More than Bags of Words: Sentiment Analysis with Word Embeddings

Elena Rudkowsky[1], Martin Haselmayer[2], Matthias Wastian[3], Marcelo Jenny[4], Štefan Emrich[3], and Michael Sedlmair[1]

[1]*University of Vienna (Faculty of Computer Science)*
[2]*University of Vienna (Department of Government)*
[3]*Technical University of Vienna (Center for Computational Complex Systems)*
[4]*University of Innsbruck (Department of Political Science)*
[5]*Drahtwarenhandlung (dwh) GmbH, Vienna*
[6]*Jacobs University Bremen (Computer Science)*

Manuscript accepted for publication in *Communication Methods and Measures*, Special Issue on Computational Methods

Keywords: textual analysis, computer modeling, communication research methods, quantitative methods

Abstract

Moving beyond the dominant bag-of-words approach to sentiment analysis we introduce an alternative procedure based on distributed word embeddings. The strength of word embeddings is the ability to capture similarities in word meaning. We use word embeddings as part of a supervised machine learning procedure which estimates levels of negativity in parliamentary speeches. The procedure's accuracy is evaluated with crowdcoded training sentences; its external validity through a study of patterns of negativity in Austrian parliamentary speeches. The results show the potential of the word embeddings approach for sentiment analysis in the social sciences.

Address correspondence to:
Elena Rudkowsky, elena.rudkowsky@univie.ac.at, T +43-1-4277-79040
Währinger Str. 29/S6, 1090 Vienna-Austria

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

Introduction

Sentiment analysis has become a major area of interest in communication research. Recent applications include analyses of media tone (Van Atteveldt et al., 2008; Hopkins and King, 2010; Young and Soroka, 2012; Soroka and McAdams, 2015; Soroka et al., 2015; Haselmayer and Jenny, 2017), agenda setting (Ceron et al., 2016), framing (Burscher et al., 2014), election forecasting (Ceron et al., 2015, 2017) and candidate evaluations (Aaldering and Vliegenthart, 2016). These studies do automated text analysis with sentiment dictionaries (Young and Soroka, 2012; Aaldering and Vliegenthart, 2016; Haselmayer and Jenny, 2017) or use machine learning (Van Atteveldt et al., 2008; Hopkins and King, 2010; Burscher et al., 2014; Ceron et al., 2016) to get sentiment scores. What these studies share is a bag-of-words approach towards text data. In a nutshell, the bag-of-words representation of text treats words as independent units. Few studies attempt to include semantic or syntactic relations between words (van Atteveldt et al., 2008, 2017; Wuuest et al., 2011).

The goal of this article is to move sentiment analysis in communication research forward by presenting a new approach that has become popular in natural language processing and computer science: the use of distributed word embeddings, (Al-Rfou et al., 2013; Le and Mikolov, 2014; Mikolov, Chen, et al., 2013; Mikolov, Sutskever, et al., 2013; Pennington et al., 2014). Word embeddings represent (or embed) words in a continuous vector space in which words with similar meanings are mapped closer to each other. New words in application texts that were missing in training texts can still be classified through similar words (Goldberg, 2016; Mikolov, Chen et al., 2013), an advantage compared to the bag-of-words approach in which new words encountered in application texts are a nuisance.

We describe a procedure with word embeddings that enables us to estimate levels of negativity in parliamentary speeches and then apply it to a sample of 56,000 plenary speeches from the Austrian parliament. The procedure's accuracy is evaluated with the help of crowdcoded training sentences; its external validity by studying negativity in Austrian parliamentary speeches. The different levels of negativity that we find for speakers in different roles (minister, parliamentary party group leader, ordinary Member of Parliament) from government or opposition parties accord with common sense hypotheses about expected patterns. From these results we conclude that the word embeddings approach offers a lot of potential for sentiment analysis, and more generally automated text analysis in the social sciences (Wilkerson and Casas, 2017; Boumans and Trilling, 2016; Lucas et al., 2015; Grimmer and Stewart, 2013; Lowe and Benoit, 2013).

Related Work

Until recently, sentiment analysis in the social sciences almost exclusively relied on the *bag-of-words* approach. Mozetič et al. (2016) compare a variety of sentiment classification applications for Twitter data and find almost all of them using it. Researchers rely on existent sentiment dictionaries (Kleinnijenhuis et al., 2013) or create customized and context-sensitive dictionaries for their research questions (Young and Soroka, 2012; Aaldering and Vliegenthart, 2016; Haselmayer and Jenny, 2017). A third group of studies uses machine learning applications (Van Atteveldt et al., 2008; Hopkins and King, 2010; Burscher et al., 2014; Ceron et al., 2015, 2017). Some studies have reported good results for measuring sentiment at the level of articles or speeches (Hopkins and King, 2010), but the assumptions and simplifications that the bag-of-words approach entails, such as loss of grammatical structure or of context-dependent word meanings have been repeatedly pointed out (Grimmer and Stewart, 2013; Lowe and Benoit, 2013).

Semantic models offer an improvement as they take relationships between words into account, but they have been rarely used in communication research (van Atteveldt et al., 2008, 2017). This is surprising because Harris' (1954) distributional hypothesis that words occurring in the same or similar contexts tend to have similar meanings is old and well known to computational linguists. Such word context information can be fruitfully employed for sentiment analysis (e.g. Nasukawa and Yi, 2003).

Turney and Pantel (2010) provide an overview of earlier work on vector space models. The word embeddings approach gained significant attention after Mikolov and colleagues introduced a more efficient architecture for creating reusable word vector representations from large text corpora (Mikolov, Chen, et al., 2013; Mikolov, Sutskever, et al., 2013). A number of applications quickly followed (Le and Mikolov, 2014; Pennington et al., 2014; Tang et al., 2014).

Supervised Sentiment Analysis with Word Embeddings

Figure 1 presents a process pipeline of our embedding-based sentiment analysis procedure. In the upper half we see three databases: a corpus of text documents for which we want sentiment scores (labeled 'application data'), a set of sentences whose sentiment scores have been established by human coders (training data), and a word embedding corpus that transforms these two data sources. The lower section depicts the steps needed to train the system and collect new sentiment scores. The depiction of processing units follow the 'Knowledge Discovery in Databases' (KDD) framework for data mining applications (Fayyad et al., 1996). Black arrows indicate input and output flows. Dashed arrows in light gray denote the training cycle for building a classification model from training data. Arrows in dark gray indicate the prediction phase that generates new sentiment scores for application data.

[Figure 1 about here]

Let us look at the three data sets first. Training and application data are closely related and therefore should come from similar text sources. In principle, that does not apply for the transformation data set. Word embeddings represent word meaning based on their occurrence in a text corpus. If the text corpus is very large and diverse, the word embedding vectors will represent the general (or dominant) meaning of a word, which should be useful in a variety of applications. Once the three data sets have been selected, the training cycle develops a classification model for the training data. Finding a good classification model usually requires multiple simulation runs with varying parameter settings. Evaluation of the model's accuracy with previously unseen samples from the training data follows. When it is considered satisfactory, it can be applied to new texts.

Application Data: Documents

The application data in Figure 1 represents the 'data of interest' for which a researcher wants sentiment scores. The documents could be newspaper articles, transcribed speeches or social media content. Our documents consist of about 56,000 German language speeches from the Austrian national parliament covering the period from 1996 to 2013 (Parliament Austria, 2013). The classifier unit predicts a sentiment score at the level of individual sentences. 578 MPs produced about 2.4 million sentences for which we want to obtain sentiment, or more specifically negativity, scores aggregated to the level of individual speeches. Metadata about the documents are helpful to interpret and present the sentiment scores. In our application they include additional variables on speaker, party and date of the speech.

Training Data: Labeled Sentences

To train a classifier in a supervised machine learning application we need 'ground truth' data. That ground truth comes from training data which should mimic the application data's vocabulary (Young and Soroka, 2012; Haselmayer and Jenny, 2017). Asking humans to code the level of sentiment for complete parliamentary speeches appears extremely challenging. So we asked to them to code sentiment for single sentences. Sentences as coding units 'tend to be efficient and reliable' (Krippendorff 2013, p. 110).

We use training data comprising sentences from party press releases, transcripts of parliamentary speeches, and media reports (Haselmayer and Jenny, 2017). Hence, they contain very similar language[1]. Our training data consist of 20,580 sentences. Each sentence was labeled by at least ten German-speaking coders recruited from the crowdsourcing platform CrowdFlower. Codings range from not negative (0) to very negative (4) on a 5-point scale. Coders could also identify a sentence as uncodable. We cover only the neutral and negative parts of the sentiment scale. Psychological research has highlighted asymmetries between positive and negative evaluations of situations, persons, or events. People devote more attention and cognitive effort to negative information, which contributes more strongly to the overall impression (e.g. Baumeister et al., 2001; Rozin and Royzman, 2001).

Coders are asked to rate only the manifest content of the text[2]. A recent experimental study reports that partisan preferences of crowdcoders do not affect sentiment ratings at the aggregate level (Haselmayer et al., 2017). During the coding process, we monitor individual coder performance to identify cheating or spamming. We use test questions with predetermined correct answers and exclude contributors failing a 75% accuracy threshold on test questions. Recent studies demonstrate that crowdcoding political issues (Benoit et al. 2016) and sentiment strength of sentences produces valid results (Haselmayer and Jenny, 2017; Lind et al., 2017). To obtain a sentence score we compute the mean of all coders. This produces equal results as more complex aggregation measures according to recent

crowdcoding studies (Haselmayer and Jenny, 2017; Benoit et al., 2016). We evaluate coding quality by comparing this with a 'expert' mean coding by some of the authors. The Pearson correlation is 0.82 for a random sample of 200 sentences. In line with previous research, we find that a group of lay coders is able to replicate expert coding (e.g. Benoit et al., 2016).

Transformation Data: Word Embeddings

The third database in Figure 1, which we call transformation data, contains the word embeddings. This component is absent in bag-of-words models since they are built only from the words appearing in the training and application data. There are ready-to-use corpora with pre-trained word embeddings, for instance, Google's word2vec (Mikolov, Sutskever, et al., 2013) or the GloVe embeddings (Pennington et al., 2014). We use a pre-trained German word embedding corpus from Polyglot (Al-Rfou et al., 2013), which is a natural language processing library for Python. These off-the-shelf embedding corpora can be used like dictionaries, but instead of translations or meanings, they return vector embeddings for the requested words. Therefore, the usage of pre-trained embeddings does not require any further computing time.

Supervised sentiment analysis tools are often trained on hundreds of thousands of training examples (Nobata et al., 2016; Wulczyn et al., 2016). Yet, gathering huge training datasets is not always possible: it is expensive, time-consuming, and thus often unaffordable. Our approach provides an affordable solution for large-scale text analysis that should be applicable to various languages and contexts. Such research may either draw on existing databases with labeled training data[3] or generate their own training data without too much effort. Our training data contain 20,580 sentences. Limited amounts of human-labeled training data lower the accuracy of the corresponding classification models. Word embeddings may increase the

accuracy of classification models as they provide information (and vector representations) on words that are not or only scarcely represented in the training data based on their similarity with other words (Goldberg, 2016).

Figure 2 shows how sentiment can be reflected by word embeddings. This simplified example illustrates the mapping of sentiment to word embeddings. A distributed word embedding for the word ‘good’ reflects to some extent this word’s relationship to other words - ‘good’ is close to ‘great’ and distant to ‘bad’. If an embedding-based classification model is trained on sentences that contain the words ‘good’ and ‘bad’ it is later on able to perceive the word ‘great’ as similar to ‘good’ even though it has never seen that word before. In contrast, bag-of-words representations treat words as single independent units (one-hot representations). If a classifier is trained on the bag-of-words representation of the word ‘good’, it is not able to perceive the word ‘great’ as similar or the word ‘bad’ as converse (unless it has been successfully trained on these words, too).

[Figure 2 about here]

Polyglot’s language-dependent word embeddings are trained on Wikipedia and represent the 100,000 most frequent words for each language. For German, those words cover 92% of the words in the German Wikipedia articles. Word coverage is significantly higher for languages using fewer morphological forms: 96% for English and 99.7% for Chinese (Al-Rfou et al., 2013). Each word embedding has 64 dimensions with each dimension being set to a floating point number. These dimensions correlate with language structure and meanings. In a well-trained embedding corpus arithmetic operations on word embeddings result in vectors that reflect underlying language patterns. A standard example to explain such relationships is that the embedding of the word ‘man’ stands to ‘woman’ as ‘king’ stands to ‘queen’. Another

example is that the arithmetic vector operation on ‘Paris’ - ‘France’ + ‘Italy’ results in a vector that is very close to ‘Rome’ (Mikolov, Chen, et al., 2013).

Sentence and Word Tokenization

The beginning of the processing pipeline in Figure 1 shows a sentence and a word tokenization component. The sentence tokenizer is necessary because our unit for training examples are sentences; therefore, we split the application data into natural sentences in the prediction phase.[4] We use the sentence tokenizer provided by Polyglot. An alternative is the Natural Language Toolkit (Loper and Bird, 2002) which also offers tokenizers at the level of sentences and words. We continue with splitting the sentences of training and application data further into single words or ‘tokens’ using the word tokenizer of the Polyglot library. We keep punctuation since Polyglot’s word embedding corpus comprises embeddings for punctuation as well. For sentiment analysis, exclamation or question marks can be useful for determining the negativity of a sentence.

Preprocessing

Preprocessing has tremendous consequences for the quality of automated text analysis. Recent studies demonstrate how preprocessing decisions impact on sentiment analysis (Haselmayer and Jenny, 2017) or dimensional scaling (Greene et al., 2016) results. Yet, the amount of necessary preprocessing also depends on the quality of the raw data. This is especially important here as the creation of sentence embeddings (see below) depends heavily on the retrieval of an embedding for each word in a sentence. Fewer matching word embeddings per sentence decrease the accuracy of sentiment prediction. As mentioned above, the Polyglot word embeddings cover the 100,000 most frequent words of Wikipedia. As the German language contains a lot of compound words, we have to deal with numerous context-specific

compounds in our training data that are not covered by those 100,000 embeddings. Our training data further include entire sentences in uppercase, incorrectly hyphenated words (due to end-of-line hyphens) or 'artificial' compound words (due to missing spaces). Using texts from similar, but not exactly the same sources introduces additional noise (Kandel et al., 2011).

In order to represent as many words per sentence as possible (by their corresponding word embedding), we apply various preprocessing techniques to words that have no match in the embedding corpus. We use lemmatization and stemming to find words that are not contained in their conjugated form. We lowercase or capitalize words to overcome the uppercase issue. We check multiple substring combinations to retrieve embeddings for substrings. We replace numbers by hashes (2018 = ####) to match Polyglot's fashion of representation of digits. These preprocessing steps reduce the number of unique words (i.e. strings separated from blanks) from roughly 40.000 to about 30,000 and increase prediction accuracy by three percentage points.

Sentence Embedding

The sentence embedding unit of our approach averages all retrieved word embeddings per sentence by calculating the mean vector. This is a basic approach for building distributed sentence embeddings which does not take the ordering of words into account. We use this simple averaging approach as our main motivation is to introduce word embeddings for sentiment analysis to social scientists in general. Recent applications, such as the doc2vec approach, combine word and document embeddings for sentiment classification (Le and Mikolov, 2014) or generate 'sentiment-specific' word embeddings (Tang et al., 2014).

Classification

After all sentences have been transformed into their corresponding embeddings a classification model is applied to determine their sentiment. Comparing several sentiment analysis applications Mozetič et al. (2016) state that: “A wide range of machine learning algorithms is used, and apparently 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." (Mozetič et al., 2016, p. 1). Moraes et al. (2013) compare support vector machines and artificial neural networks for document classification tasks in various settings. Their experiments indicate that artificial neural networks produce superior results in many applications. Thus we apply a neural network classifier using the Keras Python library (Chollet, 2015).

We use 10% of the training data as test data that the model does not see during the training cycle (illustrated by dashed arrows in light gray in Figure 1) to estimate an error rate for the application data after choosing the final model. To avoid any bias we generate ten random test samples. We build our model with the remaining training data and subsequently calculate the average accuracy for the previously unseen samples. The average accuracy is our estimate of the model’s performance on unlabeled new data.

To identify the best model we test a variety of parameter settings. To compare their performance, we split off another 10% of the remaining 90% training data as our validation data. We use the categorical cross-entropy function for our model and a 3-dimensional output layer with a softmax activation function which creates three different categorical sentiment output labels. These are typical choices for classification problems which remain stable during the hyperparameter tuning. The hyperparameter tuning of the number of neurons on the first layer (64, 128), the number of hidden layers (1, 2), the number of neurons of the hidden layers (16, 32, 48), the dropout rates on the input or hidden layers (0, 0.1, 0.2, 0.3, 0.4), the optimizer used (stochastic gradient descent, Adam) and the learning rate (0.01, 0.001) relies on the grid

search technique. The first layer of the best performing neural network has 64 neurons, a rectifier activation function and a dropout rate of 30%. This should prevent overfitting on the training data, which typically results in low accuracy on the unseen test data. Our final model applies a 32-dimensional second layer with the rectifier activation function and uses the Adam optimizer (Kingma and Ba, 2014).

Aggregation

The final aggregation and visualization component of the pipeline in Figure 1 is important to get insights from the calculated sentiment scores. Our approach produces sentiment scores at the sentence level. We further aggregate these scores to the level of speeches by calculating the mean sentiment score. We then analyze parliamentary debates using additional information on speakers, their party affiliation, the government status or the date and the legislative period of a speech. We visualize differences between these categories using Tableau (Stolte et al., 2002) and present results on patterns of negativity in the case study section.

Evaluation

We present a two-fold evaluation of our approach. First, we measure the accuracy of our model on previously unseen training examples and compare its accuracy with a bag-of-words approach. Second, we test its external validity with several hypotheses on expected patterns of negativity in parliamentary speeches in the Austrian parliament.

Accuracy Measures

Figure 3 shows the distribution of mean codings of the 20,580 sentences in our training data set. Sentences were coded on a scale ranging from 0 (not negative) to 4 (very negative). The color-coded sections indicate their allocation to three classes of negativity (not/slightly negative vs. negative vs. very negative).

[Figure 3 about here]

Imbalanced class distributions of training examples can lead to biased and inaccurate results. To balance our training data we weight the sentence embeddings according to their class frequency during the training phase. With this setting we achieve an average accuracy of 58% on our three classes on previously unseen test data. The results are validated through multiple random sampling. A bag-of-words alternative with TF-IDF (Salton and McGill, 1986) and a multinomial Naive Bayes classifier achieved an average accuracy slightly below 55% on the same data. We chose a multinomial Naive Bayes classifier for the bag-of-words implementation because it performed better than a neural network classifier for our application (Raschka, 2014). Neural network toolkits typically perform less well with very high-dimensional, sparse vectors, such as bag-of-words representations, where every feature has its own dimension (Goldberg, 2016).

[Tables 1 and 2 about here]

A detailed evaluation comparison between the bag of words and the distributed words embeddings approach is shown in tables 1 and 2. It can be easily seen that the word

embeddings approach outperforms the bag of words approach with regard to the not/slightly negative, where the F1 score is substantially higher. For the very negative class, both approaches attain similar levels of accuracy. Regarding the negative class, the bag of words approach shows a substantially higher precision and a slightly lower recall due to the tendency of the bag of words approach to predict the middle class (negative). We suggest to interpret this evaluation also in a task-dependent way: E.g., the tables tell us that if the bag of words model predicts a sentence to be not or slightly negative, we can trust this quite a bit (due to the relatively high recall). Yet, it misses a lot of correct not/slightly negative sentences (due to its low precision). The word embeddings model returns a lot more sentences classified as not/slightly negative, but due to its lower precision we cannot put as much trust in these decisions as in those made by the bag of words model. If, for example, it would be important to us to get all the not/slightly negative sentences and we wouldn't mind checking the suggested sentences manually, the word embeddings model would definitely be the model to go with. If we are not interested in getting all the not/slightly negative sentences, e.g. because we only want to present some examples, and we don't have a lot of time for checking the model output, the bag of words model could be superior. In fact, the different prediction behaviour of the two models could lead to a very promising model ensemble which is the focus of future work.

Socher et al. (2013) report benchmarks for binary and multiple classification tasks that put the performance of our approach into a broader perspective. In general, bag-of-words classifiers of longer documents work quite well even if they only rely on a few strong sentiment words. Accuracy for binary (positive vs. negative) sentiment classification at the sentence-level has remained quite stable at about 80% in recent years. For more difficult tasks, such as multiclass cases including a neutral category, accuracy is often below 60% (Socher et al., 2013: 3, Wang et al., 2012). Our word embedding approach is close to 60% accuracy for three classes. Yet, our implementation deals with degrees of negativity: not/slightly negative vs. negative vs.

very negative. These classes are much harder to separate than a change of polarity from positive to neutral and from neutral to negative (which Socher et al. 2013 refer to as 'multiclass'). In addition, we reach this level of accuracy analyzing German texts, which is more challenging than dealing with English language due to language complexity and the availability of tools for natural language processing (Haselmayer and Jenny, 2017). This also applies to the availability of domain specific word embeddings. Hence, we would expect an increase in accuracy if we could have used word vectors for political communication, rather than relying on the rather general Polyglot corpus.

Case Study: A validation of the procedure with patterns of negativity in Austrian Parliamentary Speeches

Sentiment analysis has increasingly turned to parliamentary debates as substantively interesting objects of study (e.g. Slapin and Proksch, 2014; Rheault, Beelen, Cochrane and Hirst, 2016). A key component of what political opposition parties and their members do in parliaments of democratic systems is criticizing the government parties' policy ideas and the government ministers' work. Ministers transform policy ideas into bills introduced to parliament and they are responsible for their subsequent implementation. Members of government parties provide rhetorical support for the government's bills in parliamentary debates and the crucial votes for their passage. In this role they will criticize policy proposals of the opposition, but more often they will play defense for the government in a supporting role: "in parliamentary democracies the governing parliamentary party groups and the executive form a functional 'unit' which somewhat limits the functional independence of the majority parties and their visibility as independent players in the political process, the functions of the parliamentary opposition for the political system as a whole are often more,

rather than less, tangible than that of the majority parliamentary party groups. Most authors consider the functional profile of the parliamentary opposition in parliamentary democracies to include the three tasks of criticising the government, scrutinising and checking governmental actions and policies, and representing a credible 'alternative government" (Helms, 2008, 9).

Ministers are their bills' shepherds. To ensure smooth parliamentary passage a minister negotiates with members of the parliamentary opposition, accommodates a reasonable demand of an opposition party or incorporates a sensible idea in exchange for some praise, less public criticism or even additional 'yes' votes (Müller, 1993; Müller et al., 2001; Russell and Gover, 2017). Ministers therefore usually refrain from issuing strong rhetorical attacks on the opposition. Such a task is more appropriate for the leader of the parliamentary party group or delegated to rhetorically talented MPs.

Some types of parliamentary debate are more confrontational by nature than others. Debates on bills vary widely. When a bill is passed by consensus there is no need for criticism by the opposition in the preceding debate. When a minister faces a no-confidence vote introduced by an opposition party the debate will be usually heated. A debate on a topic that an opposition party can impose on the government on short notice, as through the instrument of the Urgent Question Debate in the Austrian parliament, is also among the more confrontational ones. In the Urgent Question Debate a minister is forced to address an opposition party's criticism of his or her actions with a minimum time of preparation.

The typical roles of government ministers and MPs from government and opposition parties and the different types of parliamentary debates produce systematic variation that we will use to corroborate the external validity of our sentiment analysis procedure. We posit several

hypotheses for these patterns that an observer of parliamentary politics should consider non-controversial or even 'self-evident' statements.

We posit the following three hypotheses:

• Hypothesis 1: Speakers from government parties exhibit less negativity than speakers from opposition parties.

• Hypothesis 2: Parliamentary party group leaders are most likely to use negative statements, followed by ordinary MPs. Cabinet members are least likely to use negative statements.

• Hypothesis 3: Urgent Question Debates exhibit higher levels of negativity than other parliamentary debates.

We validate these by applying our sentiment analysis approach to Austrian parliamentary speeches from 1996 to 2013. We include all speeches of MPs from the four parties that were constantly present in parliament during that period.

H1 Government vs. Opposition. The first hypothesis expects speakers from government parties to exhibit less negativity than speakers from opposition parties. Figure 4 shows the negativity levels of the four parties that were present over the whole period. The dashed line indicates the overall trend for all parties. The negativity values are averaged per year and per party. Light gray lines indicate parties that remained either in government or in opposition during the entire period of our study. These parties are the ÖVP which was always part of the government and the Greens which remained in opposition. Parties that changed from opposition to government or the other way around during the period are visualized in darker gray. SPÖ switched from government to opposition and back to the government. The FPÖ followed the reverse pattern: opposition, government, back to opposition. The government

coalitions are indicated at the bottom of the figure, each cabinet is further separated by a horizontal line. The BZÖ took part in one government coalition, but is excluded from this Figure as it was only in parliament during a limited period.

With respect to our first hypothesis, the dashed trend line of all parties clearly indicates a party's status as being in (below the dashed trend line) or out (above the dashed line) of the ruling government coalition. The Greens (always in opposition) are above the trend line during the entire period and thus constantly exhibit a higher level of negativity. By contrast, the ÖVP (always in government) is constantly below the trend line with a lower level of negativity. The SPÖ was in government at the beginning and the end of our study. Its' negativity level is similar to the ÖVP when in government, but the party's negativity increases sharply when the SPÖ became an opposition party in 2000. The Freedom party (FPÖ) exhibits the inverse trend: as opposition party, its' negativity level is clearly above the trend line, similar to the Greens. The party apparently changed its rhetorical style once it became a government party (2000-2006). The premature ending of the Freedom Party's coalition with the ÖVP is followed by a substantive increase of negativity. Thus, there is support for our first hypothesis.

[Figure 4 about here]

H2 Parliamentary Roles. Drawing on the differences of their political roles, Hypothesis 2 expects that parliamentary party group leaders are most likely to use negative statements, followed by ordinary MPs. Cabinet members are least likely to use negative statements. The period from 1996 to 2013 covers five legislative terms (five elections) in the Austrian national parliament. Figure 5 illustrates role-based differences for each of these legislative terms. The bar chart indicates mean negativity levels for all groups per term (20th - 24th legislative period). We observe a robust pattern: Parliamentary party group leaders dole out stronger

attacks than the ordinary Members of Parliament. Cabinet members exhibit even more rhetorical restraint.

[Figure 5 about here]

H3 Urgent Question Debates. Hypothesis 3 expects that negativity in Urgent Question Debates is higher than in other parliamentary debates. An opposition party typically requests an Urgent Question Debate to jump a surprising attack on the government or a government minister on a current topic, which leads us to expect that they should exhibit stronger levels of negativity than other plenary debates. Figure 6 shows the negativity level of debates following Urgent Questions compared to all other types of debates based on mean values per legislative term. Although differences are small, Urgent Question Debates on average exhibit a higher level of negativity compared to all other debates taken together.

[Figure 6 about here]

Conclusions

The use of word embeddings introduces a new approach to the field of sentiment analysis in the social sciences that offers potential to improve on current bag-of-words approaches. The major advantage of using word embeddings is their potential to detect and classify unseen or out-of-context words that are not included in the training data. Drawing on vector representations of text that allocate similar words closer to each other, such approaches are able to supplement training data, which may improve the results of machine learning tasks. Social scientists increasingly turn to machine learning for sentiment analysis (Van Atteveldt et al., 2008; Hopkins and King, 2010; Burscher et al., 2014; Ceron et al., 2016). As training

data for these applications is typically scarce, word embeddings have the potential to facilitate applications of machine learning in the discipline.

Our validation on previously unseen training examples shows that word embeddings may improve results obtained from bag-of-words classifiers. For a difficult 'three classes of negativity' prediction task, word embeddings have a higher accuracy level than traditional bag-of-words approaches. The results also indicate that word embeddings seem to learn a more realistic class distribution than bag-of-words classifiers. Comparing the two classifiers shows the potentials of word embeddings and neural networks for text classification.

An empirical application measures negative sentiment in parliamentary debates. Using non-controversial hypotheses on patterns of negativity, our findings provide external validity for the word embeddings approach. In line with our expectations, we find that (1) opposition parties exhibit higher levels of negativity than government parties, (2) the negativity levels of speakers in parliamentary debates are consistent with their political roles, and (3) Urgent Question Debates exhibit higher levels of negativity than other parliamentary debates.

We identify some technical limitations and avenues for future improvements.

Preprocessing of German. Tools available for natural language processing of German language texts are less developed than tools available for the English language. Compound words which are a characteristic of the German language are more difficult to handle with word embedding corpora that offer a limited amount of embeddings. There is a method for splitting compound words into their single components by translating them into another language and the resulting single words back into German (Fritzinger and Fraser, 2010). The preprocessing unit of our pipeline could be extended with such an application to achieve a higher coverage of word embeddings per sentence.

Word and Sentence Embeddings and Classification. The introduction of distributed word embeddings (Mikolov, Chen, et al., 2013; Mikolov, Sutskever, et al., 2013) had a major impact on the field of natural language processing. Several approaches (Le and Mikolov, 2014; Parikh et al., 2016) build on the concept of distributed word embeddings but use more sophisticated modeling techniques than a simple averaging of 'standard' word embeddings. Future work could integrate advanced embedding approaches for modeling text documents or test word embeddings that focus on the representation of sentiment-specific features (Tang et al., 2014). Similarly, using context specific word embeddings, for example from annotated parliamentary speeches (Rauh et al., 2017) could further enhance the performance of our approach. The same applies to other neural network types like convolutional neural networks that can make use of the spatial structure of words within a sentence

Visualization. The last part of our pipeline covers the aggregation and visualization of sentiment scores according to additional structured information extending the application data (such as date, time, topic, politician, party, gender or age). We plan to implement more sophisticated text visualizations for a deeper exploration of the Austrian parliamentary debates in the future. There are multiple tools that show advanced text visualization techniques with a focus on sentiment (Diakopoulos et al., 2010; Gold et al., 2015; Gregory et al., 2006).

Web Interface. A web application that will offer end user access to the implementation of our sentiment procedure is currently under development. It will provide a graphical user interface for our pipeline. With a focus on user interaction and active learning it will support users without deeper technical background in performing machine learning on textual data. Our goal is to enable communication researchers and practitioners to apply a word embedding based sentiment analysis to their own data sets more easily.

Notes

[1] The Austrian parliament carefully revises the transcripts of speeches which then are very similar to written text.

[2] We present a translated version of the coding instructions in Appendix A.

[3] Most existing labeled data sets are in English and not in the domain of political communication. Examples include Kotzias (2015): https://archive.ics.uci.edu/ml/datasets/Sentiment+Labelled+Sentences, Socher et al. (2013): https://nlp.stanford.edu/sentiment/code.html, Maas et al. (2011): http://www.cs.jhu.edu/~mdredze/datasets/sentiment/ or the 'classic' Pang and Lee (2002) dataset: http://www.cs.cornell.edu/people/pabo/movie-review-data/.

[4] Some cases require language-dependent preprocessing prior to sentence tokenization. In our case, without custom preprocessing Polyglot would have wrongly identified academic titles which are unique to Austria.

References

Aaldering, L., and Vliegenthart, R. (2016). Political leaders and the media. Can we measure political leadership images in newspapers using computer-assisted content analysis? *Quality* and *Quantity*, *50*(5), 1871-1905.

Al-Rfou, R., Perozzi, B., and Skiena, S. (2013, August). Polyglot: Distributed word representations for multilingual NLP.In *Proceedings of the seventeenth conference on computational natural language learning* (pp. 183–192). Sofia, Bulgaria: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/W13-3520

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research*, *3*(Jan), 993–1022.

Baumeister, R.A., Bratlavsky, E., Finkenauer, C. (2001). Bad Is Stronger Than Good. *Review of General Psychology* 5(4), 323-370.

Benoit, K., Conway, D., Lauderdale, B., Laver, M., Mikhaylov, S. (2016). Crowd-sourced text analysis: Reproducible and agile production of political data. *American Political Science Review* 110(2): 278-295.

Boumans, J. W., and Trilling, D. (2016). Taking stock of the toolkit: An overview of relevant automated content analysis approaches and techniques for digital journalism scholars. *Digital Journalism*, *4*(1), 8–23.

Burscher, B., Odijk, D., Vliegenthart, R., De Rijke, M., and De Vreese, C. H. (2014). Teaching the computer to code frames in news: Comparing two supervised machine learning approaches to frame analysis. *Communication Methods and Measures*, *8*(3), 190–206.

Ceron, A., Curini, L., and Iacus, S. M. (2015). Using sentiment analysis to monitor electoral campaigns: Method matters-evidence from the United States and Italy. *Social Science Computer Review*, *33*(1), 3–20.

Ceron, A., Curini, L., and Iacus, S. M. (2016). First-and second-level agenda setting in the twittersphere: An application to the Italian political debate. *Journal of Information Technology* and *Politics*, *13*(2), 159–174.

Ceron, A., Curini, L., and Iacus, S. M. (2017). *Politics and big data: Nowcasting and forecasting elections with social media*. Routledge: London.

Chollet, F. (2015). *Keras.* Retrieved from https://github.com/fchollet/keras

Diakopoulos, N., Naaman, M., and Kivran-Swaine, F. (2010). Diamonds in the rough: Social media visual analytics for journalistic inquiry. In *2010 IEEE symposium on Visual Analytics Science and Technology (VAST)* (pp. 115–122).

Fayyad, U., Piatetsky-Shapiro, G., and Smyth, P. (1996). From data mining to knowledge discovery in databases. *AI Magazine*, *17*(3), 37.

Fritzinger, F., and Fraser, A. (2010). 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 Metrics MATR* (pp. 224–234).

Gold, V., Rohrdantz, C., and El-Assady, M. (2015). Exploratory Text Analysis using Lexical Episode Plots. In E. Bertini, J. Kennedy, and E. Puppo (Eds.), *Eurographics Conference on Visualization (EuroVis) - short papers.* The Eurographics Association. doi: 10.2312/eurovisshort.20151130

Goldberg, Y. (2016). A Primer on Neural Network Models for Natural Language Processing. *Journal of Artificial Intelligence Research*, *57*, 345-420.

Greene, Z., Ceron, A., Schumacher, G., and Fazekas, Z. (2016). The Nuts and Bolts of Automated Text Analysis. Comparing Different Document Pre-Processing Techniques in Four Countries. Retrieved from osf.io/ghxj8

Gregory, M. L., Chinchor, N., Whitney, P., Carter, R., Hetzler, E., and Turner, A. (2006). User-directed sentiment analysis: Visualizing the affective content of documents. In *Proceedings of the workshop on sentiment and subjectivity in text* (pp. 23–30).

Grimmer, J., and Stewart, B. M. (2013). Text as data: The promise and pitfalls of automatic content analysis methods for political texts. *Political Analysis, 21*(3), 267-297. Harris, Z. S. (1954). Distributional Structure. *WORD*, *10*(2-3), 146–162.

Haselmayer, M., and Jenny, M. (2017). Sentiment analysis of political communication: combining a dictionary approach with crowdcoding. *Quality* and *Quantity, 51(6): 2623-2646*.

Haselmayer, M., Hirsch L., and Jenny, M. (2017). Love is blind. Partisan bias in the perception of positive and negative campaign messages. Paper prepared for presentation at the 7th Annual Conference of the European Political Science Association (EPSA), June 22-24, Milan.

Helms, Ludger (2008). Studying Parliamentary Opposition in Old and New Democracies: Issues and Perspectives. *The Journal of Legislative Studies* 14(1-2), 6-19.

Hopkins, D. J., and King, G. (2010). A Method of Automated Nonparametric Content Analysis for Social Science. *American Journal of Political Science, 54*(1), 229–247.

Kandel, S., Heer, J., Plaisant, C., Kennedy, J., van Ham, F., Riche, N. H., ... Buono, P. (2011). Research directions in data wrangling: Visualizations and transformations for usable and credible data. *Information Visualization*, *10*(4), 271–288.

Kingma, D., and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kleinnijenhuis, J., Schultz, F., Oegema, D., and van Atteveldt, W. (2013). Financial news and market panics in the age of high-frequency sentiment trading algorithms. *Journalism, 14*(2), 271-291.

Kotzias, D., Denil, M., De Freitas, N., and Smyth, P. (2015, August). From group to individual labels using deep features. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 597-606). ACM.

Krippendorff, Klaus (2013). Content Analysis. An Introduction to its methodology. 3rd edition. Los Angeles: Sage.

Le, Q., and Mikolov, T. (2014). Distributed representations of sentences and documents. In *International Conference on Machine Learning (ICML).*

Lind, F., Gruber, M., and Boomgaarden, H. G. (2017). Content Analysis by the Crowd: Assessing the Usability of Crowdsourcing for Coding Latent Constructs. Communication methods and measures, 11(3), 191-209.

Loper, E., and Bird, S. (2002). NLTK: The natural language toolkit. In *ACL workshop on effective tools and methodologies for teaching natural language processing and computational linguistics.*

Lowe, W., and Benoit, K. (2013). Validating estimates of latent traits from textual data using human judgment as a benchmark. *Political Analysis, 21*(3), 298–313.

Lucas, C., Nielsen, R. A., Roberts, M. E., Stewart, B. M., Storer, A., and Tingley, D. (2015). Computer-assisted text analysis for comparative politics. *Political Analysis, 23*(2), 254-277. doi: 10.1093/pan/mpu019

Maas, A.L., Daly, R.E., Pham, P.T., Huang, D., Ng, A.Y., and Potts, C. (2011). Learning Word Vectors for Sentiment Analysis. *The 49th Annual Meeting of the Association for Computational Linguistics (ACL 2011).*

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *CoRR*. Retrieved from http://arxiv.org/abs/1301.3781

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems 26* (pp. 3111–3119). Curran Associates, Inc.

Moraes, R., Valiati, J. F., and Neto, W. P. G. (2013). Document-level sentiment classification: An empirical comparison between SVM and ANN. *Expert Systems with Applications*, *40*(2), 621–633.

Mozetič, I., Grčar, M., and Smailovič, J. (2016). Multilingual twitter sentiment classification: The role of human annotators.´ *PloS one*, *11*(5), e0155036.

Müller, W.C. (1993). Executive–Legislative Relations in Austria: 1945– 1992. *Legislative Studies Quarterly*, 18(4), 467–494.

Müller, W.C., Jenny, M., Dolezal, M., Steininger, B., Philipp, W. and Westphal, S. (2001). *Die österreichischen Abgeordneten: Individuelle Präferenzen und politisches Verhalten*. Wien: WUV Universitätsverlag.

Nasukawa, T., and Yi, J. (2003). Sentiment analysis: Capturing favorability using natural language processing. In *Proceedings of the 2nd international conference on knowledge capture* (pp. 70–77).

Nobata, C., Tetreault, J., Thomas, A., Mehdad, Y., and Chang, Y. (2016). Abusive language detection in online user content. In *Proceedings of the 25th international conference on world wide web* (pp. 145–153).

Pang, B., Lee, L., and Vaithyanathan, S. (2002). 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*, 79–86.

Parikh, A. P., Täckström, O., Das, D., and Uszkoreit, J. (2016). A decomposable attention model for natural language inference. *arXiv preprint arXiv:1606.01933*.

Parliament Austria (2013). *Parliamentary speeches from the austrian national parliament.* Retrieved from https://www.parlament.gv.at/PERK/NRBRBV/NR/STENO/

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the Empirical Methods in Natural Language Processing (EMNLP 2014)* (Vol. 14, pp. 1532–1543). Retrieved from https://nlp.stanford.edu/projects/glove/

Rheault L, Beelen K, Cochrane C and Hirst G (2016). *Measuring Emotion in Parliamentary Debates with Automated Textual Analysis. PLoS one* 11(12), e0168843.

Raschka, S. (2014). Naive Bayes and Text Classification I - Introduction and Theory. arXiv preprint arXiv:1410.5329

Rauh, C., De Wilde P and Schwalbach J (2017) The ParlSpeech data set: Annotated full-text vectors of 3.9 million plenary speeches in the key legislative chambers of seven European states , Harvard Dataverse, V1, http://dx.doi.org/10.7910/DVN/E4RSP9

Rozin, P. and Royzman, E.B. (2001). Negativity Bias, Negativity Dominance, and Contagion. *Personality and Social Psychology Review* 5(4), 296-320.

Russell, M. and Gover, D (2017). *Legislation at Westminster: Parliamentary Actors and Influence in the Making of British Law*. Oxford: Oxford University Press.

Salton, G. and McGill, M. J. (1986). *Introduction to modern information retrieval*. McGraw-Hill, Inc.

Slapin, J.B. and Proksch, O. (2014). Words as Data: Content Analysis in Legislative Studies. In Sh. Martin, K. Strom and Th. Saalfeld (Eds.). *The Oxford Handbook of Legislative Studies*. Oxford: Oxford University Press.

Socher, R., Perelygin, A., Wu, J. Y., Chuang, J., Manning, C. D., Ng, A. Y., and Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)* (Vol. 1631, p. 1642).

Soroka, S., and McAdams, S. (2015). News, politics, and negativity. *Political Communication*, *32*(1), 1–22.

Soroka, S., Young, L., and Balmas, M. (2015). Bad news or mad news? Sentiment scoring of negativity, fear, and anger in news content. *The Annals of the American Academy of Political and Social Science*, *659*(1), 108–121.

Stolte, C., Tang, D., and Hanrahan, P. (2002). Polaris: A system for query, analysis, and visualization of multidimensional relational databases. *IEEE Transactions on Visualization and Computer Graphics*, *8*(1), 52–65.

Tang, D., Wei, F., Yang, N., Zhou, M., Liu, T., and Qin, B. (2014). Learning sentiment-specific word embedding for twitter sentiment classification. In *ACL (1)* (pp. 1555–1565).

Turney, P. D., and Pantel, P. (2010). From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research*, *37*, 141–188.

van Atteveldt, W., Kleinnijenhuis, J., and Ruigrok, N. (2008). Parsing, semantic networks, and political authority using syntactic analysis to extract semantic relations from dutch newspaper articles. *Political Analysis*, *16*(4), 428-446. doi: 10.1093/pan/mpn006

van Atteveldt, W., Kleinnijenhuis, J., Ruigrok, N., and Schlobach, S. (2008). Good news or bad news? Conducting sentiment analysis on dutch text to distinguish between positive and negative relations. *Journal of Information Technology* and *Politics*, *5*(1), 73–94.

van Atteveldt, W., Sheafer, T., Shenhav, S. R., and Fogel-Dror, Y. (2017). Clause analysis: Using syntactic information to automatically extract source, subject, and predicate from texts with an application to the 2008-2009 gaza war. *Political Analysis*, *25*(2), 207-222.

Wilkerson, J., and Casas, A. (2017). Large-scale computerized text analysis in political science: Opportunities and challenges. *Annual Review of Political Science*, *20*(1), 529-544.

Wulczyn, E., Thain, N., and Dixon, L. (2016). Ex machina: Personal attacks seen at scale. *CoRR, abs/1610.08914*. Retrieved from http://arxiv.org/abs/1610.08914

Wueest, B., Clematide, S., Bünzli, A., Laupper, D., and Frey, T. (2011). Electoral campaigns and relation mining: Extracting semantic network data from newspaper articles. *Journal of Information Technology and Politics*, *8*(4), 444-463.

Wueest, B.,Clematide, S., Bünzli, A., Laupper, D., and Frey, T. (2011). Electoral campaigns and relation mining: Extracting semantic network data from newspaper articles. *Journal of Information Technology and Politics*, *8*(4), 444-463.

Young, L., and Soroka, S. (2012). Affective news: The automated coding of sentiment in political texts. *Political Communication*, *29*(2), 205–231.

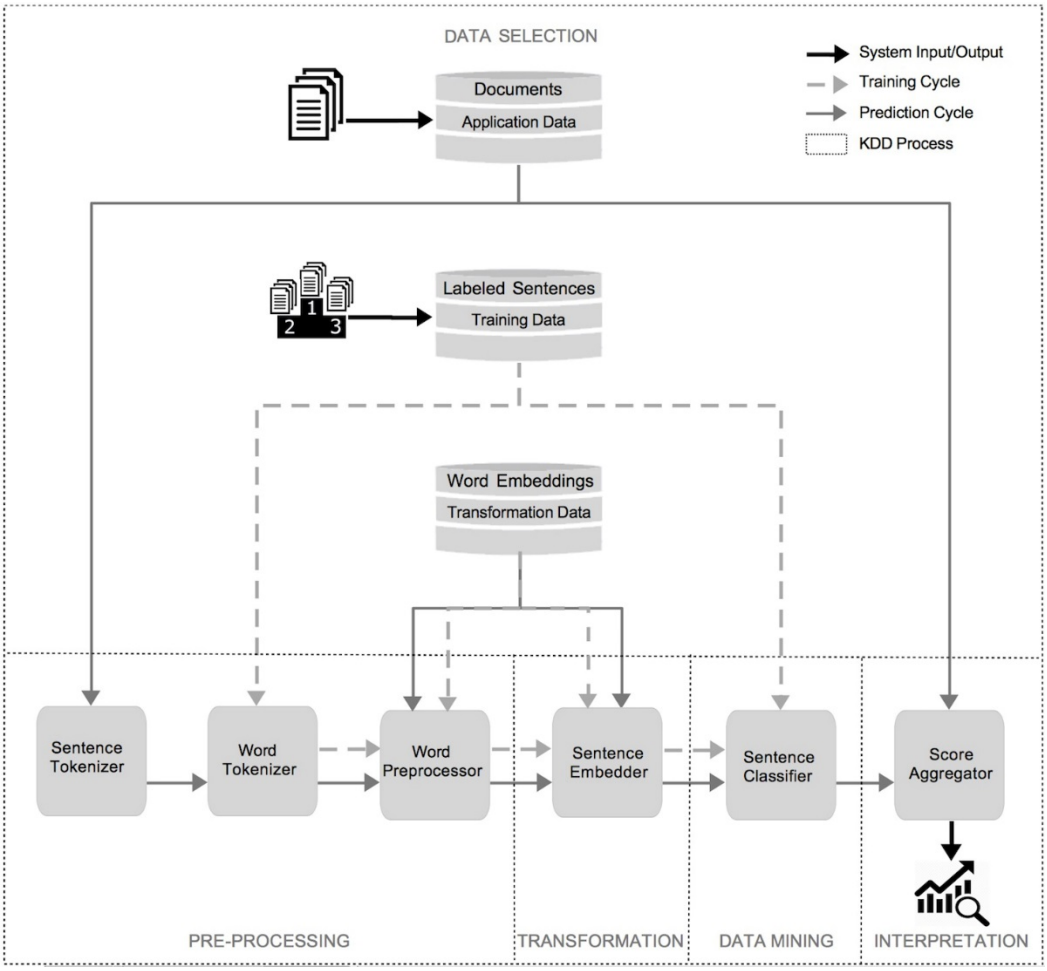

Figure 1: Supervised sentiment analysis approach with distributed word embeddings

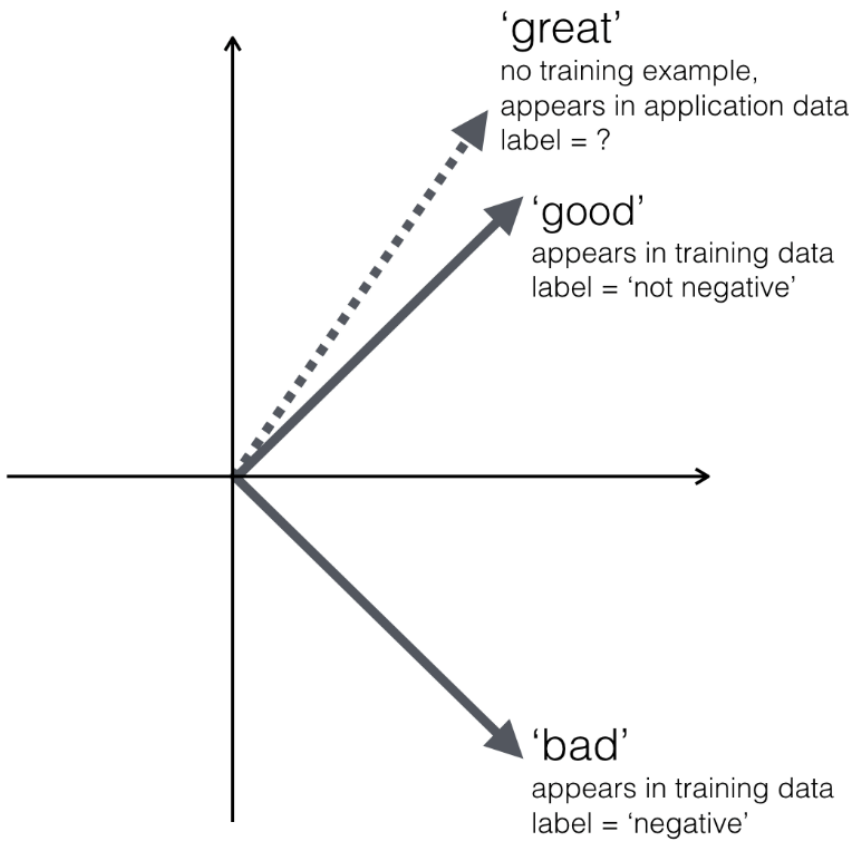

Figure 2: Illustrative example of mapping sentiment to word embedding dimensions

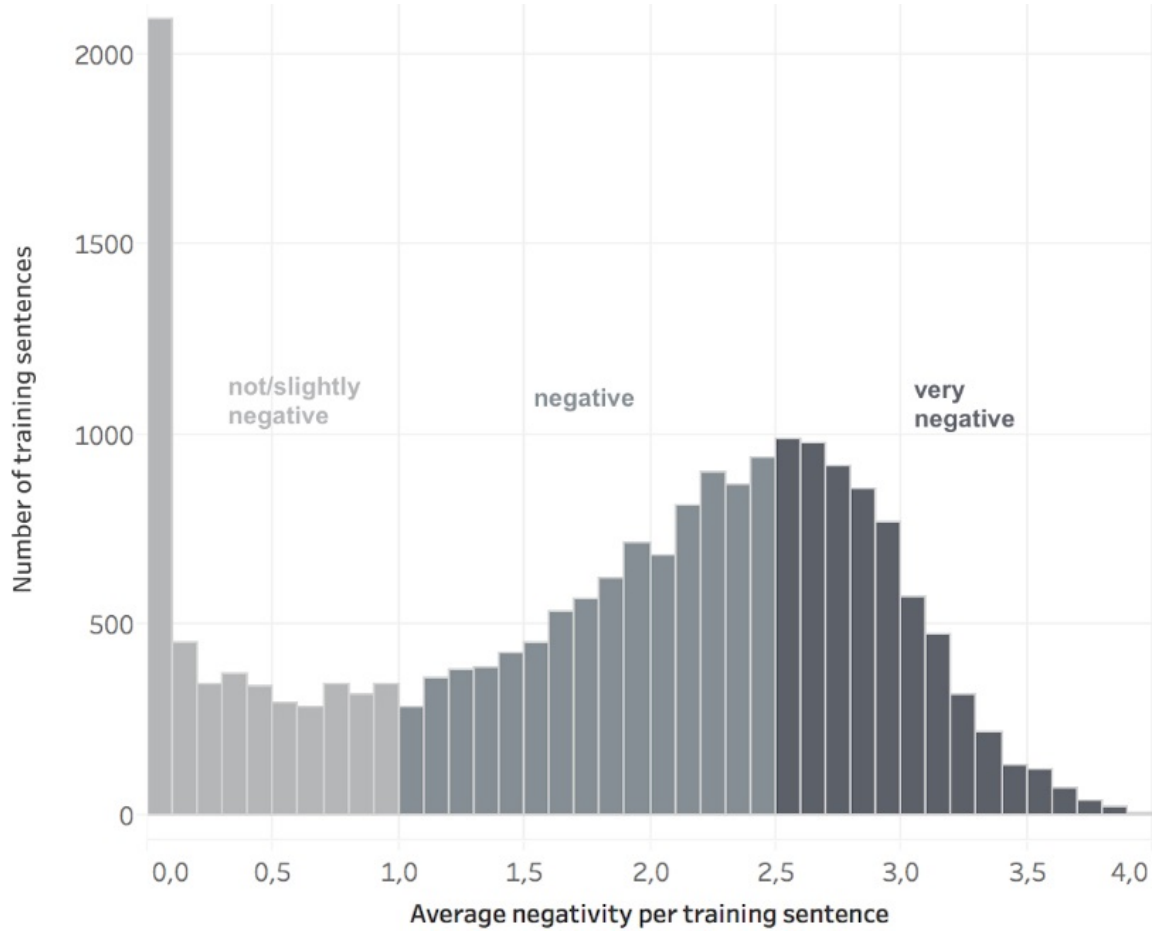

Figure 3: Negativity distribution of 20,600 training sentences ranging from 0 (not negative) to 4 (very negative), divided into three classes: not/slightly negative, negative and very negative

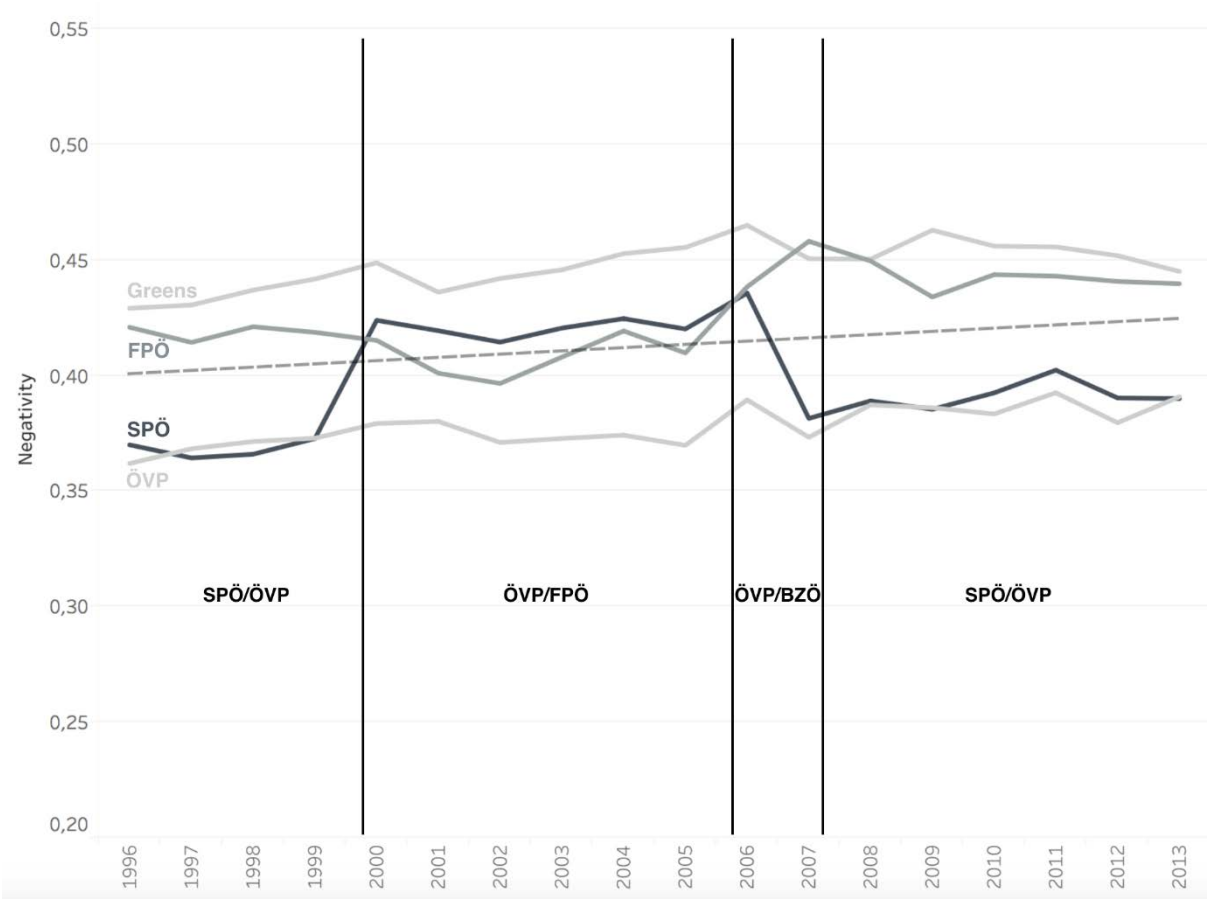

Figure 4: Negativity evolution in the Austrian parliament from 1996 to 2013 showing those four parties that were present over the whole period

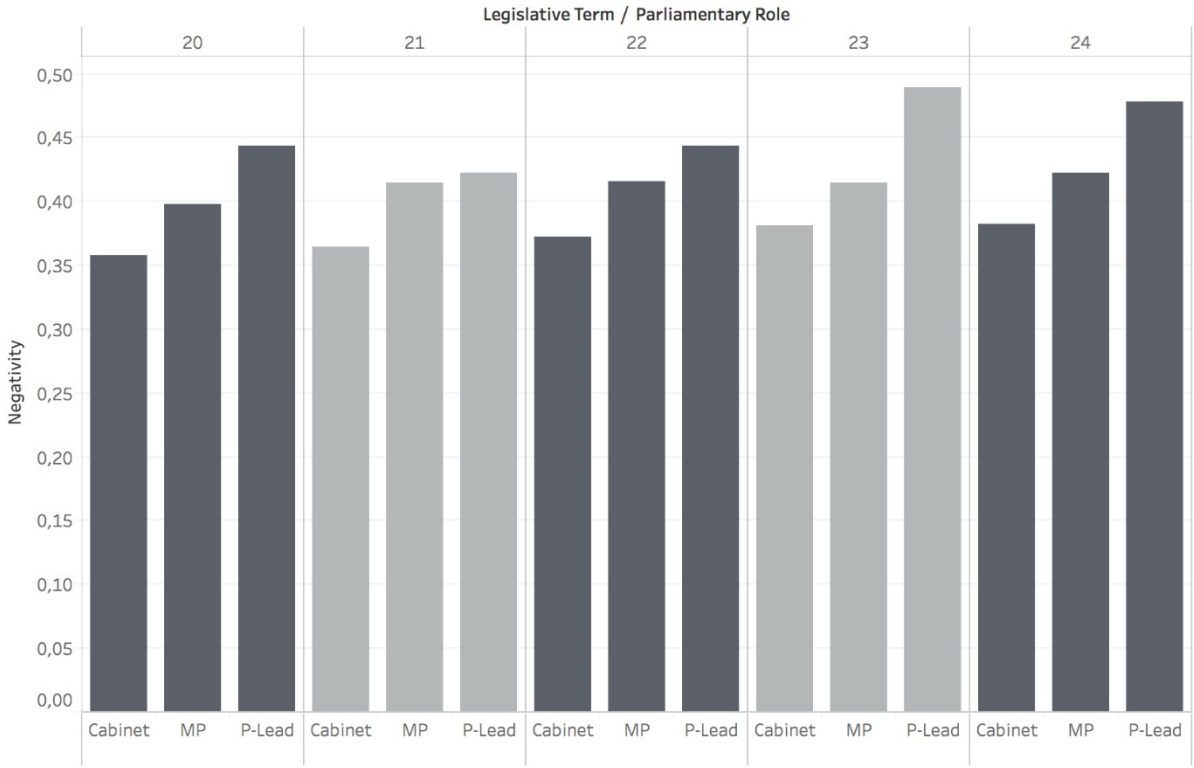

Figure 5: Negativity distinction per legislative term and parliamentary role: Average scores of cabinet members, MPs, and parliamentary party group leaders

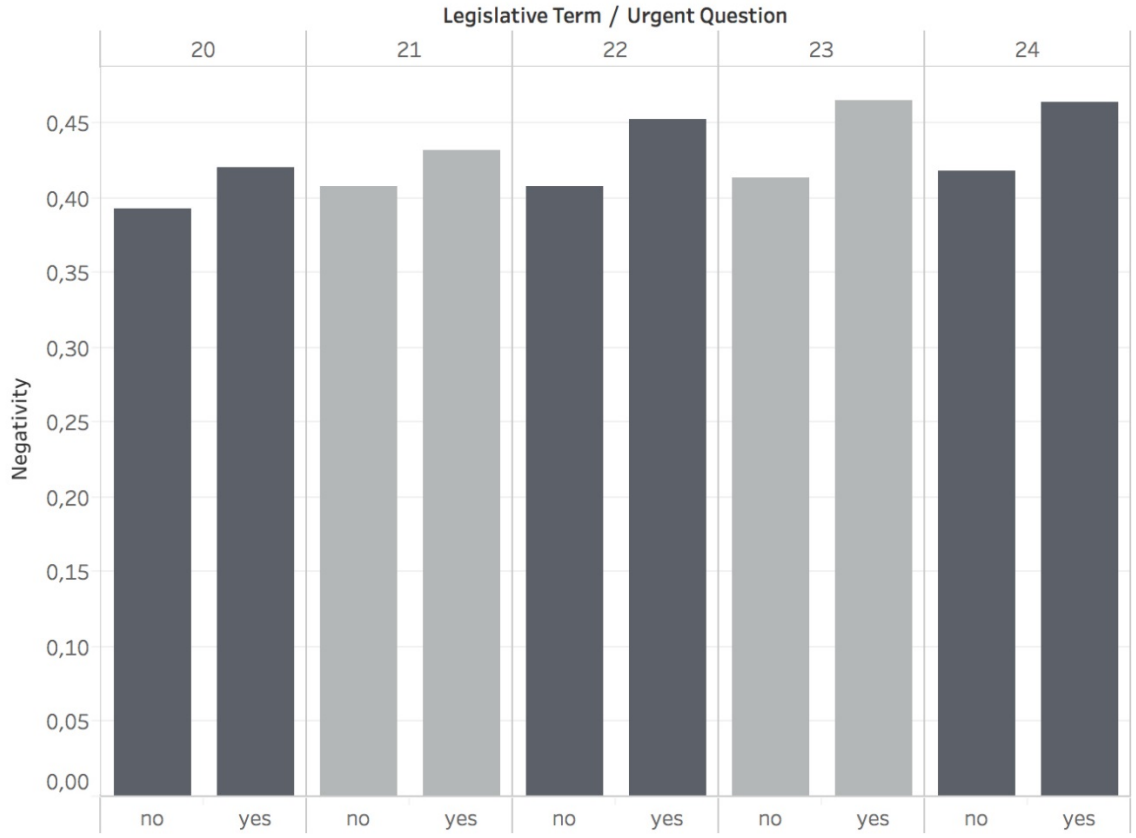

Figure 6: Negativity distinction per legislative term and type of debate: Average scores of Urgent Question

Table 1: Precision, recall and F1 score for the bag of words approach

	Actual	Predicted	Precision	Recall	F1 Score
not/slightly negative	524.3	205.6	0.33	0.83	0.47
negative	805.7	1188.7	0.71	0.48	0.57
very negative	730	665.7	0.53	0.58	0.56

Table 2: Precision, recall and F1 score for the Word Embeddings approach

	Actual	Predicted	Precision	Recall	F1 Score
not/slightly negative	522.4	575	0.65	0.59	0.61
negative	799.2	771.6	0.52	0.53	0.53
very negative	739.4	714.4	0.55	0.57	0.56

Appendix A: CrowdFlower coding instructions (translation)
These coding instructions were pretested by colleagues, student assistants and a few online coders.

How negative are these statements?

What is this about?
We present you sentences from political and media texts. Many, though not all, of these sentences include direct or indirect criticism, allegations or attacks.

Task
Please read each sentence carefully and decide, whether it includes a positive, neutral or negative statement. In a second step, we ask you to rate the intensity of the statement using the following scale:

-Not negative (neutral or positive)
- Very weakly negative
- Weakly negative
- Strongly negative
- Very strongly negative
- Not codable

What should you consider?
Only rate the actual content of the text! Stay impartial, your personal preferences towards persons or organizations should not influence your coding decisions.
Not negative
A sentence should be coded as "not negative" if it contains a neutral or positive statement.
Example "not negative":
"I serve the Austrian citizens with passion and commitment."

Not codable
A sentence is "not codable" if it is incomprehensible or if it does not make any sense to you. Some sentences may be incomplete, as they have been processed automatically. As long as you are able to purposefully decide, whether they are positive, neutral or negative, we ask you to rate them anyhow.
Example "not codable":
"Ic$%$# we retain%, that &%§"*

Negative
Negative sentences contain direct or indirect criticism, allegations or attacks in varying intensity.
Examples with increasing negativity:
"We demand that the government finally delivers a better job!"
"These are bad actions, which come at the expense of the population."
"This minister promotes corruption and consciously dupes the people."
"This is a scam on all of us: the dishonesty of these politicians stinks to high heavens."

Special case: sentences containing specific coding instructions
Some sentences may contain instructions, asking you to choose a specific category. In such cases, you should ignore all other textual information and directly follow the instructions.

Example:
"The government has failed to address these issues in the past legislative term. ***Please ignore the previous part of the text and code this unit as "not codable****"*.
In case of any question regarding the coding process or if you would like to provide us with feedback, please send us an E-Mail: crowdsourcing@autnes.at
Thank you for your contribution!

Appendix B:

Table B1 shows the average confusion matrix for unseen test data of our classifier trained on bag-of-words representations. There is an obvious tendency of this classifier to choose the middle class whereas the word embedding variant shows a clear diagonal in its confusion matrix (Table B2). The bias of the bag-of-words classifier explains its higher accuracy for middle class examples compared to the word embedding approach. The bag-of-words variant nevertheless shows lower accuracies for the two outer classes. Class 2 (very negative) is slightly lower while class 0 (not/slightly negative) is significantly lower compared to our proposed word embedding approach.

Table B1: Average confusion matrix for unseen test data: Bag-of-words

		Predicted		
		not/slightly negative	negative	very negative
Actual	not/slightly negative	170.6	284.6	69.1
	negative	27.3	569.7	208.7
	very negative	7.7	334.4	387.9

Table B2: Average confusion matrix for unseen test data: Word embeddings

		Predicted		
		not/slightly negative	negative	very negative
Actual	not/slightly negative	337.2	120.3	64.9
	negative	146.5	412.7	240.0
	very negative	91.3	238.6	409.5