

# Supervised Sentiment Analysis of Parliamentary Speeches and News Reports

Elena Rudkowsky<sup>1</sup>, Martin Haselmayer<sup>2</sup>, Matthias Wastian<sup>3</sup>, Marcelo Jenny<sup>4</sup>, Štefan Emrich<sup>3</sup>,  
and Michael Sedlmair<sup>1</sup>

<sup>1</sup>*University of Vienna (Faculty of Computer Science)*

<sup>2</sup>*University of Vienna (Department of Government)*

<sup>3</sup>*Drahtwarenhandlung (dwh) GmbH, Vienna*

<sup>4</sup>*University of Innsbruck (Department of Political Science)*

**Conference Paper:** 67th Annual Conference of the International Communication Association (ICA), Panel on ‘Automatic Sentiment Analysis’, 25-29 May 2017

**Funding:** This research was partially funded by FFG project 845898 (VALiD: Visual Analytics in Data-driven Journalism) and the Hochschuljubiläumstiftung der Stadt Wien (H-304565/2015).

**Acknowledgements:** We thank Elisabeth Graf, Lisa Hirsch, Christoph Kralj, Michael Oppermann and Johanna Schlereth for their research assistance.

## Abstract

In this paper, we use several supervised machine learning approaches and compare their success in predicting the sentiment of Austrian parliamentary speeches and news reports (German language). Prediction results in learning-based sentiment analysis vary strongly. They depend on the choice of algorithm and its parameterization, the quality and quantity of available training data as well as the selection of appropriate input feature representations. Our training data contains human-annotated sentiment scores at the phrase and sentence level. Going beyond the dominant bag-of-words modeling approach in traditional natural language processing, we also test sentiment analysis for neural network-based distributed representations of words. The latter reflect syntactic as well as semantic relatedness, but require huge amounts of training examples. We test both approaches with heterogeneous textual data, compare their success rates and provide conclusions on how to improve the sentiment analysis of political communication.

## 1 Introduction

Sentiment analysis is a major area of interest within the field of data-driven journalism [35, 40]. While a variety of definitions for data-driven journalism exist, we refer to “gathering, cleaning, organizing, analyzing, visualizing, and publishing data to support the creation of acts of journalism” [16]. When sentiment is analyzed for publishing purposes, the concept of negativity generally is crucial for a story to be news-worthy [23]. Hate speeches, shitstorms and political incivility are concepts that easily attract people’s attention. Incivility of political communication [2, 14] is often seen as a cause of increasing political polarization, lower electoral turnout and voter disaffection with politics and democracy in general [5, 17, 18, 20, 27]. However, there is no agreement on the definition or measurement of incivility. In this paper, we describe a framework for measuring negative and uncivil language based on a supervised document classification approach. Although we focus on negativity, the process can also be applied to a sentiment scale ranging from positive to negative statements. The focus of this work is the introduction of a novel feature representation for document classification purposes to the social sciences. This model has several advantages compared to the common bag-of-words modeling approach. The method illustrated in this paper is not restricted to be applied to sentiment prediction. Other document classification approaches can be realized as well.

The main contribution of this paper is the introduction of a supervised document classification method to the social sciences. We demonstrate this method by predicting sentiment of parliamentary speeches. This method relies on distributed word embeddings [21, 25] to model sentences as fixed-size feature vectors. Distributed word embeddings (also known as word2vec) outperform traditional bag-of-words modeling approaches in many analysis and prediction tasks since they inherently reflect syntactic and semantic relationships between words. To predict sentiment scores for all modeled distributed sentence embeddings we use a neural network-based classifier. The resulting sentence-level scores are eventually aggregated to reflect emotions on different scales. Such scales cover individual speeches, authors, topics and time frames.

We show the effectiveness of our proposed method with a case study on the determination of negativity in the Austrian national parliament. Covering 56,000 German language parliamentary speeches from 1996 to 2013, we explore different political, institutional and individual factors that affect the level of negativity. Our findings are validated against hypotheses on negativity in parliament predefined from political scientists. The factors we investigate include the status (opposition/government) and the role (MP, cabinet, party lead) of politicians, which both have a noticeable effect on the level of negativity observed in parliamentary speeches. We also take a closer look at the most negative topics discussed in the Austrian national parliament.

The final results of our case study are published as web-based data story in German and English. Data stories serve to communicate results from long-term, data-intense investigations to the public. Most often this happens in a visual or interactive manner [22, 36, 38]. An extensive report on “The Art and Science of Data-driven Journalism” [16] includes a section describing the need for “New Tools to Wrangle Unstructured Data”. We address this need by designing a system that optimally fits into the data-journalistic tool chain, since sentiment analysis is of high relevance for journalists who work with unstructured data [40].

## 2 Method: Supervised Sentiment Analysis with Word Embeddings

In this section, we introduce our approach for supervised sentiment analysis of documents. The documents that represent our ‘data of interest’ are political speeches in German from Austria. We furthermore call them application data (see topmost database in Figure 1). The basic idea for our system is to predict a negativity score for each individual sentence within the application dataset (done by the classifier unit in Figure 1). These negativity scores can eventually be aggregated to higher level scores to predict negativity for entire speeches, certain politicians or whole parties. All available structured (meta) data, complementing the application dataset, is helpful for giving insights at the final aggregation stage. Such meta data can range from chronological to location- or source-dependent information information about the document corpus.

Since our method is based on supervised machine learning the core of this process is to build a classification model that reliably predicts sentiment scores. Towards that goal, at the beginning some ‘ground truth’ data is needed that can be used to train such a classifier. Our training data comprises politics-related sentences from political media releases and parliamentary speeches. This data is labeled by human annotators from a crowd-coding platform. Each individual sentence is labeled by at least 10 different participants on a scale from 0 (not negative) to 4 (very negative). As both our training and application data stems from real world texts of different sources they comprise a certain degree of messiness [19]. The first important step of working such datasets is to preprocess and clean them. After the application and training datasets have been preprocessed the single ‘data units of interest’ (in our case: individual sentences) need to be transformed into fixed-sized feature vectors to make them processable by the classifier.

In traditional natural language processing, documents are often transformed into a so-called bag-of-words. These ‘one-hot’ representations treat different words as different features of a vector. The length of this vector is equal to the size of the vocabulary within the considered corpus. The appearance of a word within a document results in incrementing one particular feature within the corresponding document vector. This modeling approach is sufficient for analytic tasks that, for instance, focus on the amount of different/shared words between documents (e.g. term frequency comparison or topic modeling [3]). A task like sentiment prediction that is tightly intertwined with natural language construction either needs tremendous amounts of training data or additional, complex, dictionary-based logic to be fulfilled with this basic model. This is due to the fact that bag-of-words representations like TF-IDF [34] do not take the ordering of words into account and therefore are not able to reflect syntactic or semantic relationships. We introduce a feature representation for supervised sentiment classification that builds on pre-trained distributed word embeddings (section 2.3 gives more background information on this modeling approach).

After all sentences have been transformed according to their distributed word embeddings a classification (or regression) model can process them. As soon as a model with high prediction accuracy has been found (trained with and tested against the training data), it can be used to predict negativity scores of the so far unseen application data. Figure 1 gives a technical overview of our proposed method and illustrates the main steps. The upper half of the architecture shows the three different databases which serve as basis for this supervised learning approach (application data, training data, transformation data). Below of these databases the processing units are visualized in their order of execution. The supervised learning phase is color-coded with blue arrows. The classification (or prediction) phase is marked green. At the left and at the bottom of the methodological pipeline all processing units are incorporated into the corresponding units of the Knowledge Discovery in Databases (KDD) process [9].

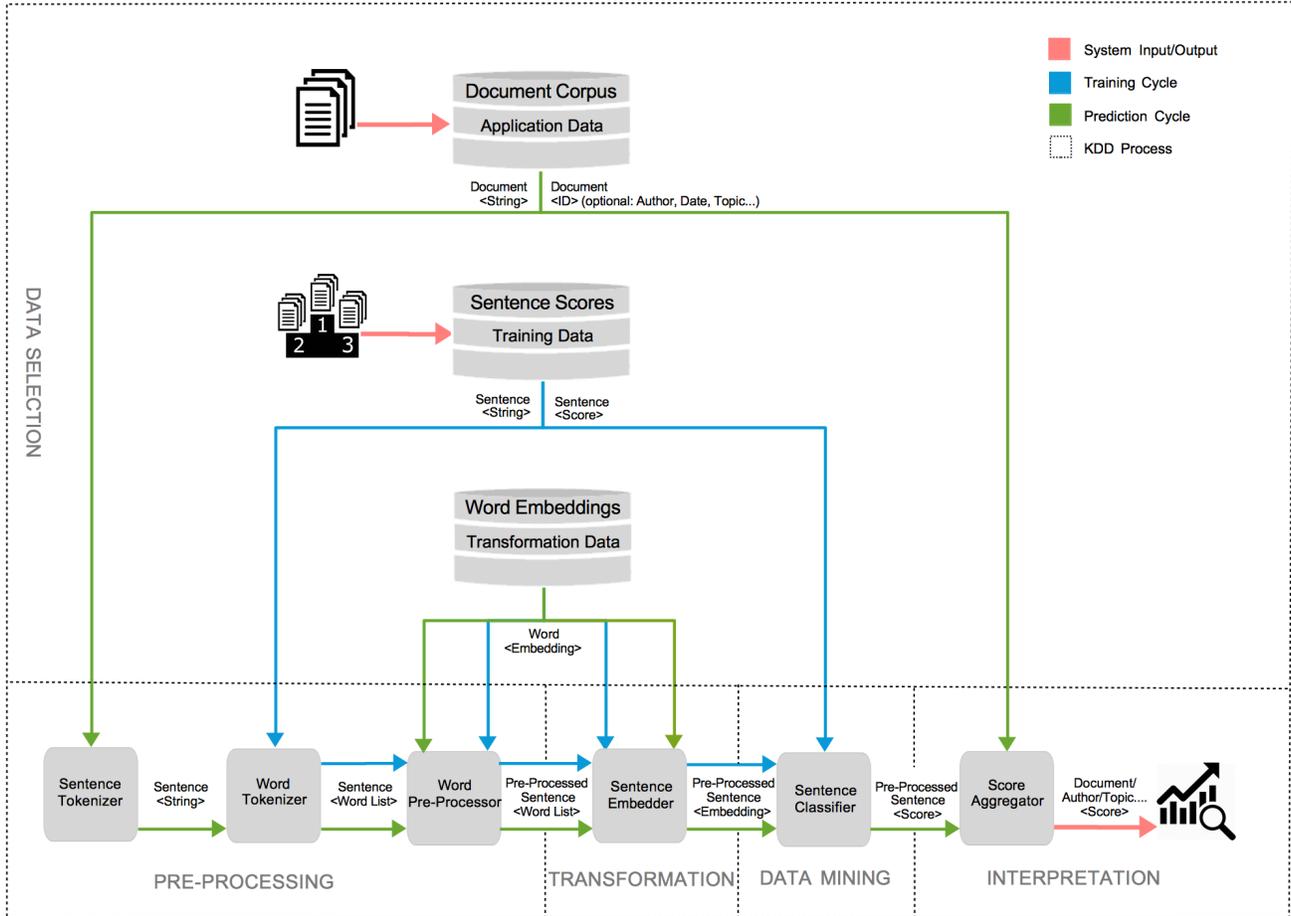


Figure 1: System overview for supervised sentiment analysis based on distributed word embeddings

The single units of Figure 1 are explained in more detail within the following sections. We start with introducing the different data sources (application, training and transformation data) that are needed to apply this method. We explain the characteristics of the particular datasets we used for each unit. The orange arrows that point to databases within Figure 1 indicate sources that are highly dependent on the use case. These application-dependent data sources need to be collected and integrated by the end user of an equivalent system.

## 2.1 Document Corpus (Application Data)

For our use case we have created a database with about 56,000 speeches from the Austrian national parliament. This data was scraped from the official web page of the national parliament [32]. This is our ‘data of interest’ for which we want the system to return sentiment (in our case negativity) scores. The speeches we use for our analyses cover the legislative periods from 1996 to 2013 and sum up to a total of 2.4 million sentences. All speeches are in German. They cover speeches from 578 politicians and provide multiple additional (meta) data fields. This meta data contains fields like name/gender/date of birth of politicians, party membership, and the date of a speech.

These supplementary data can later-on be used to evaluate and interpret the sentiment scores and build custom visualizations around them. For other kinds of use cases the application data could for instance comprise social media comments from Facebook and Twitter or product reviews from business applications.

## 2.2 Sentence Scores (Training Data)

Our training dataset consists of about 20,000 German sentences. Figure 1 shows the origins of these sentences covering different politics-relevant sources. Each sentence is labeled by at least ten different German-speaking coders from the crowd-coding platform CrowdFlower. The sentences are rated on a 5-point negativity scale ranging from 0 (not negative) to 4 (very negative). The sixth possible rating is ‘uncodable’ and leads to the exclusion of that sentence instance from further analyses.

Individual coder performance is monitored during the coding process to identify cheating or spamming. Each participant has to answer four test questions correctly before the actual coding task starts. In addition, one out of five sentences presented during the coding task is actually a test item. Since sentiment coding on a five-point ordinal scale is a difficult task [15, 30], we accepted two adjacent options on the five-point scale as correct answers for all test items. These adjacent options were defined by reference codings established by some of the authors. The probability of passing the entry test by guessing 3 out of 4 test items correctly is only 4%. Including the ‘uncodable’ answer a coder has six options of which two are accepted as correct:  $(\frac{2}{6})^3 = 0.04$ . This probability gets smaller with each additional test item.

Source	Time period	N	%
Party press releases	1995-2015	14,242	70
Parliamentary debates	1995-2013	3,923	19
Media reports	2013	2,327	11
Total		20,593	100

Table 1: Training data of crowd-coded sentences

## 2.3 Word Embeddings (Transformation Data)

Previous work for supervised sentiment prediction in the social sciences heavily relies on the bag-of-words (respectively the TF-IDF) feature representation. A publication of the year 2016 from Mozetič et al. compares more than 15 different sentiment classification systems for Twitter data within its related work section. The section finishes with the statement that “Typically, features are based on the bag-of-words presentation of tweets, but there are many subtle choices to be made.”

We propose to use another feature representation that covers the semantic meaning of words and is named distributed word embeddings (also known as word2vec). In 2010 Turney and Pantel published a survey on methods for building so-called ‘vector space models of semantics’. However, the big popularity of word embeddings started in 2013 when Mikolov et al. introduced a new architecture to build them [24, 25]. Their proposed method for learning word embeddings (as part of an unsupervised deep neural network training) was significantly more efficient than prior techniques and triggered many further applications [21, 33, 41].

We recommend semantic word embeddings as feature representation for supervised sentiment analysis since semantic and sentiment are tightly connected with each other. Already in the year 2003 Nasukawa and Yi applied semantic analysis to achieve better results in sentiment analysis [28]. They stated that “In order to improve the accuracy of the sentiment analysis, it is important to properly identify the semantic relationships between the sentiment expressions and the subject.” Word embeddings cover these semantic relationships since they consider the surrounding context words during the phase of learning a word’s embedding. The famous distributional hypothesis of Harris in 1954 states that words that occur in the same or similar contexts also have similar meanings [13]. These statements emphasize the value that semantic feature representations have for field of supervised sentiment analysis.

### 2.3.1 Simplified Comparison of Word Embeddings and Bag-Of-Word (BOW) Models

Supervised sentiment analysis systems for negative or abusive language are often trained on hundreds of thousands of training examples [29, 43]. Not in every use case these amounts of training data are available. Our 20,000 training sentences cover some, but lack in many other words or phrases used in daily political language. This section explains why word embeddings are the better feature representation choice if the training data quantity gets smaller.

Bag-of-words representations treat words as single independent units. If a classifier is once trained on the BOW representation of the word ‘good’ it is not able to perceive the word ‘great’ as similar or the word ‘bad’ as contrary (unless it has been trained on these words too). On the contrary, a distributed word embedding for the word ‘good’ does to some extent reflect that word’s relationship to other words like ‘bad’ or ‘great’. This is due to the fact that word embeddings are the result of an unsupervised training on huge corpora of unstructured text, which cover all the relationships between the single words. These word embedding training corpora are independent of all

Table 2 tries to explain this relationship (although it clearly simplifies vector sizes as well as mappings of sentiment to word embedding dimensions). In ‘real’ word embeddings a dimension (like in table 2 the first) cannot be mapped so easily to the meaning of ‘sentiment’ as in this example. Since word embeddings are the outcome of unsupervised trainings the resulting dimensions and their relations are unclear and unknown from a human perspective. What is clear is that they reflect language syntax and semantics ‘in some way’. This ‘inherent knowledge’ of word embeddings can extend the available training data in supervised classification systems. In other words: Word embeddings as feature representations for classification purposes can somehow increase training data quantity by adding their ‘inherent knowledge’ to the process.

feature representation	word	representation	sentiment training	sentiment prediction
bag-of-words	‘good’	[0, 1, 0, 0, 0, 0, 0, 0, 0, 0]	0	who knows?
	‘bad’	[0, 0, 0, 0, 0, 1, 0, 0, 0, 0]	1	
	‘great’	[0, 0, 0, 0, 0, 0, 1, 0, 0, 0]		
word2vec	‘good’	[ 0.8, 0.1, 0.2, 0.3, 0.4, 0.5]	0	very likely 0
	‘bad’	[-0.8, 0.1, 0.2, 0.1, 0.4, 0.5]	1	
	‘great’	[ 0.9, 0.1, 0.2, 0.4, 0.4, 0.5]		

Table 2: Comprehension example: Training of word sentiments with different feature representations

### 2.3.2 Pre-Trained Word Embedding Corpus

Distributed word embeddings in fact do need huge amounts of training data while they are created themselves. When a ready-to-use corpus of pre-trained word embeddings is employed, these embeddings already cover language dependencies and similarities learned from the corpus they were trained on beforehand. We used the pre-trained German word embedding corpus from polyglot [1]. Polyglot is a natural language processing (NLP) library for python. There are other word embedding corpora, for instance Google’s word2vec ?? (in English) or the GloVe embeddings ??.

The German embeddings that we use (from polyglot) are trained on the German part of Wikipedia. Polyglot offers other languages as well. Each embedding corpus covers the 100,000 most frequent words from the corresponding Wikipedia corpus. These words cover 92 percent of the content of the German Wikipedia web page. The resulting word embeddings have 64 dimensions. Each dimension is a floating point number. The polyglot word embeddings do not just cover words but also punctuation tokens.

## 2.4 Sentence and Word Tokenization

In our case, the application and training data reflect different text items. The application data covers whole speeches while the training data just consists of single sentences. To predict sentiment scores on sentence level we use a sentence tokenizer for all parliamentary speeches (see figure 1). We use the sentence tokenizer provided by polyglot. An alternative is the Natural Language Toolkit (NLTK) ?? which offers tokenizers on sentence and word level as well.

Sometimes language-dependent pre-processing steps have to be implemented even before the sentence tokenization unit. In our case the Austrian variant of German comprises some academic titles that do not exist in Germany. These titles followed by periods for abbreviation otherwise would have led to wrong sentence boundaries recognized by polyglot. To avoid this we simply replaced some Austrian academic titles with the word ‘title’ (‘Titel’ in German).

As soon as the application and training data is available as individual sentence strings the tokenization task proceeds. All sentences need to be split into their single words (or ‘tokens’ if punctuation is considered as well). For this step we also used the built-in word tokenizer of the polyglot library.

## 2.5 Pre-Processing

For building appropriate sentence embeddings (next section) it is crucial to retrieve many word embeddings per sentence from the word embedding corpus. The lower the amount of matches of words per sentence with the embedding corpus the worse the sentiment prediction accuracy gets. As said before our word embedding corpus covers the 100,000 most frequent words of the German Wikipedia pages. The German language provides the ability to produce new words by simply concatenating existing ones. This leads to a very high number of possible words which obviously are not all covered within the 100,000 most frequent words of our embeddings.

Additionally our training data sources partly contain messy data. As described before, the training sentences are sampled from several input sources. Due to an imperfect initial extraction of these sentences they encounter the following problems:

- complete sentences written in upper case, e.g. titles of press releases
- unnatural words with hyphens because of surviving end of line hyphens
- unnatural compound words because of missing spaces between words originally separated by line breaks

While the German language has a lot of compound words, many of the compound words in our training dataset do not exist in reality. To represent as much words per sentence as possible (by their corresponding word embedding), our pre-processing unit comprises the following steps if a word is not present within the embedding corpus:

- lemmatization and stemming (to find words simply not covered in their conjugated form)
- lowering and capitalization (to find words written in upper case)
- complex substring mechanism (to find original words in unnatural compound words)
- replacement of numbers by # (2017 = ####) since polyglot treats number tokens like that

Some of these steps were applied in a nested manner. According to the initial counts our training dataset comprises 40,000 unique words (or more exactly: strings separated by blanks). After these pre-processing steps the number dropped to about 30,000 unique words.

## 2.6 Sentence Embedding

To come from word to sentence embeddings we simply average all word embeddings related to a sentence. After pre-processing the words (to achieve high word coverage), all embeddings for the pre-processed words are retrieved from the embedding corpus. These (in our case) 64-dimensional word vectors are then averaged to build a 64-dimensional sentence vector. This is a basic approach for building distributed sentence embeddings. It is possible to build embeddings for entire documents (not just single sentences) as well. This averaging approach does not take the ordering of words into account. There are further, more advanced techniques to build sentence or document embeddings. We mention some of them in our conclusions section.

## 2.7 Classification

When Mozetič et al. compare different sentiment analysis applications state that: “Apparently, there is a wide range of machine learning algorithms used, and there is no consensus on which one to choose for the best performance. Different studies use different datasets, focus on different use cases, and use incompatible evaluation measures. There are additional factors with considerable impact on the performance, such as the natural language pre-processing of tweets, and formation of appropriate features.” [26]

We use a neural network classifier to learn and predict sentiment. Our classifier is built with the python library keras ??, runs 60 epochs and is configured as follows:

- one layer that takes 64 sentence input dimensions and maps them to 124 (rectangular activation function)
- one layer that maps 124 dimensions back to 64 (rectangular activation function)
- final layer that maps 64 dimensions down to 3 classes/buckets (softmax activation function)
- categorical cross-entropy loss function and adam optimizer

We also tested a dropout layer in-between that randomly sets 15% of the nodes to zero. These kinds of layers help to prevent overfitting of a model. A model is overfitted if it performs very well on training examples but not well on unseen test examples. However, our tests revealed that the accuracy for unseen test examples was better without the dropout layer.

We apply 10-fold cross-validation for validating our classification model. This technique trains the model on 90 % of the training data and evaluates it on the remaining (unseen) 10 %. This procedure is repeated ten times for ten different splits.

## 2.8 Aggregation

Since our system is trained on sentence-level sentiment scores (and therefore classifies sentences) we need some aggregation logic at the end of our pipeline. The simplest aggregation is to average all sentence-level sentiment scores to build document-level (or speech-level in our case) scores. If there is additional meta data available that describes the original documents (or speeches), such data can be very useful to both validate the model results and get further insights into the application data.

Our application data comprises many additional fields regarding names, gender and ages of politicians, background information for parties, topics, times and dates. These data items come from different sources and are stored in different databases. We use Tableau ?? for merging these data items all together to the sentence level. Furthermore we aggregate and visualize our sentiment scores by using built-in functions of Tableau. Some results of these aggregations can be seen in section 4.

## 3 Evaluation

We evaluated our method in two different ways. The next section shows how our word embedding approach compares to bag-of-words approaches by comparing the classifier accuracy measures precision and recall. The section afterwards illustrates a validation of our approach according to hypotheses that our political scientists defined prior to the implementation of our system.

### 3.1 Accuracy Measures on Training Data

Figure 2 shows the distribution of the average codings per sentence of our training data. The total number of sentences is 20,000. The scale ranges from 0 (not negative) to 4 (very negative). The color-coded sections reflect how we eventually split these training sentences into three buckets of an equal amount of training examples. The grey areas reflect gaps (of width 0.5) which are not considered for training and testing. This decision is based on the assumption that sentences classified as being near the border to the neighbor bucket are difficult to classify for

both humans and machines. To have clear definitions what should be in a bucket are not, we trained and evaluated our classification model without the sentences within these grey areas.

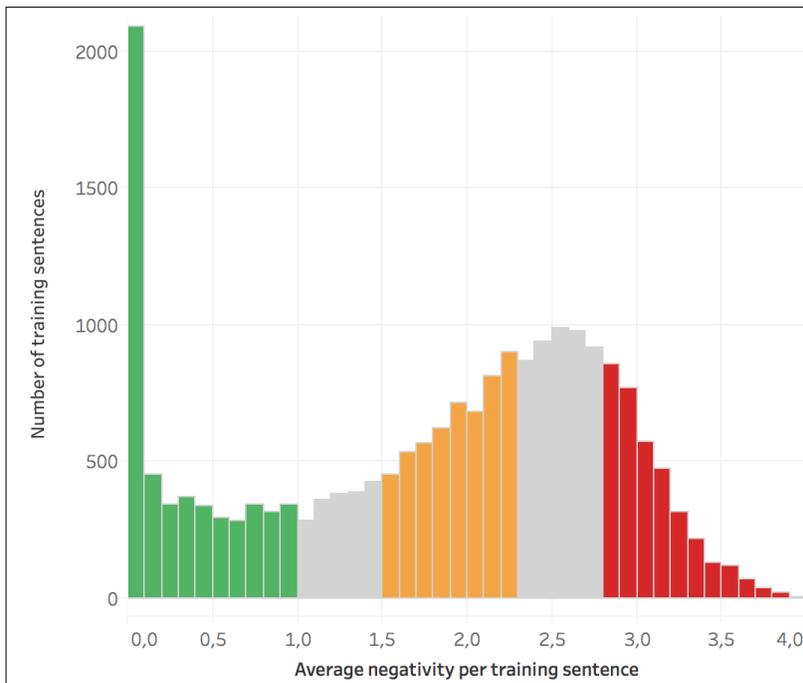


Figure 2: Negativity distribution of 20,600 training sentences from 0 (not negative) to 4 (very negative)

We actually tested four different bucketing settings. Two settings with two buckets and two settings with three buckets. All settings have an equal number of training examples within each bucket. Therefore, no over-sampling of specific buckets is needed. For each number of buckets two different settings are configured. One is without any gaps and therefore uses the whole training data set. The other configuration removes all sentences within 0.5 broad gap areas between the buckets. The accuracy results (precision and recall) for all four bucketing settings are shown in table 3. The configuration which we finally used for applying our method to Austrian parliamentary speeches is the setting in the last row of 3 and reflected by the distribution in 2.

number of buckets	precision c1	precision c2	precision c3	recall c1	recall c2	recall c3	summary
2	73	72		68	76		68-76
2 (with gap)	82	82		83	81		81-83
3	69	54	63	68	54	64	54-69
3 (with gap)	75	65	68	74	67	68	65-75

Table 3: Precision and recall in percent for 2 and 3 buckets with or without gaps (see description)

To interpret the results from table 3 we refer to a statement from Socher et al. [37]: “Bag of words classifiers can work well in longer documents by relying on a few words with strong sentiment like ‘awesome or ‘exhilarating’. However, sentiment accuracies even for binary positive/negative classification for single sentences has not exceeded 80% for several years. For the more difficult multiclass case including a neutral class, accuracy is often below 60% for short messages on Twitter (Wang et al., 2012).” Therefore a bag-of-words approach for the three classes positive/neutral/negative is often below 60% accuracy. The third row of table 3 reflects a three class approach based on word embeddings. The classifier in the third row is trained and validated on sentences from the whole distribution (without any gaps). Therefore the accuracy results of this row can be compared to the 60% reference for bag-of-words classifiers. The only difference here is that the definition of our buckets is not as clear as positive vs. neutral vs. negative, since we are working with a pure negativity scale. Our three buckets therefore are more fine-grained: not/slightly negative vs. negative vs. very negative. These three classes are harder to distinguish since changes on a negativity scale are not as obvious as changes from positive to neutral or from neutral to negative. However, precision and recall for the two outer classes are all above 60% (see third row in table 3). Just

the difficult middle class is 6% below that mark for both precision and recall. Therefore two out of three classes of our distributed word embedding approach received a higher accuracy than the average bag-of-words modelling approach even though our negativity range is harder to distinguish than a general sentiment range.

## 3.2 Hypotheses Validation on Application Data

Prior to our implementation of a supervised sentiment analysis system based on distributed word embeddings the political scientists in our team defined several hypotheses which they wanted to validate against our final classification results. We validated our approach against the following four hypotheses:

- Hypothesis 1: Debate speakers from government parties exhibit less negativity than debate speakers from opposition parties.
- Hypothesis 2: The level of negativity/incivility in parliamentary debates has increased over time.
- Hypothesis 3: Parliamentary party group leaders are most likely to use negative/provocative statements, followed by ordinary MPs. Cabinet members are least likely to use negative/provocative statements.
- Hypothesis 4: Urgent Question debates exhibit higher levels of negativity than other parliamentary debates.

We validate these hypotheses within the following section which covers a case study on negativity in Austrian parliamentary speeches.

## 4 Case Study: Negativity in Austrian Parliamentary Speeches

We now illustrate the benefits of our proposed method by using it to analyze Austrian parliamentary speeches from 1996 to 2013. What this data comprises is described in more detail in section 2.1. A quick summary: 56,000 speeches with 2.4 million sentences, 578 politicians and 7 political parties. We use our model, to investigate the set of research questions defined in section 3.2. All diagrams of this section are created with Tableau [39].

### 4.1 Government vs. Opposition

Hypothesis 1: Debate speakers from government parties exhibit less negativity than debate speakers from opposition parties.

Figures 3 to 6 show the average negativity level of five Austrian parties from 1996 to 2013. Two parties were excluded from this analysis since their speeches just cover small periods of time. Figure 3 shows the two parties that remained throughout the whole period on different sides of the government-opposition divide. The Peoples Party (ÖVP) was always a government party, though as junior or senior partner in different coalitions. The Greens remained always in opposition. The negativity of their speeches shows the expected pattern. The speeches of Green Members of Parliament (MPs) on average constantly score higher on the negativity scale than the speeches of ÖVP MPs. The trendline shown in the figure could also serve as a separator for government vs. opposition state.

Figure 4 shows the Social Democratic Party (SP"O) and the Freedom Party (FP"O). What these two parties have in common is that they switched from/to government/opposition back and forth. The line chart reflects this pattern clearly. The FP"O starts in opposition (with a higher average level of negativity) while the SP"O starts in government (with a lower negativity level). In the middle of the time axis there are about 6 years where both parties simultaneously change their behavior, before going back to their prior level. This is exactly the period of time, where the FP"O switched to government and the SP"O switched to opposition.

Figure 5 shows all five parties (excluding two short-term parties) of the Austrian national parliament during the investigated period of time. The figure shows the already mentioned four parties plus the orange-marked Alliance for the Future of Austria (BZ"O). This party broke away from the FP"O in 2005. When the BZ"O appeared, it directly went into the government for about 1.5 years without being elected before. This was a topic of discussion among all remaining parties and explains why the orange line is even high during this short period of being in government. If one would average the blue and the orange lines (back to one single blue FP"O line) the FP"O would continue its negativity level from the beginning (slightly above the Greens).

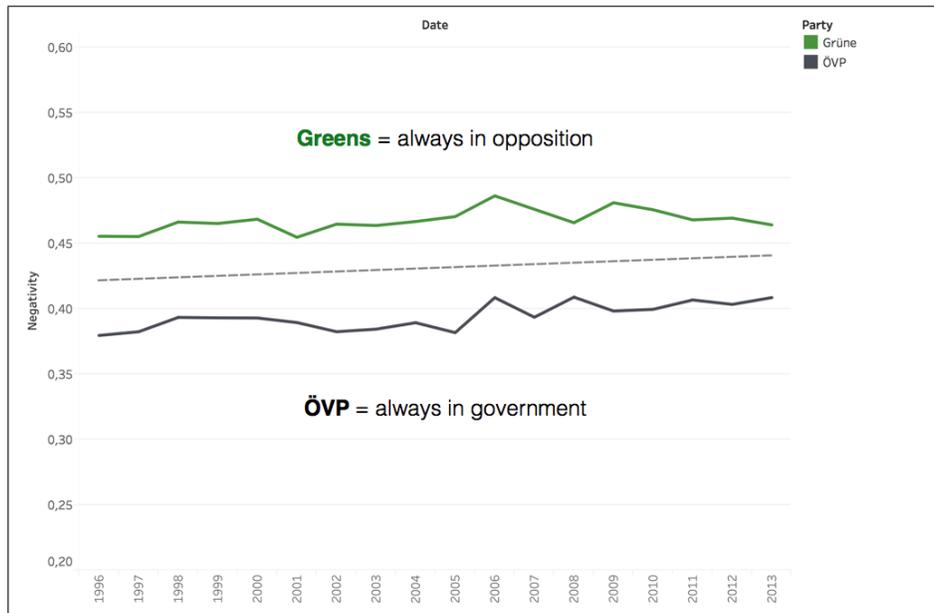


Figure 3: Negativity evolution of Austrian Greens and "ÖVP from 1996 to 2013 (including trendline)

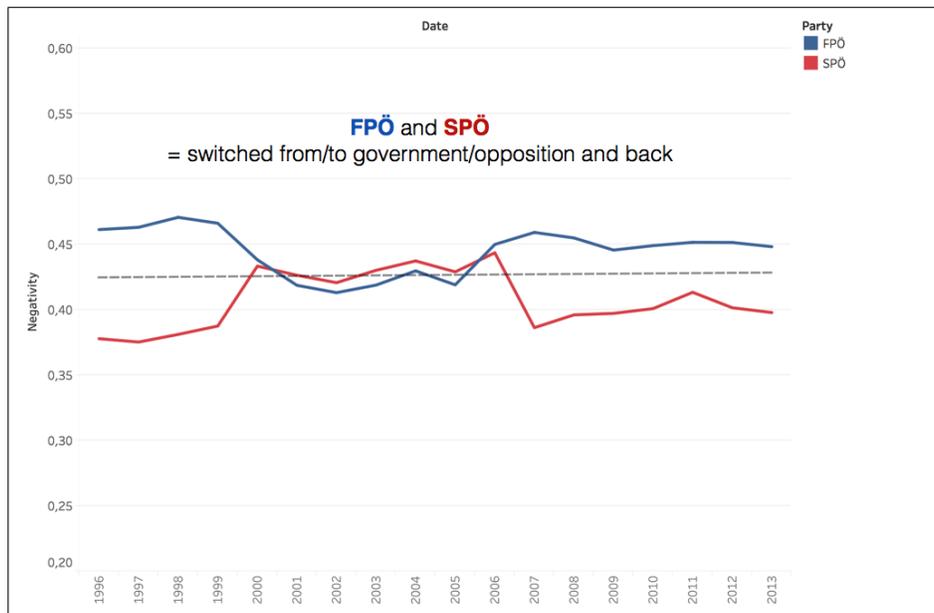


Figure 4: Negativity evolution of Austrian SP"O and FP"O from 1996 to 2013 (including trendline)

Figure 6 also shows all mentioned parties at once, including all the government coalitions for the years from 1996 to 2013. The changes for the blue and the red marked parties can be mapped clearly to their changes from opposition to government. The overall trendline shows that opposition and government parties nearly can be separated by this line.

## 4.2 Overall Trend

Hypothesis 2: The level of negativity/incivility in parliamentary debates has increased over time.

This hypothesis is easy to validate by just looking at the increase of the trendline in figure 6.

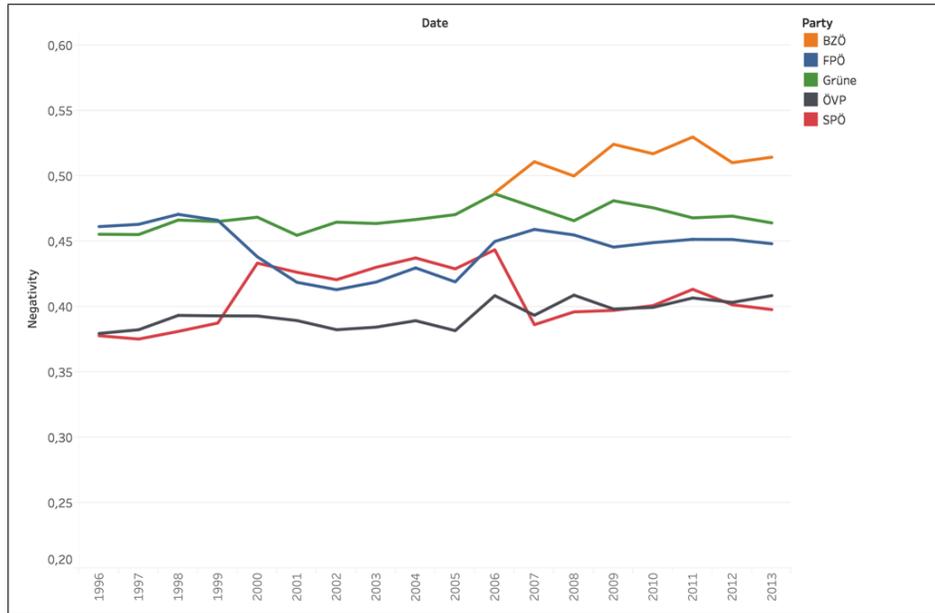


Figure 5: Negativity evolution of all Austrian parties from 1996 to 2013

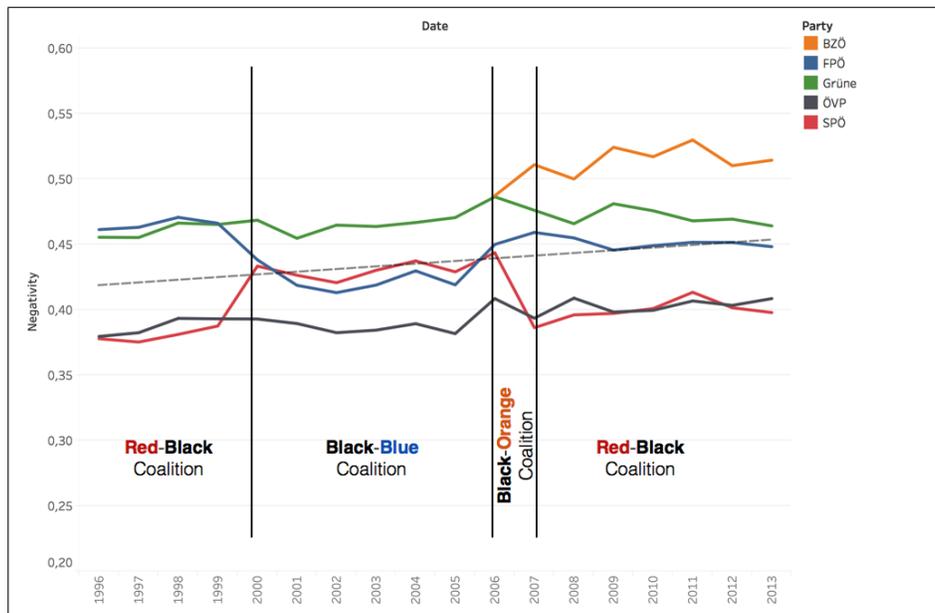


Figure 6: Negativity evolution of all Austrian parties from 1996 to 2013 (including trendline)

### 4.3 Parliamentary Roles

Hypothesis 3: Parliamentary party group leaders are most likely to use negative/provocative statements, followed by ordinary MPs. Cabinet members are least likely to use negative/provocative statements.

Figure 7 provides a fine-grained role differentiation: ministers, parliamentary group leaders, and ordinary MPs. The time period from 1996 to 2013 covers five legislative terms (five elections) in the Austrian national parliament. For every term (20st - 24th legislative period) the politicians who belong to one group are averaged. The pattern which can be seen here is the expected one. Parliamentary party group leaders dole out stronger attacks than the average Members of Parliament. Ministers furthermore tend to exhibit even more rhetorical restraint.

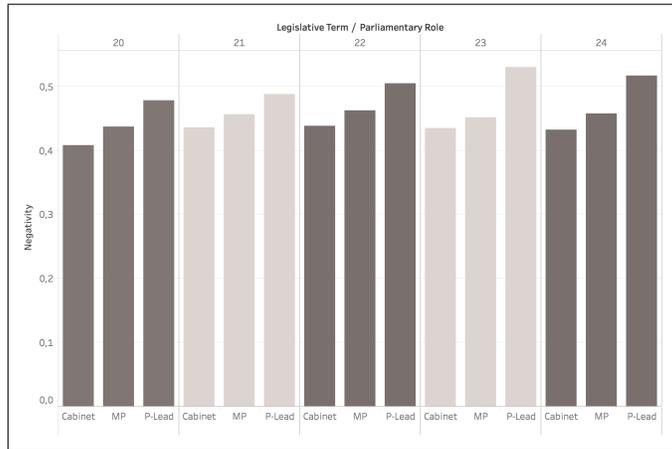


Figure 7: Negativity by parliamentary role: Averages on cabinet, MP (Member of Parliament) and P-Lead (parliamentary party group leader) per legislative term

#### 4.4 Urgent Questions

Hypothesis 4: Urgent Question debates exhibit higher levels of negativity than other parliamentary debates.

Figure 8 shows the average negativity level of ‘Urgent Questions’ compared to all other types of debates. The averages are built for all five legislative terms. It is easily perceivable that the urgent questions always receive a higher level of negativity in average than the other debate types.

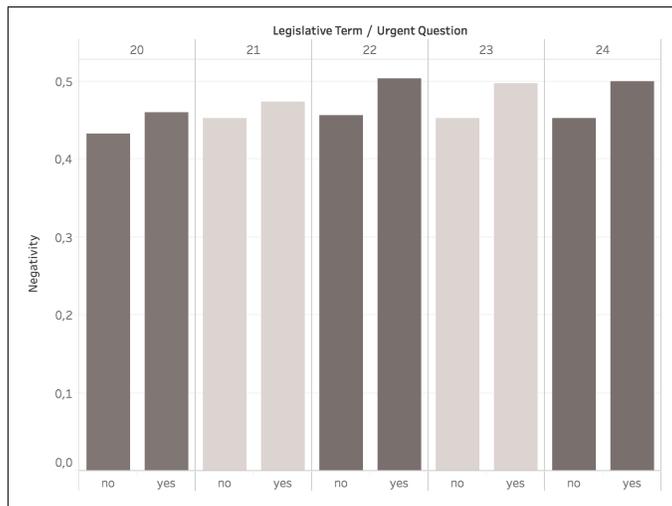


Figure 8: Average negativity of ‘Urgent Questions’ compared to other debates per legislative term

#### 4.5 Data Story

While we believe that the results of our case study are solid, we still want to conduct further analyses before reaching out broadly to media. Nevertheless the drahtwarenhandlung, who is a supplier of data driven stories for the ORF (Austrian broadcaster) has already published our preliminary results. As provider of data-journalistic content they did present the findings and a description of the method behind it in an English [8] and a German version [7].

## 5 Conclusions

We presented a novel method for supervised sentiment analysis based on distributed word embeddings. The advantage of this approach is that these types of embeddings inherently cover syntactic as well as semantic relationships of natural language. While we achieved good accuracy results and validated hypotheses from political scientists there are still some limitations of our evaluation and also some open issues for future work:

**Pre-Processing:** There are approaches to for handling the long compound words which are a characteristic of the German language. One method to handle these long compound words is the translation of words that are not present in the embedding corpus ‘unknown’ words into another language and the following translation of resulting single words back to German [10]. The word pre-processing unit of our proposed pipeline could be extended with such a logic to achieve better sentence embeddings by having a higher coverage of single word meanings (embeddings).

**Sentence/Document Embedding:** The introduction of distributed word embeddings [25, 24] in 2013 had a major impact on the field of natural language processing. Today, there are several approaches which build on the concept of distributed embeddings for modelling documents. Many of modelling solutions are more sophisticated than the concept of averaging ‘standard’ word embeddings which we introduced in this paper. We used this basic approach to introduce the concept of distributed word embeddings for supervised sentiment learning to social scientists. Future work could compare the accuracy results of more sophisticated embedding approaches to our current implementation. One example of another distributed embedding technique are document embeddings built according to the doc2vec approach [21]. Another option is to build and average ‘sentiment-specific’ word embeddings [41] instead of standard ones. The decomposable attention model [31] introduces another very promising method for document modelling. At the time of writing not all these approaches are suitable for German language. Parikh et al.’s model for instance relies on the Stanford Natural Language Inference (SNLI) dataset [4] which currently is just available for English.

**Visualizaton:** The last part of our proposed pipeline covers the aggregation and visualization of sentiment scores according to different meta data variables. These meta data fields depend comprise additional structured information extending the textual application data. Since our application data comprises many of these additional structured data items, we could implement a more sophisticated text visualization tool for exploring the whole dataset. There are many example publications that introduce advanced sentiment visualization tools [6, 12, 11]

**Deployment:** We plan to make our method implementation available as open source web application. Any research community that is interested in predicting sentiment based on a supervised learning approach should be enabled to do this with our tool. The feature set and language support of this future sentiment analysis web application are not defined at the time of writing.

## References

- [1] R. Al-Rfou, B. Perozzi, and S. Skiena. Polyglot: Distributed word representations for multilingual nlp. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 183–192, Sofia, Bulgaria, August 2013. Association for Computational Linguistics. URL <http://www.aclweb.org/anthology/W13-3520>.
- [2] J. M. Berry and S. Sobieraj. *The Outrage Industry: Political Opinion Media and the New Incivility*. Oxford University Press, 2013.
- [3] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3 (Jan):993–1022, 2003.
- [4] S. R. Bowman, G. Angeli, C. Potts, and C. D. Manning. A large annotated corpus for learning natural language inference. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2015.
- [5] D. J. Brooks and J. G. Geer. Beyond Negativity: The Effects of Incivility on the Electorate. *American Journal of Political Science*, 51(1):1–16, 2007.

- [6] N. Diakopoulos, M. Naaman, and F. Kivran-Swaine. Diamonds in the rough: Social media visual analytics for journalistic inquiry. In *Visual Analytics Science and Technology (VAST), 2010 IEEE Symposium on*, pages 115–122. IEEE, 2010.
- [7] S. Emrich. Politikerinnen auf den Mund geschaut. Drahtwarenhandlung (dwh) GmbH, 2017. URL <http://film.drahtwarenhandlung.at/de/news/politikerinnen-auf-den-mund-geschaut/>.
- [8] S. Emrich. Sentiment analysis of austrian members of parliament. Drahtwarenhandlung (dwh) GmbH, 2017. URL <http://film.drahtwarenhandlung.at/en/news/sentiment-analysis-of-austrian-members-of-parliament/>.
- [9] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. From data mining to knowledge discovery in databases. *AI Magazine*, 17(3):37, 1996.
- [10] F. Fritzing and A. Fraser. How to avoid burning ducks: Combining linguistic analysis and corpus statistics for german compound processing. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR*, pages 224–234. Association for Computational Linguistics, 2010.
- [11] V. Gold, C. Rohrdantz, and M. El-Assady. Exploratory Text Analysis using Lexical Episode Plots. In E. Bertini, J. Kennedy, and E. Puppo, editors, *Eurographics Conference on Visualization (EuroVis) - Short Papers*. The Eurographics Association, 2015. doi: 10.2312/eurovisshort.20151130.
- [12] M. L. Gregory, N. Chinchor, P. Whitney, R. Carter, E. Hetzler, and A. Turner. User-directed sentiment analysis: Visualizing the affective content of documents. In *Proceedings of the Workshop on Sentiment and Subjectivity in Text*, pages 23–30. Association for Computational Linguistics, 2006.
- [13] Z. S. Harris. Distributional Structure. *WORD*, 10(2-3):146–162, 1954.
- [14] R. Herbst. *Rude Democracy: Civility and Incivility in American Politics*. Temple University Press, Philadelphia, 2010.
- [15] D. J. Hopkins and G. King. A Method of Automated Nonparametric Content Analysis for Social Science. *American Journal of Political Science*, 54(1):229–247, 2010.
- [16] B. A. Howard. The art and science of data-driven journalism. *Tow Center for Digital Journalism, Columbia University*, 2014.
- [17] K. H. Jamieson. *Dirty Politics*. Oxford University Press, New York, NY, 1992.
- [18] K. F. Kahn and P. J. Kenney. Do Negative Campaigns Mobilize or Suppress Turnout? Clarifying the Relationship between Negativity and Participation. *American Political Science Review*, 93(04):877–889, 1999.
- [19] S. Kandel, J. Heer, C. Plaisant, J. Kennedy, F. van Ham, N. H. Riche, C. Weaver, B. Lee, D. Brodbeck, and P. Buono. Research directions in data wrangling: Visualizations and transformations for usable and credible data. *Information Visualization*, 10(4):271–288, 2011.
- [20] R. R. Lau and I. B. Rovner. Negative campaigning. *Annual Review of Political Science*, 12:285–306, 2009.
- [21] Q. Le and T. Mikolov. Distributed representations of sentences and documents. In *International Conference on Machine Learning (ICML)*, 2014.
- [22] B. Lee, N. Henry Riche, P. Isenberg, and S. Carpendale. More than Telling a Story: A Closer Look at the Process of Transforming Data into Visually Shared Stories. *IEEE Computer Graphics and Applications*, 35(5):84–90, Sept. 2015. doi: 10.1109/MCG.2015.99. URL <https://hal.inria.fr/hal-01158445>.
- [23] G. Lengauer, F. Esser, and R. Berganza. Negativity in political news: A review of concepts, operationalizations and key findings. *Journalism*, 13(2):179–202, 2012.
- [24] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *ICLR Workshop*, 2013.

- [25] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc., 2013. URL <https://code.google.com/archive/p/word2vec/>.
- [26] I. Mozetič, M. Grčar, and J. Smailović. Multilingual twitter sentiment classification: The role of human annotators. *PloS one*, 11(5):e0155036, 2016.
- [27] D. C. Mutz and B. Reeves. The New Videomalaise: Effects of Televised Incivility on Political Trust. *American Political Science Review*, 99(01):1–15, 2005.
- [28] T. Nasukawa and J. Yi. Sentiment analysis: Capturing favorability using natural language processing. In *Proceedings of the 2nd international conference on Knowledge capture*, pages 70–77. ACM, 2003.
- [29] C. Nobata, J. Tetreault, A. Thomas, Y. Mehdad, and Y. Chang. Abusive language detection in online user content. In *Proceedings of the 25th International Conference on World Wide Web*, pages 145–153. International World Wide Web Conferences Steering Committee, 2016.
- [30] B. Pang, L. Lee, and S. Vaithyanathan. Thumbs Up? Sentiment Classification Using Machine Learning Techniques. *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP-2002)*, Philadelphia, Pennsylvania, July 6-7, 2002, pages 79–86, 2002.
- [31] A. P. Parikh, O. Täckström, D. Das, and J. Uszkoreit. A decomposable attention model for natural language inference. *arXiv preprint arXiv:1606.01933*, 2016.
- [32] Parliament Austria. Parliamentary speeches from the austrian national parliament, 2013. URL <https://www.parlament.gv.at/PERK/NRBRBV/NR/STENO/>.
- [33] J. Pennington, R. Socher, and C. D. Manning. GloVe: Global Vectors for Word Representation. In *Proceedings of the Empirical Methods in Natural Language Processing (EMNLP 2014)*, volume 14, pages 1532–1543, 2014. URL <https://nlp.stanford.edu/projects/glove/>.
- [34] G. Salton and M. J. McGill. Introduction to modern information retrieval. 1986.
- [35] P. Sam. Feelings, nothing more than feelings: The measured rise of sentiment analysis in journalism, 2013. URL [www.niemanlab.org/2013/01/feelings-nothing-more-than-feelings-the-measured-rise-of-sentiment-analysis-in-journalism/](http://www.niemanlab.org/2013/01/feelings-nothing-more-than-feelings-the-measured-rise-of-sentiment-analysis-in-journalism/).
- [36] E. Segel and J. Heer. Narrative visualization: Telling stories with data. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1139–1148, 2010.
- [37] R. Socher, A. Perelygin, J. Y. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the conference on empirical methods in natural language processing (EMNLP)*, volume 1631, page 1642. Citeseer, 2013.
- [38] C. D. Stolper, B. Lee, N. H. Riche, and J. Stasko. Emerging and recurring data-driven storytelling techniques: Analysis of a curated collection of recent stories. *Microsoft Research Technical Report*, 2016.
- [39] C. Stolte, D. Tang, and P. Hanrahan. Polaris: A system for query, analysis, and visualization of multidimensional relational databases. *Visualization and Computer Graphics, IEEE Transactions on*, 8(1):52–65, 2002.
- [40] J. Stray. What do journalists do with documents? Field notes for natural language processing researchers. In *Computation + Journalism Symposium at Stanford University*, 2016.
- [41] D. Tang, F. Wei, N. Yang, M. Zhou, T. Liu, and B. Qin. Learning sentiment-specific word embedding for twitter sentiment classification. In *ACL (1)*, pages 1555–1565, 2014.
- [42] P. D. Turney and P. Pantel. From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research*, 37:141–188, 2010.
- [43] E. Wulczyn, N. Thain, and L. Dixon. Ex machina: Personal attacks seen at scale. *CoRR*, abs/1610.08914, 2016. URL <http://arxiv.org/abs/1610.08914>.