts resulted in specifications for the extensible markup language (XML), RDF and the Web Ontology Language (OWL) [9]. The latter two refer specif-

¹ <https://www.wikidata.org/>

² <http://rtw.ml.cmu.edu/rtw/>

³ <https://concept.research.microsoft.com/>

⁴ <https://www.google.com/intl/es419/insidesearch/>

ically to knowledge bases or the knowledge acquired from documents in an application area [10]. RDF was developed in the 1990s and was officially released in 1999 by the W3C. While the focus was mainly on the metadata (data about a given data set) [10], the Semantic Web increasingly influenced the use of RDF. It is now a formal language for the description of structured information [10], with which data on the web can be exchanged without loss of meaning, which makes a difference to XML and HTML (typically responsible for the illustration of information). A revised RDF specification was published in February 2004 [10].

The knowledge graph representation of RDF or a collection of RDF documents is made by the idea of a directed graph, consisting of nodes which are connected by directed edges. Unique identifiers are assigned to both elements. In contrast to XML, RDF has not been designed for hierarchical structures, but for the general description of relationships between resources. In addition, RDF has been developed for the description of data on the WWW and other electronic networks with decentrally store information. The combination of those resources is not a problem in RDF, whereas tree structures like XML are hardly suitable for this.

As mentioned, the nodes and edges in RDF have an identifier. This is motivated by the fact that resources in different data sources are maybe the same, but are not identical, or identical, but described differently. To counteract this, RDF uses an Uniform Resource Identifier (URI). Online documents in the WWW can be uniquely identified by the Uniform Resource Locator (URL), while offline resources don't have a URL, which are then referred to by Uniform Resource Names (URN). URNs and URLs complement each other.

Although the graph-based representation of RDF is descriptive and comprehensible, it is not directly suitable for processing in computer systems. Therefore, a serialization is needed which is the conversion of complex data objects into character strings [10]. This converts the RDF description into a syntactic form which is machine-readable. Each edge of an RDF graph is defined by its start and end point, as well as its label. In this way, an RDF statement is composed by these three components, the subject, predicate, and object [11], and form an RDF triple [10]. Ultimately, the representation of a graph for semantic relationships is realized by a collection of triples.

A method for the extraction of semantic triples (see 2.4.4), which is a related method from the field of text mining, also uses RDF to represent automatically extracted knowledge facts from textual resources.

Information about data sources, their collection and the characteristics of big data analytics in the relation of knowledge graphs are given in the following section.

2.2 Big Data and Big Data Analytics

Big data defines very large volumes of data in a wide range of applications. It has become a popular term used to describe the exponential growth, availability, and use of information, both from structured and unstructured data [12]. The application of advanced data analysis methods in real or near-real time is called big data analytics. Big data analytics is the process of investigating large amounts of data with heterogeneous data types to discover hidden patterns, unknown correlations, market trends, user preferences, and other useful information [13]. The results of big data analytics can provide important insights into areas such as the behavior of customers based on social media analysis with regard to a particular product [14]. These results enable companies to improve their productivity and efficiency, and to make efficient and well-informed decisions [13]. Fast big data technologies for big data analytics are using in-memory technologies and apply distributed parallel processing to handle data from different sources [15]. In this way, data processing is much faster compared to conventional processing and storage techniques. For large-data scenarios with an increasing volume of data, NoSQL databases are particularly suitable. These databases do not rely on any fixed schema and are therefore compatible with many data types. Furthermore, they allow data formats and structure to be adjusted without affecting the used application landscape.

Big data is often characterized primarily by the large size of the information. However, there are also other important aspects; typical features are volume, data diversity (variety), and a high speed of data generation (velocity).

Volume: The first feature of big data is the amount of data. Companies are increasingly managing larger amounts of data. This is the case of Google, Yahoo or Facebook, for example, which stores about 500 terabytes of new data per day. A total of more than 4.4 zettabytes of data exist today. The requests from the approximately 1.37 billion daily active users on Facebook⁵ (average for September 2017) must be processed simultaneously. Relational databases are not very efficient for these data sets, therefore database technologies

⁵ <https://newsroom.fb.com/company-info/>

with horizontal scalability are necessary. It becomes virtually impossible for a server to process such data alone because the data processing times are too long. The data from current statistics predict that 40 zettabytes of data could be reached in 2020.

Variety: Another important feature of big data is the variety of data. Big data is the storage of structured, semi-structured and unstructured multimedia data (text, images, audio and video) [16]. In fact, most of the big data is unstructured data. The unstructured data is mostly information that cannot be stored in a relational database or conventional data structure. Latest statistics show that more than 80 percent of all available data is unstructured. These data come mostly from different social networks like Facebook, Twitter or YouTube. Such large, polystructured data sets (consisting of numbers, texts, images, videos, relationship data, etc.) have high potential in prediction and forecast analysis [17].

Velocity: The third mentioned aspect of big data is the velocity. Velocity means the speed at which data is generated, stored, analyzed and processed [18]. Big data needs to be processed quickly to help organizations in making decisions. In the figurative sense, the speed of the data is becoming increasingly faster. For example, sensors, which are placed on streets to analyze road traffic, capture thousands of data values per minute. This data needs to be processed in real-time to predict traffic. Consequently, the speed of data processing is important in predicting of our climate, financial markets, and many other areas which apply forecasting models and methods.

The analysis of unstructured textual data sources from different perspectives is called integrative text mining. A literature background is given in the following section.

2.3 Integrative Text Mining

The terms text mining, web mining and data mining are often used in a similar context, although they are different application areas. The term text mining is also often associated with data mining as its upper category. Alternatively, text mining and web mining are described as a special form of data mining [19]. Data Mining, also known as Knowledge Discovery in Databases (KDD), is concerned with the search and analysis of large amounts of data and the analysis of recognizable patterns and rules [20]. The core idea here is the extraction of useful, reusable information within the data sets. For example, data mining techniques are used in the field of market research in order to recognize the customer's buying

behavior and to evaluate it accordingly. In addition, future behavioral rules can be derived from the data. For example, within a purchase, milk is purchased with a probability of 90%, if bread and butter are part of the purchase [20].

Text Mining, sometimes synonymously stated as Knowledge Discovery from Text (KDT), is not a clearly defined term [21]. There are different views and definitions depending on the application. Essentially, methods of information retrieval (IR), information extraction (IE), and Natural Language Processing (NLP) are used and combined with KDD methods. On this basis, several definitions can be distinguished which are based on the discipline's view. In IE, for example, text mining aims at the concrete extracting of facts from texts [21]. Text Mining can be also understood as a text data mining process for recognizing meaningful patterns in texts by applying algorithms from the field of machine learning (ML) and statistics. In this case, preprocessing steps and, where appropriate, NLP operations are necessary to prepare the text, so that subsequently adapted data mining methods can be used [21]. Furthermore, text mining can be seen as a KDD process, whereas several individual (pre-processing) steps have to be done in order to apply data mining methods for statistical analysis [21]. As an extension, integrative text mining is an approach where individual, specialized text mining methods are applied in combination in order to process and analyze data from multiple perspectives.

Many different areas influence text mining. The previously stated IR, IE, NLP, and ML are important to be mentioned. In the case of IR, typically search requests from users are processed by finding suitable documents. The automatic processing of text data is achieved by using statistical measures and methods to transform text into a suitable mathematical model. IR has been studied in the scientific field for a long time and was initially used for the indexing of documents [22]. It was again given greater importance by the dissemination of the WWW and the associated requirements on mature search engines. IR in the traditional sense deals with questions and the associated answers. However, document retrieval systems based on indexing are often assigned to the IR field [21]. IE aims to extract information relevant to a context from text documents and to make it available (for example in the form of databases) [23], [21]. NLP should provide better understanding of natural language when using computers [24]. NLP tasks include a range from the simple processing of character strings up to the automated question answering with regard to natural language queries [21]. ML is related to artificial intelligence and involves the development of methods for learning by analyzing data. ML can, for

example, independently analyze unknown textual data and apply already trained classifiers based on acquired knowledge or experience.

Specialized methods from the text mining field which are from high importance for MKR are introduced in the following section.

2.4 Related Text Mining Methods

This section provides a description of related methods from the integrative text mining domain. Named entity recognition, topic detection, sentiment analysis and semantic triple extraction are described in the following as specialized text mining analysis for the construction of MKR based knowledge graphs.

2.4.1 Named Entity Recognition

Named Entity Recognition (NER) is a process of IE, in which specifically named persons, organizations, locations and other entities are identified within texts. In many NLP, IE, and IR systems, it is one of the first steps [25] and a core task, for example, in automated summarization or machine translation [26]. Moreover, for knowledge extraction, knowledge graph construction and more specifically, the extraction of semantic triples (in the form of subject, predicate, and object), NER is of considerable importance, since names often represent subjects and, in some cases, objects and therefore must be recognized.

According to Nadeau and Sekine, NER methods are classified according to their learning approach [27]. Therefore, there are existing supervised methods that use, for example, Hidden Markov Model (HMM) [28], Decision Tree [29], Support Vector Machine (SVM) [30], Maximum Entropy Model (MEM) [31], or Conditional Random Field (CRF) [32]. Semi-supervised approaches require at least small amount of user input, such as a training set. The process is called bootstrapping, in which case the user could provide example names that allow the system to recognize other named entities from similar contexts [27]. The unsupervised methods depend on the analysis. Clustering for example is often used to create named entities from contextually similar groups.

NER is a complex task that depends on many factors such as the language of the data. In English, for example, good candidates for named entities can be already guessed by large initial letters [33]. In contrast, in the German language, this rule-based approach is not possible due to a completely different grammar. Other writings, character or symbol languages need again different approaches.

In general, two variants are available for the realization of NER, the rule-based variant and the list-based variant [27]. While the rule-based variant is based on grammatical, statistical or other information (whether a word, word tuple or n-gram is a named entity) the list-based variant takes potential names and compares them with dictionaries or lexicons from named entities. There are also mixed forms of both variants.

Previous studies like the one from Nadeau and Sekine [27] show that three learning methods are differentiated in order to recognize named entities. At the beginning of the research in the area of NER, emphasis was placed on the recognition of entities by means of manually created rules. Over the course of time, the supervised learning method of ML was applied for NER. With the help of large annotated document collections, also referred to as corpus, the properties of the entities are studied with positive and negative examples and rules for recognition are automatically created. Supervised methods can be described with the following similarities [27]: 1. Reading in a large annotated corpus; 2. Save a list of entities; 3. Creating disambiguation rules based on exclusionary properties.

If no corresponding corpus is available, the semi-supervised learning or the unsupervised learning methods from ML could be used [27].

Previous NER research on the MUC-6 and the MUC-7 corpus show improvements in the NER task. On the basis of the MUC-6 training data Palmer and Day did a transfer of 21% [34]. 42% of the place names, 17% of the organizations and 13% of the family names were reproduced [34]. Mikheev et al. calculated 76% for location names, 49% for organizations, and 26% for family names with a precision of 70% to 90% on the MUC-7 corpus [35].

For NER based on supervised learning, statistical machine learning forms the foundation. It is based on sequence labeling problems such as other NLP methods (e.g., Part-of-Speech (POS) Tagging) and is a general problem in machine learning. Jiang formulates it as follows: The sequence of observations is given as $x = (x_1, x_2, \dots, x_n)$. Each observation is represented by a feature vector [36]. Each observation x_i is assigned to a label y_i . The label for y_i can be predicted based on x_i in standard classification. However, it is assumed that the label y_i does not only depend on x_i , but on other observations and labels in the sequence. Typically this dependency is limited to observations and labels from close neighbors in the sequence of current position i [36]. It is further important to note that the named entity boundaries and named entity types need to be considered. The BIO notation known from text chunking is useful in this regard [37], [36].

2.4.2 Topic Detection

Topic detection is a classification task. The classification includes the assignment of documents or parts from documents to predefined classes. Texts (or text fragments) can, for example, be assigned to related subjects (sports, politics, business, culture, etc.). In this case, a training set (a predefined amount of documents) is typically used, where documents have already been assigned to existing classes. The goal in this statistical approach is to derive a model for classification.

A new document is typically compared with documents already allocated to topics. The higher the number of matching features from the new, unassigned document, compared to the documents already classified, the higher is the likelihood to also be related to the comparison document's topic. One of the methods for generating document features is the *TF-IDF* measure [38]. It is a combination of the term frequency (TF) and the inverse document frequency (IDF). TF-IDF evaluates single terms from a document and compares them within a document collection. If a term occurs in a small number of documents from the collection and occurs multiple times within a document, it is considered very characterizing.

For the evaluation of assigned documents to topics, the key figures precision and recall are well known. In the former, the relevant (correctly classified) documents are compared with all recorded documents, while recall indicates the relationship to all relevant documents [21]. However, if one of the key figures is to be improved, this inevitably has conflicting consequences for the other. This conflict can be analyzed by determining the F-Score. The F-Score contains both key figures and thus offers an attempt at a compromise [21].

Latent Dirichlet Allocation (LDA) is a well-known topic model, where each document is understood as a mixture of topics, and topic can be specified as a discrete probability distribution, which tells how likely a word belongs to a given topic [39]. Latent Semantic Analysis (LSA) is a method for extraction and representation of contextual meaning of words by statistical computations applied on a large text corpora [40]. Compared to LSA which stems from linear algebra and performs a singular value decomposition of co-occurrence tables, Hofmann proposed Probabilistic latent semantic analysis (PLSA) which is based on a mixture decomposition derived from a latent class model [41].

In addition to those models, it is also possible to carry out next-neighbor classification on textual data. On the basis of the texts already assigned, a comparison is made with new documents, which are then sorted into the same class as the one to which it is most simi-

lar. The similarity is determined, for example, by coinciding words within the texts [21]. Further similarities are given in [42]. Other alternatives are binary decision trees or classification rules which are discussed in [43].

For the detection of multiple topics within text document structures, Klahold et al. defined Associative Gravity as a new method to separate documents into topic-related clusters [44]. The method is based on the text mining method entitled CIMAWA, which imitates the human ability of word association [45]. Different topics are identified within documents by utilization of the word association strength between identified keywords.

2.4.3 Sentiment Analysis

Sentiment analysis is a method related to text mining [46][47]. For the concept of sentiment analysis, different terminologies can be found in the literature. Frequently used terms are opinion mining, sentiment analysis and subjectivity analysis [48][47]. Others are using the expression polarity detection [49]. The inconsistent terminology is mainly due to the different backgrounds of the researchers. While the term opinion mining has its origins in the fields of web search and IR, the terms sentiment analysis and subjectivity analysis come from NLP research [48]. Although these terms may be used for slightly different tasks or viewing angles, they represent the same research field [50], [51]. Broß defines sentiment analysis in general as a research field that uses NLP techniques to automatically identify and analyze subjective information in natural language texts [52]. One of the objectives is to determine which subjective information is expressed against which objects (entities) in the text [52]. Subjective information can manifest itself both explicitly through the expression of opinions or facts, as well as explicitly in the author's attitude [52]. In the literature the sentiment analysis was essentially examined on the following three levels [53]:

Document: At this level, the task of sentiment analysis is to determine whether an entire document expresses a positive or negative sentiment [51]. This form is known as a "document-level sentiment classification" and can, for example, be used to determine whether a product review overall expresses a positive or a negative sentiment about the product [51]. Since a single document can contain several topics and thus different sentiments, the assignment of a single sentiment for the entire document is generally inaccurate.

Sentence: The task of the sentiment analysis on the sentence level is to recognize which sentences of a document express a positive, negative or neutral sentiment [51]. In the example of the product review, the sentiment analysis at the sentence level allows to examine which sentences of the reviews express a positive, negative or no/neutral sentiment about the associated product.

Entity and Aspect: Sentiment analysis on entity and aspect levels (also called aspect-oriented sentiment analysis) is a fine-granular form of sentiment analysis. Unlike the other levels, the sentiment is analyzed as to what the author likes or dislikes [51]. There are no documents, paragraphs or sentences as a whole in the focus, but the opinion itself [51]. It is based on the assumption that an opinion consists of a sentiment and objectives (entities), towards which the sentiment is expressed [51]. In this regard, objectives of the sentiment can be entities or their different aspects [51]. In the example of a product review, the reviewed product corresponds to an entity, while product properties represent aspects of the entity.

The polarity in sentiment analysis can be generally divided into positive and negative [46]. It is thus a binary sentiment classification. An example of another binary sentiment classification is the Agreement Detection [47]. Kaur, Gupta and others, on the other hand, have a neutral polarity as a third possibility of polarity detection [49].

The importance of sentiment analysis has increased since Web 2.0. Forums, blogs, review portals, social networks etc. are becoming increasingly popular. Here, people write their opinions and feelings on topics that move or interest them. Thus there is an increasing interest in sentiment analysis from a number of external parties, particularly marketing departments. Due to the enormous number of texts and information, automated sentiment analysis is indispensable in this regard.

New approaches in sentiment analysis are emotion detection, transfer learning, building resources [46] or the analysis of word associations for sentiment evaluation [54]. Emotion detection is a task of sentiment analysis, which in contrast to classical polarity detection is divided not only in positive and negative. The goal is to recognize concrete emotions. Medhat et al. call the eight emotions joy, sadness, anxiety, fear, trust, disgust, surprise and anticipation [46]. In transfer learning, sentiment classification are transferred from one domain to another. This means, the knowledge gained in a domain is also used to improve the learning process in another domain. Building resources means that resources are created that support sentiment analysis. For example,

dictionaries and corpora are created in which polarities are assigned to the individual concepts [46].

There are a number of methods that are applied in the field of sentiment analysis. Basically, the methods of sentiment analysis can be divided into two broad areas. These are, on the one hand, the lexicon-based and ML approaches [46].

Lexicon-based: The lexicon-based sentiment analysis approach is again divided into the areas of dictionary-based approaches and corpus-based approaches [46]. The dictionary-based approach begins with words whose polarity is already known. Now, in word databases such as WordNet, synonyms and antonyms for these words are searched. The found words are then added to the original list. The process is iteratively repeated until no new words can be added to the dictionary [46]. One of the strengths of the corpus-based approach is the search for opinions with context dependency. The corpus-based approach can in turn be divided into two areas, the statistical and the semantic part [46]. The statistical approach considers the occasional frequency. If a word is used more often in positive texts, its polarity is positive. For a more frequent use in negative texts, accordingly negative. If the word is used evenly, the polarity is neutral. On the other hand, the fact is used that similar expressions are frequently used together in one text. If these words are identified, they can be assigned the same polarities [46]. In the semantic approach, in which semantic similarity between words is considered, similar polarities are assigned if two words have a strong semantic similarity. The semantic approach can also be combined with the statistical approach [46].

Machine Learning: The ML-based sentiment analysis methods are divided into the categories supervised and unsupervised learning methods. Supervised machine learning is dependent on the existence of annotated test data to train the methods. Since it is often difficult to obtain such suitable test data, unsupervised machine learning is an alternative. Here, for example, keyword lists and similarity measurements are used to classify the texts [46]. A detailed overview of the methods of sentiment analysis is given by Medhat et al. [46] and Varghese & Jayasree [53].

2.4.4 Semantic Triple Extraction

Akbik and Broß introduced “Wanderlust”, a system by which semantic relations can be derived from grammatically correct English text [55]. The basic idea of this approach is the identification and usage of grammatical connections between words and sentences. Their

assumption is that syntactic, grammatical connections, also imply semantic associations. Therefore, the aim is to derive triples consisting of a subject, predicate, and object, which should correspond to the statements in RDF [55]. The link grammar (identifying relations between pairs of words) allows to determine the linkages within sentences by means of a detailed linguistic analysis of the sentences [56]. Unlike a PoS Tagger, the individual words are not marked by their own characteristics (tag), but rather their connection to subsequent words. The link between two different words is called linkpath. Akbik and Broß explained it further with the example “Essen” (start term) “is” \rightarrow “city” \rightarrow “in” (wordpath) \rightarrow “Ruhr Area” (end term) [55]. The complete predicate in this example “IsCityIn” can be generated by the concatenation of the elements from the wordpath.

The important condition here is that no further additional semantic information, but only the grammatical structure of the sentences was used to determine RDF-like triples. Akbik and Broß further integrated NER into their system and created a training set with 46 linkpaths for the analysis of Wikipedia content [55].

The presented method from Akbik and Broß facilitates an initial semantic analysis on text resources based on grammatical structures. In this way, well-written textual resources can be easily (rule-based) scanned for contained entities and their given relationships. Compared to other methods from semantic analysis or association measures, the presented method does not rely on any large semantically annotated text corpus, ontology or any other training data. However, the facts which are acquired from the grammatical analysis are much lower in the number of occurrence, but with very high precision. Of course, the method cannot be used on content which is not written in sentence structure or without a correct writing style and grammatical correctness.

3 Multidimensional Knowledge Representation

Multidimensional Knowledge Representation (MKR) for text analytic results has been conceptually proposed in related work by the authors [2]. In this article, the practical implementation of MKR and the aspect of method combination through integrative text mining in order to utilize MKR are focused.

MKR is a representation format which allows the combination of different analysis results from integrative text mining and related methods in the knowledge base. Therefore, the results which are achieved by synchronously or asynchronously calculated text mining methods are considered as dimensional information and

inserted in a uniform representation structure. By the combination of different analysis perspectives and their results, the knowledge base is able to provide a broad overview on gained information and uses it for further analysis and requests. In this way, the integration of the analysis results into a holistic knowledge representation structure offers advantages in visualization of data and knowledge base inference.

While normal representation structures are limited to and based on a specific predefined analysis result, multidimensional representation structures are able to offer further unexpected analysis results, context or additional information for the process and methods of knowledge discovery. This is especially reflected in the knowledge visualization since suitable knowledge graphs can add further results from different other analysis dimensions or perspectives for adjustment or adaptation. In this way, the results are used to contextualize, filter, compare, combine or even for mining of unexpected results in further analysis.

As an example, if a knowledge base is queried for entities which are directly related to a specific topic in a given time frame, MKR can directly offer more information about sentiment information or extracted facts, both related to the entities, time frame and topics. The visualizations of the entities, e.g. entity type-specific nodes with a color indicator from red (negative) to green (positive), are used in this scenario to show the sentiment relationship within in the same visualization while, if desired, edges could indicate a semantic relationship or other nodes indicate a topic relationship or influence the layout arrangement.





As this example shows, MKR is specifically meaningful in the knowledge visualization since the results from other analysis methods are directly added to the main visualization without modifying the knowledge base query at all. Table 1 provides an overview of the typical visualization possibilities for different text mining methods and MKR provided modification.

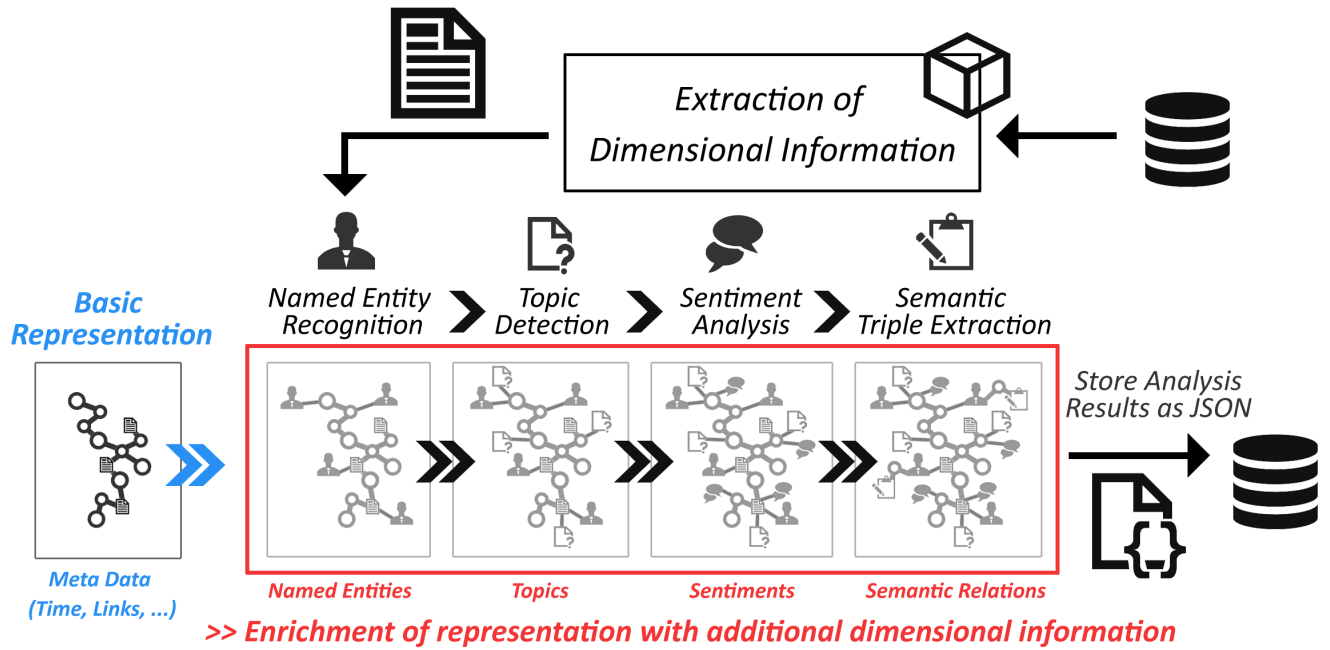
In the next section, the process of MKR to enrich basic knowledge representation is explained in detail.

3.1 Processing and Representation

Typical knowledge management tools are Document Management Systems (DMS). DMS normally use indexing on various variables or features (e.g. meta data) to browse and search for documents in the knowledge base. The representation of documents are typically a set of properties in a relational database. With schema-less structures in NoSQL databases, representation methods are allowed to be more flexible. In this way, the

Table 1 Typical Visualization Types for Text Mining Analysis Results

	 Tile View	 Topic Map	 Area Chart	 Bar Chart	 Scatter Chart	 Entity Graph	 Map Chart
 Named Entity Recognition	Tiles show corresponding entities. Size and color is variable on type or number of occurrence.	Topic rectangles show corresponding entities. Size and color is variable on type or number of occurrence.	Areas show the number of entity occurrence in text collection over time.	Bars (stacked) indicate the number of entity occurrence in text collection over time.	Markers show the number of entities which have been identified in text collection over time.	Documents are represented as node collection based on content and included entities.	Parts of the map chart (countries) are colored based on the number of occurrence of country names which have been identified.
 Sentiment Analysis	Tiles are colored in green-red scale based on sentiment evaluation.	Topic rectangles are colored in green-red scale based on overall sentiment evaluation.	Areas show the number of positive, neutral or negative evaluated texts from a collection over time.	Bars (stacked) show the number of positive, neutral or negative evaluated texts from a collection over time.	Markers show the calculated sentiment evaluation of texts from a collection over time.	Nodes visualize documents in different colors in green-red scale based on overall sentiment evaluation.	Parts of the map chart (countries) are colored based on their (document) related sentiment evaluation.
 Topic Detection	Tiles are in different colors which represent different topics.	Topic rectangles are colored in different colors that represent different topics.	Areas show the number of texts which have been assigned to different topics from a text collection over time.	Bars (stacked) show the number of texts which have been assigned to different topics from a text collection over time.	Markers show the number of texts which have been assigned to different topics from a text collection over time.	Nodes visualize documents in different colors that represent different topics. Nodes are connected with topic node.	Parts of the map chart (countries) are colored based on their related topic.
 Semantic Triple Extraction	Tiles visualize Subjects or Objects from extracted triples (Subject - Predicate - Object).	Topic rectangles visualize Subjects or Objects from extracted triples (Subject - Predicate - Object).	Areas show the number of texts over time in which Subjects or Objects have been identified.	Bars (stacked) show the number of texts over time in which Subjects or Objects have been identified.	Markers show the number of texts over time in which Subjects or Objects have been identified.	Nodes and edges visualize the extracted triples (Subject - Predicate - Object). Relationships are visualized.	If Subject or Object is a location, parts of the map chart (countries) are colored.

**Fig. 1** The Process of Multidimensional Knowledge Representation (MKR)

basic document representation format from knowledge bases can be dynamically enriched by additional dimensional information or analysis result whenever it is available. The support of this enrichment process is the main intention of MKR.

MKR wants to prevent multiple queries on the knowledge base by the combination and storage of different text mining analysis results in one representation for-

mat. The contained additional pieces of information are provided by the methods of integrative text mining, namely named entity recognition, topic detection, sentiment analysis and the extraction of semantic relationships. It is notable, that other text mining methods can be added to the MKR process and into the same resulting representation structure. As an example long-term and short-term word associations from entities within

the document could be added into MKR and would be available for temporal analysis on documents over time without re-calculation.

Pre-processed documents are taken from the knowledge base and analyzed essentially in four steps: 1) Extraction of named entities 2) Detection of topics 3) Determination of sentiments 4) Extraction of semantic relations. Afterwards, the analysis results are collected, stored and represented in the database in JSON format. Fig. 1 illustrates the MKR process.

It is further important to mention, that the MKR process does not necessarily need to follow a four step process. Methods could be skipped, other methods included, or, even executed independently - if no dependencies. Due to schema-less representation of MKR JSON format, results can be added in the knowledge base whenever they appear or have finished calculation.

The next section provides a more detailed description about the visualization possibilities of MKR. State-of-the-art visualization methods are used to explain the benefits of MKR and the utilization of them as multi-dimensional visualization instruments.

3.2 Adaptation of State-of-the-Art Visualization

In the following, visualization types will be explained which can be mostly considered as state-of-the-art visualizations from related work in data mining and knowledge discovery [57][58]. By shortly explaining the main goal of the visualization types, the advantages of MKR should be highlighted by description of the visualization modification and adaptation possibilities mentioned in Table 1.

3.2.1 Overview Graphs

Overview visualizations are used to provide a summary of the most important, the latest, or the most unexpected information at a given time. In the following, the instruments tile view and topic map are presented as overview visualization tools which work very well in combination with MKR.

Tile View: The tile view is an overview visualization which represents a selected set of items. The items can typically vary in size and/or color. The number of items is also variable. For textual data, a tile normally contains a summary of the textual content or the first part of the text as a preview. Here, results from named entity recognition, sentiment analysis or topic detection, but also semantic triples can be used to modify the colors and size of the tiles. For results from named entity recognition, images are also useful to represent the

text's containing entities and they can be used for visualization purpose in the background of a tile for example.

Topic Map: The topic map is another overview visualization which represents a selected set of items. The items can typically vary in size and/or color. Each part of the topic map represents a dimensional information at a certain hierarchy level. Fig. 2 a) provides an example of the topic map visualization. Each rectangle represents the topic of a collection of documents in which the topics have been recognized based on a topic detection method and the analysis results have been stored in MKR. Related subtopics, also referenced to topics and represented in MKR, are further visualized here - in order to preview the distribution of the document collection based on rectangle size. The topic map is interactive and allows to navigate to different hierarchy or dimensional levels. For example, a navigation from all topics into a specific topic is shown in Fig. 2 c) where the topic "society" has been selected and the layout of the topic map has been changed to visualize related entities of type "person" represented in MKR (similar to a drill down operation). As shown in Fig. 2 c) the small collection of rectangles inside the topic indicate the number of occurrence of the corresponding entity in the document collection. Further drill down into the document subset which matches the "society" topic, and a specific entity is again possible to select and so forth. In general, MKR structure enables the adjustment of the topic map in desired dimensions. Another example is shown in Fig 5 in which the sentiment analysis results from MKR representation are used as another filter.

3.2.2 Statistical Graphs

Statistical graphs are usually used to visualize information from a category (e.g. knowledge dimension) over a given time dimension. Therefore, statistical graphs typically represents the time dimension with an adjustable scaling. Area chart, bar chart and scatter chart are typical examples and are explained in the following.

Area Chart: In the area chart different time series are represented in the form of stacked areas. All data values of the time series represent a summary of analysis results at a given time (or time period). Colors and the appearance as well as number of stacked areas are modified based on selected dimensions.

Bar Chart: Typically, an amount of data is represented horizontally or vertically at different given discrete values (e.g. time stamps) in order to create a bar chart.

Bars can include different categories and represent analysis results in stacked or grouped form. Fig. 2 b) illustrates an example in which the topic map has changed to a bar chart while the same result MKR representation is active. Here, different topics are visualized together with a time dimension and show the number of documents related to topics over time.

Scatter Chart: The scatter chart is a visualization type which represents data observations in two dimensions. Each point in the graph (also called marker) represents one observation in the data from one dimension at a specific value from another dimension. Points can vary in size and color to visualize additional information. Fig. 2 d) illustrates an example in which documents from “society” topic have been arranged by MKR’s sentiment analysis result. Each marker represent the number of documents (modifying the marker size) within a configurable interval over time. The color indicates sentiment values on a scale from positive (green) to negative (red).

3.2.3 Entity Graphs

The entity graph is a visualization type which uses nodes and edges to show relationships between data. Each node typically represents an entity or a collection of entities. Nodes and edges can vary in size, distance and color to illustrate additional information. The layout of the entity graph can be adjusted based on several layout algorithms. Fig. 2 e) illustrates an example in which documents are visually connected to the extracted entities from the textual content. The entity graph is the best option to visualize the semantic relationship from extracted entities and RDF triples (e.g. nodes for subjects/objects, and labels for predicates).

3.2.4 Map Graphs

Documents which contain named entities with type “location” can be used for abstract visualization in a map graph. Each named geographic location can be used to highlight different parts of the map. For example, countries are visualized on the map and can be used to show analysis results. Fig. 2 f) illustrates an example with countries which are mentioned as named entities in the contents of a document collection. The color intensity is for example based on the frequency of named entity occurrence or based on other method results, stored in MKR, such as the sentiment or topic relationship.

3.3 Transformation

MKR enables an easier transformation from one knowledge visualization into another because of the generic representation structure and combination of dimensional analysis results. Dimensional selection and dimensional filtering are two potential MKR operations which can be used to adjust the MKR analysis results for visualization purpose. An overview about exemplary visualization transformations is given in Table 2. It further describes use cases of MKR by potential combination of different dimensions in different visualizations.

Dimensional selection separates the knowledge base’s MKR dataset into a smaller subset by choosing one or multiple values for the data source, time, or language. In this way, a specific subset from the knowledge base can be prepared or adjusted dynamically.

The dimensional filtering operation creates a smaller MKR subset by filtering of analysis results with specific values from one or multiple dimensions. The analysis results and therefore searched knowledge items, represented by MKR in the knowledge base, can be greatly reduced through dimensional filtering. As an example, a dimensional filtering creates a knowledge base query, which executes a search for documents which are related to a specific topic (e.g. politics), have a specific range of sentiment (e.g. positive) and contain at least the occurrence of specified entity (e.g. White House).

4 Results and Discussion

4.1 Implementation

The main implementation of this research was created in C# and the Windows Presentation Foundation (WPF) framework. For the data collection task, a script written in R has been used to externally crawl German and English news portals on distributed machines and storage in a NoSQL database (MongoDB). PhantomJS (a headless browser used for automating web page interaction) is executed by the R script to load and iteratively crawl the content of websites. A test dataset has been created between September 2016 and April 2017. The dataset is about 19.7 GB in size and includes 305,281 articles, together with images and pre-processed content from different typical news domains (e.g. finance, economy, politics, sport, lifestyle, culture, science and others) with an average length of 6,115.77 characters and 618.56 words each. German and English languages have been chosen by the authors to see how the implemented methods can be adjusted or directly used on both languages and to identify language-specific prob-



Fig. 2 Knowledge graph types and different visualizations of MKR representation: a) Topic map with subtopics, b) Bar chart which shows the number of documents in different topics over time, c) Topic map with selected topic “society” - visualizing all named entities from type “person”, d) Scatter chart with sentiment analysis results from documents in the “society” topic, e) Entity graph with visualization of documents from the topic “society” - connected with named entities, identified by named entity recognition (and represented in MKR), and other semantically related knowledge items, f) Map graph visualization which indicates the reference of named entities from type “location” within documents from the topic “society” - visualizing the result of document’s sentiment analysis.

lems in order to solve them in this research and future work.

Fig. 3 illustrates the whole implementation workflow. More specific descriptions of the implementation parts are given in the following sections.

Table 2 Knowledge Graph Visualization Examples based on MKR

	Dimensions					Visualization							Description / MKR
	Time	Named Entity	Senti-ment	Topic	Facts	Tile View	Topic Map	Area Chart	Bar Chart	Scatter Chart	Entity Graph	Map Chart	
Time Stamp	✓	✓	✓	✓							✓		Documents are represented as a node collection based on content and included entities. Topic and semantic relationships are inserted into the entity graph as nodes and edges. Sentiment is represented with colors or by modification of edges and/or nodes (size). Furthermore, layout algorithms consider dimensional information to arrange node distances.
Time Span			✓	✓				✓	✓	✓			Sentiment and topic information from selected documents is represented in separate areas, bars or markers within the visualization. The best indicators are sentiment or topic related colors and size (number of documents).
Time Stamp	✓	✓	✓									✓	Sentiment and topic information about selected documents is represented in separate parts (countries) of the map chart. The best indicator is color. Each country in the map is highlighted by its related topic or sentiment color.
Time Stamp	✓	✓	✓				✓						Entities, sentiments or topics are presented inside the topic map. For example, all entities which have a strong relationship to the corresponding topic are visualized in the topic rectangle as smaller rectangles. Rectangles can be further highlighted according to the assigned sentiment, named entity or topic.
Time Stamp			✓	✓		✓							Tiles typically consist of parts from the document's content / a text summarization. Sentiment or topic information is visualized in the tile overview by color modifications.

4.1.1 Data Collection

Dynamic Web Crawling: Due to the individual, dynamic structure of web pages in the WWW, it is hardly possible to know where the actual headline, body, date, author name or other meta data from a published (textual) news is located on a specific website. For example, some pages use simple <p> elements (text paragraphs), others use <div> elements (block of paragraphs and multiple elements, such as images, tables, etc.) to define parts of the HTML document. Elements normally also use class or id attributes in conjunction with Cascading Style Sheets (CSS) to apply a defined layout or style on the web page. Consequently, in order to extract text information from a website it must be a priori known, in which HTML tags or CSS elements the actual message text can be found. Moreover, the news text can also spread over several elements.

In general, there are two possibilities to solve the general web crawling problem. For each news portal, a customized web crawler has to be developed which is specially adapted to the characteristics of the web page, or a generic crawler needs to be used which stores all the elements of a web page, and if available, all the associated information such as CSS identifiers. The disadvantage of the first variant lies in the fact that with an increasingly large number of different news portals also many crawlers or crawler variations would have to be written, updated or even adjusted in case of changes in the structure of the portal or web pages. The second approach is a generic web crawling approach which can be applied on almost every CSS styled web page. The crawling is done based on CSS element selection. Each available class or id attributes from the HTML elements on a web page, formatted with CSS, are used as selection parameter in XML Path Language (XPath) queries. This results in the extraction of different larger meaningful parts from web pages. Of course, a lot of

unnecessary content is extracted in the same way, too. However, the rule-based cleaning and pre-processing of the documents that result from the second crawling approach are easier than the manual customization of different web crawlers depending on the target news portal or even web page level customization.

Crawling Procedure: The crawling algorithm is explained as follows: The crawler first retrieves a summary of lately published news articles from the news portal's RSS feed. Secondly, articles which are missing in the MongoDB database are loaded (if crawling is allowed on the web page at all) with the headless browser PhantomJS in order to fully load the content. Thirdly, web page's metadata (e.g. favicon, time of creation, title) and all contained links are retrieved from the web page's source code by means of using regular expressions. All CSS style identifiers are determined in the next step, whereupon the associated textual content is selected by an XPath query and stored in a temporary list. For the identifiers to be unambiguous, corresponding endings are added in case of multiple occurrences. Finally, all the extracted information is evaluated for length and a general heuristic to detect the main article of a web page. Afterwards it is stored in BSON (Binary JavaScript Object Notation) format in a MongoDB document collection.

4.1.2 Data Pre-Processing

The pre-processing steps which are implemented in the C# prototype are described in the following sections. The tasks which are applied on loaded documents from the MongoDB include language detection, POS tagging and sentence splitting. An own implementation of the pre-processing steps within the prototype has been chosen over state-of-the-art because of customization possi-

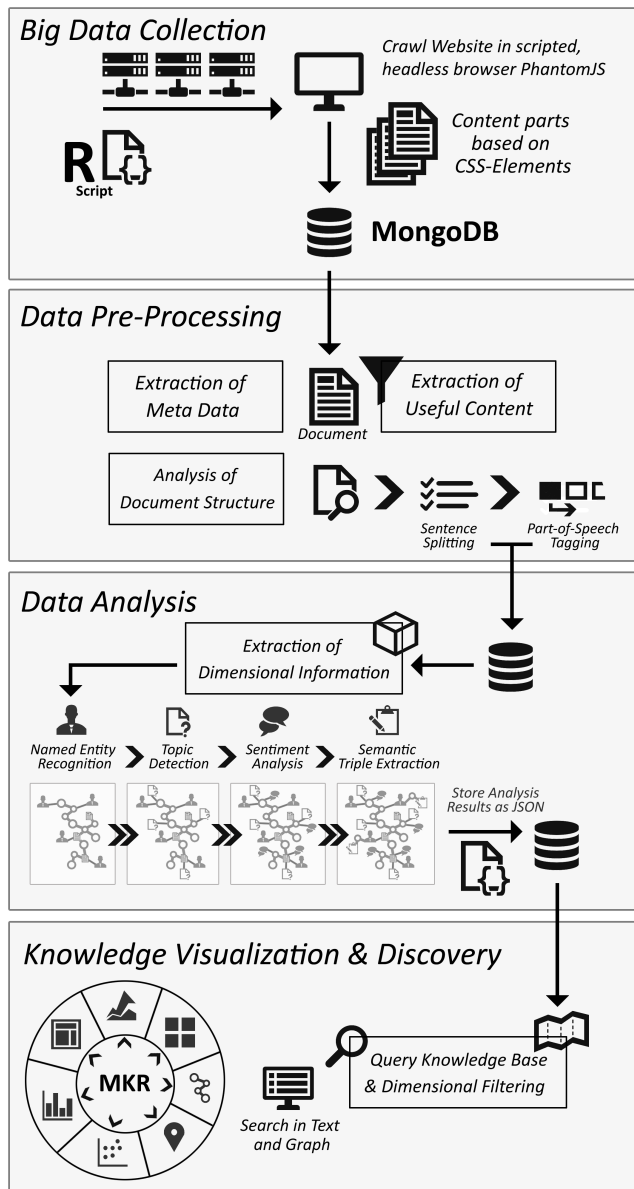


Fig. 3 Implementation Overview

bilities, research purpose and especially for future work adaptation.

Language Detection: The prototype uses a language detection system which analyzes the textual content from MongoDB documents and determines the stop words within the news text. Two different lists of stop words for German and English language are available and contain several language-specific, non meaningful words. Each word from the document is compared to the entries in the stop word lists and the corresponding language with the most matches is finally detected as the document's language. This detection approach works very well with large documents and considerably well with short texts. Documents which contain very small

text fragments or text phrases without using any stop word are labeled for manual user detection.

Sentence Splitting: A logic process has been implemented in the prototype which splits documents into sentences. A rule-based model has been created to split sentences based on a set of regular expressions (e.g. applied on punctuation). Moreover, additional language-specific rule sets are used to avoid wrong sentence splitting in many different occasions.

Part-of-Speech Tagging: The prototype has an implemented POS tagger based on Hidden Markov Model (HMM). The POS tagger uses the STTS tagset [59] for German language and the Penn Treebank tagset [60] for English language. Both implemented POS taggers are using a separate corpus (training set) with POS annotated sentences and around 1 million words. Also the viterbi algorithm [61] is applied in both taggers.

4.1.3 Data Analysis

In the data analysis step, pre-processed documents are analyzed by the introduced text mining methods, in order to extract dimensional information for the MKR structure. Therefore, the data analysis in the implementation follows the steps of the MKR process which have been explained in Section 3.1 and visualized in Fig. 1. The implementation details of named entity recognition, sentiment analysis, topic detection and semantic triple extraction are given in the following sections.

Named Entity Recognition: Named entities are firstly recognized within documents based on the POS tags in splitted sentences. The POS tag sequences provided from pre-processing are further analyzed in a second step to combine coherent words, which all have been tagged as named entity into one entity. Afterwards, a lexical resource and reference to German or English Linked Open Data is used to evaluate and distinguish potential suggested named entities. The resources which are language-dependent contain the named entity categories date, location, organization, person and miscellaneous. Identified named entities are placed into the categories by a classification approach. Within the application, a blacklisting of wrongly tagged named entities is also possible to improve the quality of the named entity collection. Each document's entities can be selectively edited, updated or deleted for manual adjustments.

Sentiment Analysis: Sentiment analysis has been implemented in the application to identify words and n-grams which express a positive or negative polarity. Adjectives and adverbs are recognized based on POS tag

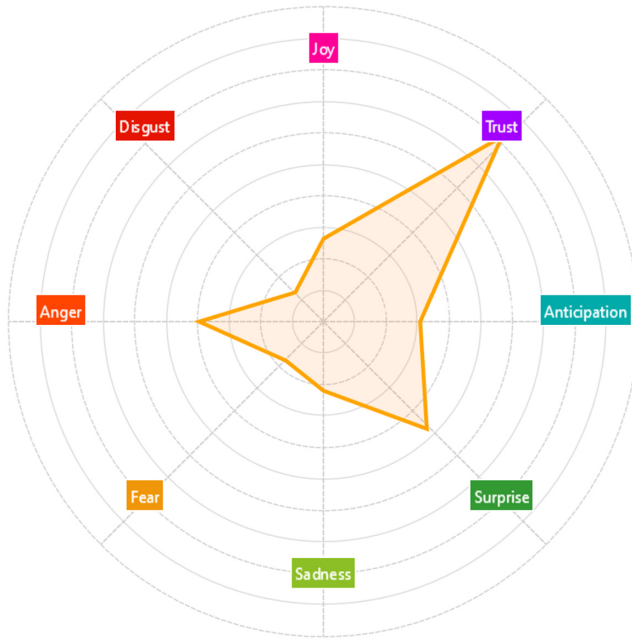


Fig. 4 Sentiment Evaluation based on Document's Emotion Classification (Normalized Scale from 0 to 1)

information and language-specific lexical resources are used to evaluate them. In the English language, polarity shifters (“no”, “not” or the ending “n’t”) before adjectives or adverbs, are recognized within sentences and the corresponding sentiment values are turned around for negation handling. In the German language the token “nicht” is another typical example for a polarity shifter. Sentiment intensifiers are used in the prototype’s sentiment analysis to increase the sentiment value of several words or n-grams in the positive or negative sense. Typical examples for the English language are “most”, “many”, “biggest”, “much”, “more”, “major”, “all”, “very”, “big”, “strong”, “every”, “truly”, “slightly”, “overwhelming” or “increasingly”. The evaluation of the document’s sentiment is based on the German corpus SentiWS [62] and the English corpus NRC Emotion Lexicon [63]. Both corpora contain terms with negative and positive evaluation on a scale from -1 to +1. Sentiment detection is also done on more detailed level. Sentences which contain named entity information are further analyzed and a relationship between n-grams from the text and the entity is stored in MKR.

In addition to the regular sentiment analysis, an emotion classifier has been added to the implementation in order to further distinguish the sentiment evaluation into their specific emotions. Eight categories are used for the emotions, namely joy, sadness, anxiety, fear, trust, disgust, surprise and anticipation. Fig. 4 visualizes the emotion classification from an example document.

Topic Detection: For the detection of topics two language-specific lexical resources were created, which contain words and n-grams on different subject areas. For the German language the corpus is based on Dornseiff, a german thesaurus arranged by subject groups [64]. Topics and subtopics have been extracted in the scope of this research. An English version of the topic corpus has been created manually from the German corpus to also cover translated topics and subtopics. In the implementation, a document is assigned to a topic and subtopic area according to the word occurrences and correspondences of the n-grams with the annotated corpora. The classification process based on n-gram occurrence works very well for longer documents. In the case of shorter documents or documents which frequently change the topic focus, the implemented topic recognition is susceptible to errors. To eliminate the errors, a function has been implemented which provides the user with a notification for disambiguation on frequent occurrence of words from several topic areas. The list of topics based on [64] contains “nature and environment” with 25 subtopics, “life” (43), “room, position, shape” (46) “size, quantity, number” (52), “entity, relationship, event” (47), “time” (35), “visibility, light, color, sound, temperature, weight, aggregate conditions” (68), “location and location change” (46), “will and action” (83), “feeling, affects, character traits” (60), “thinking” (56), “sign, message, speech” (63), “science” (27), “arts and culture” (24), “human living” (80), “food and drink” (22), “sport and leisure” (28), “society” (33), “devices and technology” (27), “economy and finance” (50), “law and ethics” (35), “religion and spiritual” (20) and “other” topics. A dictionary with translations of topics and subtopics from English to German and vice versa has been created and is used by the application to sort the analysis results into a uniform topic structure.

Semantic Triple Extraction: For the semantic triple extraction, the approach from Akbik and Broß (see 2.4.4) is used in the application on the pre-processed sentences. A chunker has been implemented to assist the POS tagger in order to identify noun phrases (NP), proper noun phrases (PNP) and verb phrases (VP). For each sentence, it is checked whether two NP or PNP are contained. The results from named entity recognition are used in this step. If there are two such phrases, rules determine whether they are extracted as subject and object with an associated predicate.

The details of the extraction process are described as follows: First of all, cases in which NP or PNP pairs have been recognized in subordinate clause are removed from the analysis set. The extraction is also interrupted for certain punctuation marks between the phrases. Anal-

ogously, this is the case with text sections in brackets. Next, there must be at least one verb between the potential subject and the object, and a maximum of five words should be between these two, so that a direct semantic context is likely. If a NP is followed by a PNP, this is chosen instead of the NP, for a maximum of five words. This number has been found to be satisfactory during the development since larger distances sometimes led to incorrect associations and smaller distances to missing information. If these decision criteria are satisfied, a predicate is formed from the words between the subject and the object, in which, all the articles are omitted. Another rule-based filter which is similar to the one presented in Akbik and Broß [55] is applied in the last step on the created predicate to check whether it represents a semantic relationship (e.g. relation is "is-a" type).

4.2 Representation

After the text mining methods have been applied and the results have been collected, the MKR is created by the implementation and inserted in the knowledge base. Therefore, the MKR is saved in the MongoDB by utilizing a JSON format. Listing 1 describes the MKR structure for a referenced document in the knowledge base, while all the mentioned relationships are shown by way of example. A time stamp is created with all relationships in order to use the time as another analysis dimension and necessary information for different visualizations.

In the structure depicted in Listing 1, the results from previously mentioned sentiment analysis, topic detection, entity recognition and semantic relation are shown. Sentiment n-grams are provided with an ObjectId for further analysis of relationships between documents and entities with similar polarity. The relation between the sentiment n-gram and entity information (both referenced by ObjectId) is further described with additional details about intensification or sentiment shifting in negation cases. The (multi-) topic relation is represented by referenced topic names and subtopic names. In the entity relation part, named entities from the document are listed with their name, type, document language and their created timestamp for temporal analysis and potential analysis of entity occurrence over time. Semantic relationships between entities of the document are represented in the last part of MKR representation in Listing 1. Here, the subject, predicate and object of one RDF triple is stored in the knowledge base together with the timestamp and related Ids for further analysis of similarity and correspondence of entity references.

```
{
  "_id" : ObjectId(...),
  "created" : ISODate(...),
  "_documentId" : ObjectId(...),
  "language" : "en-EN",
  "sentimentRelation" : [
    {
      "_id" : ObjectId(...),
      "language" : "en-EN",
      "nGram" : "attack",
      "value" : -0.847,
      "created" : ISODate(...)
    },
    { ... }
  ],
  "documentSentimentValue" : -0.0767,
  "documentSentimentEmotion" : {
    "Anger" : 0.00,
    "Anticipation" : 0.60,
    "Disgust" : 0.20,
    "Fear" : 1.000,
    "Joy" : 0.20,
    "Sadness" : 0.60,
    "Surprise" : 0.800,
    "Trust" : 0.00
  },
  "sentimentEntityRelation" : [
    {
      "_entityId" : ObjectId(...),
      "_sentimentNgramId" : ObjectId(...),
      "language" : "en-EN",
      "IsShifted" : true,
      "Shifter" : [
        "not"
      ],
      "IsIntensified" : true,
      "Intensifier" : [
        "many"
      ]
    },
    { ... }
  ],
  "topicRelation" : [
    {
      "language" : "en-EN",
      "topic" : "Law and ethics",
      "subtopic" : "judge, lawyer"
    },
    { ... }
  ],
  "entityRelation" : [
    {
      "_id" : ObjectId(...),
      "language" : "en-EN",
      "entityName" : "U.S. Supreme Court",
      "entityType" : "Miscellaneous",
      "created" : ISODate(...)
    },
    { ... }
  ],
  "semanticRelation" : [
    {
      "_subjectId" : ObjectId(...),
      "language" : "en-EN",
      "subject" : "Donald Trump",
      "predicate" : "is-a",
      "_objectId" : ObjectId(...),
      "object" : "U.S. President",
      "created" : ISODate(...)
    },
    { ... }
  ]
}
```

Listing 1 JSON Format of MKR (Example Analysis Result of one Document)

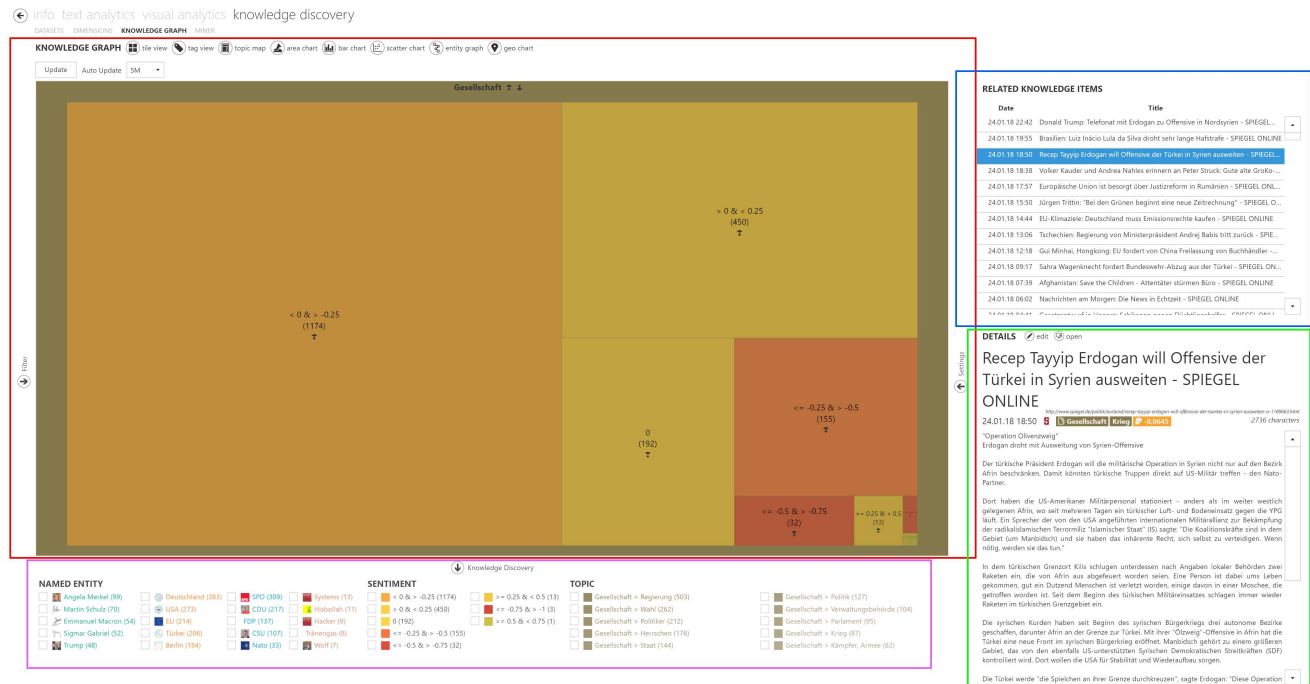


Fig. 5 Knowledge Discovery Components: Knowledge Graph (red), Related Knowledge Items (blue), Details of Selected Knowledge Item (green) and MKR Results from Named Entity, Sentiment, Topic Dimension in Preview (purple). The Screenshot of the prototype shows a subset of SPIEGEL Online articles related to the topic “society”, organized in a topic map visualization using the document’s sentiment dimension as layout arrangement.

4.3 Knowledge Discovery

MKR is especially important for graphical search and the knowledge discovery process in knowledge-based systems. Through the integration of different analysis results from text mining methods and their availability (pre-calculated results), MKR empowers different kinds of knowledge graphs to visualize a multidimensional perspective. Importantly, further analysis results from other dimensions can be integrated in the knowledge visualization by request. MKR operations and transformations, namely the selection and filtering of dimensions, adapt visualizations for specific queries. Moreover, knowledge graphs can be even exchanged by other visualizations and still integrate the same dimensions in the graph.

Fig. 5 shows different components which can be used in the knowledge discovery process. The different types of knowledge graphs (selectable in red highlighted area) provided by the prototype support the user by the search and visualization of desired information. The preview of further (not necessarily expected or desired) results from MKR besides the current visualization (purple) contextualize the information and indicate results from other dimensions or further possibilities for dimensional filtering or selection. If such a filter is selected, the query on the knowledge base is automatically adjusted

and potential unexpected results are provided. In this way, the knowledge discovery process is exploratory and different information visualizations can be transformed into each other in order to adapt the knowledge graph to show suggested or desired results. The results from the knowledge discovery are always provided as list of knowledge items (e.g. documents) in the prototype and also single documents can be selected to read details, modify or export them in order to support the user’s knowledge discovery process or information search.

4.4 Summary

The data set which contains articles from various news domains in German and English language has been analyzed by the presented text mining methods. The results have been integrated in the presented MKR structure for knowledge discovery and information visualization purpose. Each of the 305,281 articles has been analyzed for topics (and related subtopics), sentiment on different levels, named entities (with types person, location, organization and miscellaneous) and semantic relationships in RDF triple format.

The results show an average of 9.45 extracted named entities per document. The most assigned topics are “human living” and “will and action”. According to the topic detection analysis results, the corpus contains the

lowest amounts of documents which are related to “science” and “food and drink”. A total amount of 3131 semantic relationships have been extracted from the documents by further analysis of the named entities. The average sentiment from the overall analysis is slightly negative with a value of -0.002. Documents assigned to the topics “religion and spiritual” have the most positive sentiment values in average (0.064), whereas “law and ethics” related documents the most negative sentiment values (-0.072).

5 Conclusion and Future Work

MKR has been presented in this research as a knowledge representation method based on integrative text mining results. MKR enables knowledge graphs to visualize multidimensional perspectives on analysis results by transformation operations namely dimensional selection and filtering. Through a dynamic adjustment of the knowledge graphs, further analysis results from other knowledge dimensions are integrated in the visualization by request without any additional analysis or calculation. This is considered advantageous compared to traditional knowledge representation (e.g. the combination of RDF, XML and OWL, text databases or other knowledge base representation types such as frames or rules), and state-of-the-art visualization methods which are focused only on selected analysis dimensions. In this research, a data set of German and English news articles has been analyzed by an implementation which uses the presented MKR structure in order to integrate different text mining methods in the analysis and visualization. Therefore, the methods named entity recognition, topic detection, sentiment analysis and semantic triple extraction have been implemented in the application and used for knowledge graph visualization.

In future work, the MKR framework will be used as a basis for knowledge discovery processes and intelligent selection mechanism for extractive and abstractive text summarization, computer-aided writing and text generation. The authors are currently working on an extension of the prototype towards this direction, which is used for text summarization based on dimensional selection and filtering. In this way, a knowledge-intensive search process is supported which is able to retrieve and summarize relevant information in the knowledge base, already filtered by other dimensions (e.g. an extractive text summarization of all relevant sources related to entity e in topic t above entity-level sentiment value s in a selected timeframe between t_x and t_y).

References

1. T. Berners-Lee, J. Hendler, O. Lassila *et al.*, “The semantic web,” *Scientific american*, vol. 284, no. 5, pp. 28–37, 2001.
2. J. Zenkert and M. Fathi, “Multidimensional knowledge representation of text analytics results in knowledge bases,” in *Electro Information Technology (EIT), 2016 IEEE International Conference on*. IEEE, 2016, pp. 0541–0546.
3. K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor, “Freebase: a collaboratively created graph database for structuring human knowledge,” in *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*. AcM, 2008, pp. 1247–1250.
4. M. Fabian, K. Gjergji, W. Gerhard *et al.*, “Yago: A core of semantic knowledge unifying wordnet and wikipedia,” in *16th International World Wide Web Conference, WWW*, 2007, pp. 697–706.
5. A. Simitsis, A. Baid, Y. Sismanis, and B. Reinwald, “Multidimensional content exploration,” *Proceedings of the VLDB Endowment*, vol. 1, no. 1, pp. 660–671, 2008.
6. D. Zhang, “Integrative text mining and management in multidimensional text databases,” Ph.D. dissertation, University of Illinois at Urbana-Champaign, 2013.
7. C. X. Lin, B. Ding, J. Han, F. Zhu, and B. Zhao, “Text cube: Computing ir measures for multidimensional text database analysis,” in *Data Mining, 2008. ICDM’08. Eighth IEEE International Conference on*. IEEE, 2008, pp. 905–910.
8. O. Lassila and R. R. Swick, “Resource description framework (rdf) model and syntax specification,” 1999.
9. D. L. McGuinness, F. Van Harmelen *et al.*, “Owl web ontology language overview,” *W3C recommendation*, vol. 10, no. 10, p. 2004, 2004.
10. P. Hitzler, M. Krötzsch, S. Rudolph, and Y. Sure, *Semantic Web: Grundlagen*. Springer-Verlag, 2007.
11. J. Broekstra, M. Klein, S. Decker, D. Fensel, F. Van Harmelen, and I. Horrocks, “Enabling knowledge representation on the web by extending rdf schema,” *Computer networks*, vol. 39, no. 5, pp. 609–634, 2002.
12. P. Michalik, J. Stofa, and I. Zolotova, “Concept definition for big data architecture in the education system,” in *Applied Machine Intelligence and Informatics (SAMi), 2014 IEEE 12th International Symposium on*. IEEE, 2014, pp. 331–334.
13. M. Marjani, F. Nasaruddin, A. Gani, A. Karim, I. Hashem, A. Siddiqua, and I. Yaqoob, “Big iot data analytics: Architecture, opportunities, and open research challenges,” *IEEE Access*, 2017.
14. M. Bohlouli, J. Dalter, M. Dornhöfer, J. Zenkert, and M. Fathi, “Knowledge discovery from social media using big data-provided sentiment analysis (somabit),” *Journal of Information Science*, vol. 41, no. 6, pp. 779–798, 2015.
15. C. P. Chen and C.-Y. Zhang, “Data-intensive applications, challenges, techniques and technologies: A survey on big data,” *Information Sciences*, vol. 275, pp. 314–347, 2014.
16. D. Fasel and A. Meier, *Big Data: Grundlagen, Systeme und Nutzungspotenziale*. Springer-Verlag, 2016.
17. A. Gadatsch and H. Landrock, “Zielsetzung von big-data-projekten,” in *Big Data für Entscheider*. Springer, 2017, pp. 11–16.
18. S. V. Nandury and B. A. Begum, “Strategies to handle big data for traffic management in smart cities,” in *Advances in Computing, Communications and Informatics*

- (ICACCI), 2016 International Conference on. IEEE, 2016, pp. 356–364.
19. U. Bohnacker, L. Dehning, J. Franke, and I. Renz, “Textual analysis of customer statements for quality control and help desk support,” in *Classification, Clustering, and Data Analysis*. Springer, 2002, pp. 437–445.
 20. D. Abts and W. Mülder, *Grundkurs Wirtschaftsinformatik: eine kompakte und praxisorientierte Einführung*. Springer-Verlag, 2009.
 21. A. Hotho, A. Nürnberger, and G. Paaß, “A brief survey of text mining,” in *Ldv Forum*, vol. 20, no. 1, 2005, pp. 19–62.
 22. G. Salton, A. Wong, and C.-S. Yang, “A vector space model for automatic indexing,” *Communications of the ACM*, vol. 18, no. 11, pp. 613–620, 1975.
 23. Y. Wilks, “Information extraction as a core language technology,” *Information Extraction A Multidisciplinary Approach to an Emerging Information Technology*, pp. 1–9, 1997.
 24. Y. Kodratoff, “Knowledge discovery in texts: a definition, and applications,” *Foundations of Intelligent Systems*, pp. 16–29, 1999.
 25. D. Lin and X. Wu, “Phrase clustering for discriminative learning,” in *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*. Association for Computational Linguistics, 2009, pp. 1030–1038.
 26. D. Balasuriya, N. Ringland, J. Nothman, T. Murphy, and J. R. Curran, “Named entity recognition in wikipedia,” in *Proceedings of the 2009 Workshop on The People’s Web Meets NLP: Collaboratively Constructed Semantic Resources*. Association for Computational Linguistics, 2009, pp. 10–18.
 27. D. Nadeau and S. Sekine, “A survey of named entity recognition and classification,” *Linguisticae Investigationes*, vol. 30, no. 1, pp. 3–26, 2007.
 28. D. M. Bikel, S. Miller, R. Schwartz, and R. Weischedel, “Nymble: a high-performance learning name-finder,” in *Proceedings of the fifth conference on Applied natural language processing*. Association for Computational Linguistics, 1997, pp. 194–201.
 29. S. Sekine *et al.*, “Nyu: Description of the japanese ne system used for met-2,” in *Proc. Message Understanding Conference*, 1998.
 30. M. Asahara and Y. Matsumoto, “Japanese named entity extraction with redundant morphological analysis,” in *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*. Association for Computational Linguistics, 2003, pp. 8–15.
 31. A. Borthwick, J. Sterling, E. Agichtein, and R. Grishman, “Nyu: Description of the mene named entity system as used in muc-7,” in *In Proceedings of the Seventh Message Understanding Conference (MUC-7)*. Citeseer, 1998.
 32. A. McCallum and W. Li, “Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons,” in *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*. Association for Computational Linguistics, 2003, pp. 188–191.
 33. D. Downey, M. Broadhead, and O. Etzioni, “Locating complex named entities in web text,” in *IJCAI*, vol. 7, 2007, pp. 2733–2739.
 34. D. D. Palmer and D. S. Day, “A statistical profile of the named entity task,” in *Proceedings of the fifth conference on Applied natural language processing*. Association for Computational Linguistics, 1997, pp. 190–193.
 35. A. Mikheev, M. Moens, and C. Grover, “Named entity recognition without gazetteers,” in *Proceedings of the ninth conference on European chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, 1999, pp. 1–8.
 36. J. Jiang, “Information extraction from text,” in *Mining text data*. Springer, 2012, pp. 11–41.
 37. L. A. Ramshaw and M. P. Marcus, “Text chunking using transformation-based learning,” in *Natural language processing using very large corpora*. Springer, 1999, pp. 157–176.
 38. C. D. Manning, P. Raghavan, H. Schütze *et al.*, *Introduction to information retrieval*. Cambridge university press Cambridge, 2008, vol. 1, no. 1.
 39. D. M. Blei, A. Y. Ng, and M. I. Jordan, “Latent dirichlet allocation,” *Journal of machine Learning research*, vol. 3, no. Jan, pp. 993–1022, 2003.
 40. T. K. Landauer and S. T. Dumais, “A solution to plato’s problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge,” *Psychological review*, vol. 104, no. 2, p. 211, 1997.
 41. T. Hofmann, “Probabilistic latent semantic analysis,” in *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 1999, pp. 289–296.
 42. R. Baeza-Yates, B. Ribeiro-Neto *et al.*, *Modern information retrieval*. ACM press New York, 1999, vol. 463.
 43. R. Kuhlen, T. Seeger, and D. Strauch, *Grundlagen der praktischen Information und Dokumentation: Band 1: Handbuch zur Einführung in die Informationswissenschaft und -praxis, Band 2: Glossar*. Walter de Gruyter, 2004.
 44. A. Klahold, P. Uhr, F. Ansari, and M. Fathi, “Using word association to detect multitopic structures in text documents,” *IEEE Intelligent Systems*, vol. 29, no. 5, pp. 40–46, 2014.
 45. P. Uhr, A. Klahold, and M. Fathi, “Imitation of the human ability of word association,” *International Journal of Soft Computing and Software Engineering (JSCSE)*, vol. 3, no. 3, 2013.
 46. W. Medhat, A. Hassan, and H. Korashy, “Sentiment analysis algorithms and applications: A survey,” *Ain Shams Engineering Journal*, vol. 5, no. 4, pp. 1093–1113, 2014.
 47. E. Cambria, B. Schuller, Y. Xia, and C. Havasi, “New avenues in opinion mining and sentiment analysis,” *IEEE Intelligent Systems*, vol. 28, no. 2, pp. 15–21, 2013.
 48. B. Pang, L. Lee *et al.*, “Opinion mining and sentiment analysis,” *Foundations and Trends® in Information Retrieval*, vol. 2, no. 1–2, pp. 1–135, 2008.
 49. A. Kaur and V. Gupta, “A survey on sentiment analysis and opinion mining techniques,” *Journal of Emerging Technologies in Web Intelligence*, vol. 5, no. 4, pp. 367–371, 2013.
 50. K. Schouten and F. Frasincar, “Survey on aspect-level sentiment analysis,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 3, pp. 813–830, 2016.
 51. B. Liu, “Sentiment analysis and opinion mining,” *Synthesis lectures on human language technologies*, vol. 5, no. 1, pp. 1–167, 2012.
 52. J. Broß, “Aspect-oriented sentiment analysis of customer reviews using distant supervision techniques,” Ph.D. dissertation, Freie Universität Berlin, 2013.
 53. R. Varghese and M. Jayasree, “A survey on sentiment analysis and opinion mining,” *IJRET: International Journal of Research in Engineering and Technology eISSN*, vol. 23191163, 2013.

54. P. Uhr, J. Zenkert, and M. Fathi, "Sentiment analysis in financial markets a framework to utilize the human ability of word association for analyzing stock market news reports," in *Systems, Man and Cybernetics (SMC), 2014 IEEE International Conference on*. IEEE, 2014, pp. 912–917.
55. A. Akbik and J. Broß, "Wanderlust: Extracting semantic relations from natural language text using dependency grammar patterns," in *WWW Workshop*, 2009.
56. D. D. Sleator and D. Temperley, "Parsing english with a link grammar," *arXiv preprint cmp-lg/9508004*, 1995.
57. U. M. Fayyad, A. Wierse, and G. G. Grinstein, *Information visualization in data mining and knowledge discovery*. Morgan Kaufmann, 2002.
58. F. Stahl, B. Gabrys, M. M. Gaber, and M. Berendsen, "An overview of interactive visual data mining techniques for knowledge discovery," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 3, no. 4, pp. 239–256, 2013. [Online]. Available: <http://dx.doi.org/10.1002/widm.1093>
59. A. Schiller, S. Teufel, and C. Thielen, "Guidelines für das tagging deutscher textcorpora mit stts," *Universitäten Stuttgart und Tübingen*, 1995.
60. M. P. Marcus, M. A. Marcinkiewicz, and B. Santorini, "Building a large annotated corpus of english: The penn treebank," *Computational linguistics*, vol. 19, no. 2, pp. 313–330, 1993.
61. G. D. Forney, "The viterbi algorithm," *Proceedings of the IEEE*, vol. 61, no. 3, pp. 268–278, 1973.
62. R. Remus, U. Quasthoff, and G. Heyer, "Sentiws-a publicly available german-language resource for sentiment analysis," in *LREC*, 2010.
63. S. M. Mohammad and P. D. Turney, "Crowdsourcing a word-emotion association lexicon," vol. 29, no. 3, pp. 436–465, 2013.
64. F. Dornseiff, *Der deutsche Wortschatz nach Sachgruppen*. de Gruyter, 2004.