Text Classification for Monolingual Political Manifestos with Words Out of Vocabulary

Arsenii Rasov[1,*], Ilya Obabkov[1], Eckehard Olbrich[2] and Ivan P. Yamshchikov[2]
[1]*Ural Federal University, Mira Street, 19, Yekaterinburg, Russia*
[2]*Max Planck Institute for Mathematics in the Sciences, Inselstrasse 22, Leipzig, Germany*

Keywords: Electoral Programs, Text Corpus, Classification of Political Texts.

Abstract: In this position paper, we implement an automatic coding algorithm for electoral programs from the Manifesto Project Database. We propose a new approach that works with new words that are out of the training vocabulary, replacing them with the words from training vocabulary that are the closest neighbors in the space of word embeddings. A set of simulations demonstrates that the proposed algorithm shows classification accuracy comparable to the state-of-the-art benchmarks for monolingual multi-label classification. The agreement levels for the algorithm is comparable with manual labeling. The results for a broad set of model hyperparameters are compared to each other.

1 INTRODUCTION

Computational social science is a field that leverages the capacity to collect and analyze data at scale. One hopes that automated data analysis of such data may reveal patterns of individual and group behaviors, (Laser et al. (2009)). Analysis of political discourse is one of the prominent fields where data analysis overlaps with sociology, history, and political science. Scientists study electoral processes, interactions of political actors with one another and with the public. In these works, researchers use different types of data that could describe such processes. However, the demand for well-annotated high-quality datasets is continuously higher than the supply of new data. Political scientists are in general need of annotated datasets to do their work. It can be a document-wise annotation, which matches the whole documents with specific categorical labels, retrieving the document's basic idea, or sentence-wise labeling, that matches each sentence with a particular label.

There are many widely-used sources, which provide different types of political data. Some researchers use data from social networks, such as twitter[1]. For those who are interested in parliament debates, there are such projects as EuroParl corpus (Koehn(2005)), Linked EP[2], ConVote dataset (Gentzkow et al. (2018)).

One of the most popular corpora in political science is the Manifesto Project (Lehmann et al. (2018)). It is a large human-annotated, open-access, cross-national text corpus that consists of electoral programs. Here, the experts implemented human annotation (or the so-called "coding") based on the content analysis of electoral programs. The sentences are divided into statements (quasi-sentences). Each sentence is coded with one of 57 categories (which, in turn, form 7 broad topics). Currently, the corpus includes more than 2300 machine-readable documents, more than 1150 of them are coded already. There are about 1 000 000 coded quasi-sentences in the corpus (Volkens et al. (2015)).

The annotation process in Manifesto is a very challenging task. It is carried out by the groups of experts, specially trained to perform such labeling. This process is a very time-consuming, labor-intensive, and expensive procedure. Moreover, it is not a trivial task to label each quasi-sentence with only one of 57 categories; indeed, the level of agreement of the experts is only about 50% (Mikhailov et al.(2012)).

One way to overcome those challenges is to use algorithms of supervised machine learning for quasi-sentence classification. For a long time, text classification was perceived as a monolingual task. However,

*This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 732942.

[1]UK MPs: https://twitter.com/twittergov/lists/uk-mps

[2]https://linkedpolitics.project.cwi.nl/web/html/home.html

Rasov, A., Obabkov, I., Olbrich, E. and Yamshchikov, I.
Text Classification for Monolingual Political Manifestos with Words Out of Vocabulary.
DOI: 10.5220/0009792101490154
In *Proceedings of the 5th International Conference on Complexity, Future Information Systems and Risk (COMPLEXIS 2020)*, pages 149-154
ISBN: 978-989-758-427-5

there are more and more recent results that treat it as a multilingual one. Monolingual methods are trained on the data represented in one language only. Such methods could be used when there is enough monolingual data for training. Naturally, since some languages can be scarcely represented, one would like to use multilingual methods and transfer some information learned in the data-rich languages to the more challenging ones. Such methods are called multilingual. The majority of these methods are based on the idea that one can construct specific semantic space and embed texts in different languages into one shared space. Training language-specific embedder algorithms, in this case, could have nothing to do with the classification task per se. However, the joint multilingual embedding space equipped with a particular measure of semantic similarity could be used for classification purposes. Such methods are harder to train but can be useful if we are interested in the languages that are underrepresented in the training data. In the next section, we briefly review the latest multi- and monolingual results relevant to our project.

2 STATE OF THE ART

There are several baselines for the classification of Manifesto texts that vary across different formulations of the classification task. One could split the works in this area into two huge sections: the researchers that build algorithms for seven coarse high-level topics and the researcher that classify individual labels.

For seven high-level topics classification, one of the baselines is the paper (Glavas et al.(2017)). Here authors implement multilingual text classification using convolutional neural networks to match the sentence for a given manifesto with seven coarse-grained classes. They outperform the state-of-the-art for Italian, French, and English languages. In a monolingual setting (Zirn et al.(2016)) present the method, which combines the topic-classification method and topic-shift method using the Markov logic network for seven sparse-level categories, reaching 74.9% of macro-average F1-score.

The classification process for individual labels is more challenging due to the lack of data for training and the sheer fact that it is typically harder to build classification algorithms with more categories. (Subramanian et al. (2017)) use a joint sentence-document model for both sentence-level and document-level classification. They propose the neural multilingual network-based approach for fine-grained sentence classification and demonstrate the state-of-the-art quality for different languages. In (Subramanian et al. (2018)), authors improve their performance using a hierarchical bidirectional LSTM approach.

The current state-of-the-art benchmark for the Manifesto quasi-sentence classification on 57 fine-grained labels is presented in (Merz et al. (2016)). The authors describe the approach of monolingual text classification, using the SVM algorithm. They show 42% accuracy for German manifestos.

In this research, we propose our method that outperforms the (Merz et al. (2016)) benchmark for 57 labels. We also modify the experimental conditions to make it more similar to the real conditions and address the out-of-vocabulary words problem that we describe in detail further.

3 CLASSIFICATION

Since some labels are under-represented in the training sample, it is hard to balance the training, and it is futile to expect that a multi-parameter model such as a deep neural network could be trained on such scarce data in a monolingual setting. In this work, we suggest focusing on basic machine learning methods that are robust under the variation of the training categories sizes. Further, we experiment with support vector machines (Vapnik, V. (1998)) and gradient boosting (Chen, T. and Guestrin, C. (2016)).

3.1 Training

We perform the following preprocessing of the manifesto data. We remove punctuation, split all sentences into the lists of separated words, and remove stopwords.

We train a tf-icf matrix to vectorize each word in a semi-sentence. Tf-icf is a supervised version of tf-idf, which includes supervised term-weighting, see (Lan et al. (2009)). In tf-icf scheme, we build the term-category matrix instead of the term-document one. To do that, we join all semi-sentences of each class in separate new documents and train tf-icf matrix on them.

Then for each sentence, we create weighted one-hot vectors, using a scheme, proposed at (Merz et al. (2016)): we take a sum of weighted one-hot vectors of the target sentence (weighted by 1/2) and vectors of 4 nearest sentences (weighted by 1/3 and 1/6 w.r.t. distance to the target sentence). In this work, we also experiment with the different sizes of such kernel: three, five, and seven sentences. After the multiplication of such vectors by the tf-icf matrix we receive

Table 1: Various results for English, German and Spanish. A longer window of seven sentences seems to yield better results. Unigram-based method outperforms bigrams in Spanish.

Kernel size	N-gram range	Metric	Language		
			Eng	*Ger*	*Span*
7	*1*	*accuracy*	**0.485**	**0.438**	0.461
		correlation	0.876	0.891	0.617
	2	*accuracy*	0.484	0.437	0.453
		correlation	0.875	0.890	0.601
5	*1*	*accuracy*	0.471	0.425	**0.468**
		correlation	0.863	0.890	**0.690**
	2	*accuracy*	0.481	0.431	0.450
		correlation	**0.878**	**0.892**	0.605
3	*1*	*accuracy*	0.468	0.409	0.446
		correlation	0.878	0.892	0.636
	1	*accuracy*	0.468	0.408	0.438
		correlation	0.878	0.891	0.616

Table 2: Various results for English, German, and Spanish without out-of-vocabulary words. Bigrams with longer window kernel demonstrate higher accuracy across all languages.

Kernel size	N-gram range	Metric	Language		
			Eng	*Ger*	*Span*
7	*1*	*accuracy*	**0.430**	**0.368**	0.434
		correlation	0.866	0.878	0.604
	2	*accuracy*	**0.430**	**0.368**	**0.435**
		correlation	0.866	0.877	0. 606
5	*1*	*accuracy*	0.427	0.364	0.430
		correlation	0.866	**0.880**	0.611
	2	*accuracy*	0.427	0.364	0.430
		correlation	**0.867**	**0.880**	0.611
3	*1*	*accuracy*	0.416	0.345	0.418
		correlation	**0.867**	0.878	0.638
	2	*accuracy*	0.416	0.354	0.418
		correlation	**0.867**	0.878	**0.643**

57-dimensional vectors. We also experiment with different sizes of n-grams (uni- and bigrams). This way, one could hope to retrieve more information about the context, taking into consideration more than one word as a bit of meaningful information.

Finally, we train a machine learning algorithm using the obtained matrix as input and labels as a target.

3.2 Reproducing Experiments

(Merz et al. (2016)) also use the supervised version of tf-icf vectorization. In the experiments, authors train the final tf-icf-based matrix on the whole dataset, including the train and the test parts. Then they train the ML algorithm on the train part of the dataset and benchmark it on the test set. Here we first reproduce that experiment with various parameters.

Since the data in Manifesto dataset is historical, it makes sense to train the algorithm on older documents and test the resulting quality on newer ones. Here we use the documents of the most recent year in the dataset as a test set. These would be the year 2017 for German, and 2016 for English and Spanish.

We use accuracy as a quality metric for our experiments. It is analogous to the agreement level for human coders and provides the possibility to compare the classification quality to the human' annotation. As another quality metric, we use the document-wise Pearson correlation between human-annotated categories and algorithm-annotated ones, proposed kin (Merz et al. (2016)). This metric helps to estimate the similarity of code assignment at the aggregate level. The results of the experiments are shown in Table 1.

Figure 1 shows scatter-plots for the frequencies of all manually assigned categories versus automatically assigned ones. The plots are drawn for the best performing models in English, German, and Spanish, respectively.

For German and English, the highest agreement with human annotators is achieved when including bigrams to the tf.icf vocabulary. The accuracy and the correlation score for German texts outperforms the state-of-the-art one (0.42 and 0.88, (Merz et al. (2016))). The accuracy for English and Spanish languages are comparable to the state-of-the-art models.

3.3 Out-of-Vocabulary Words

Due to the supervised nature of the tf-icf algorithm, it is fair to say that in real-life conditions, one does not have the annotation to the new historical data. One has to classify these new data as it arrives. That means that the method described above could only be partially reproduced: one can not build a complete tf-icf matrix that would include every word in the new data, since some of the words may not occur in the training dataset. These words out of vocabulary constitute a significant portion of the vocabulary that can not be ignored. If we use the latest datasets for the test, there would be 3485, 3266, and 8018 out-of-vocabulary (O-o-V) words for German, English, and Spanish datasets, respectively. Table 2 shows that if one initializes O-o-V words with zeros, it drastically reduces the quality of the classification.

One should notice here that without any information on the out-pf-vocabulary words, the best accuracy is achieved on a bigger kernel with bigrams. This stands to reason: due to the absence of information on new words that were not observed in the training set, the model needs to rely on a broader context to achieve higher accuracy. Table 3 compares the accuracy for the model with a full if-icf matrix (with

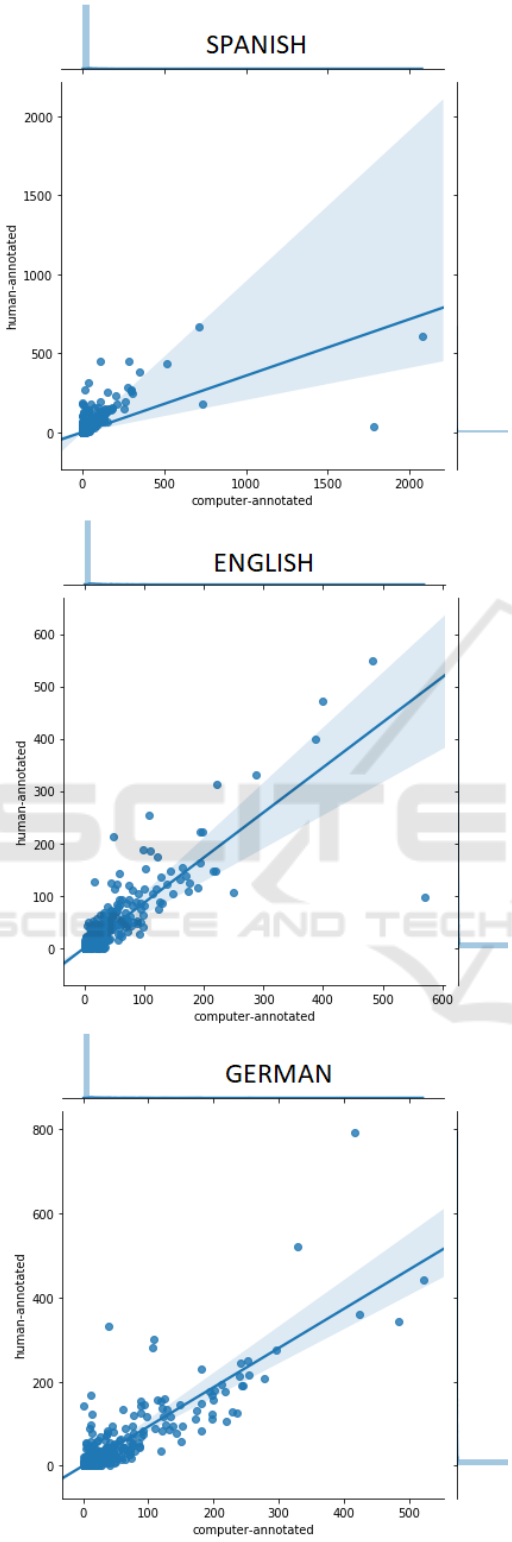

Figure 1: Comparison of code frequencies of 57 categories, trained on the whole dataset, in six electoral programs by human and semi-automatic coding for Spanish, English and German texts respectively.

bigrams and kernel size 7) and the same model but without information on the O-o-V words. There is a drastic decline in accuracy for all three languages.

Table 3: Overview of the change in accuracy for the algorithm that does not use words out-of-vocabulary.

Language	Accuracy with full tf-icf matrix	Drop in accuracy without words O-o-V
English	0.485	−11.3%
German	0.436	−15.6%
Spanish	0.461	−5.6%

To overcome this problem, we propose a specific method of word replacement, based on the FastText word embeddings (Bojanovski et al. (2016)). One can use pre-trained Wikipedia FastText vectors (Verberne et al. (2018)) for all of the words in our dataset and replace out-of-vocabulary words with the closest ones from the training set, using cosine distance between FastText vectors as a distance metric. This manipulation helps to keep part of the information that comes with the out-of-vocabulary words intact during the vectorization process. Table 4 shows the results for various parameters of the algorithm across all three languages.

However, again the best accuracy is achieved using bigrams and the kernel of size 7 for all languages.

Figure 2 shows scatter-plots for the frequencies of all manually assigned categories versus automatically assigned ones. The plots are drawn for the best performing models in English, German, and Spanish, respectively.

Table5 shows relative accuracy improvement when O-o-V words are substituted with their nearest neighbors in the FastText embeddings.

Looking at Table 5, one can see that replacing the out-of-vocabulary words with their nearest FastText

Table 4: Various results for English, German and Spanish with out-of-vocabulary words replacements. Bi-grams with longer window kernel demonstrate higher accuracy across all languages.

Kernel size	N-gram range	Metric	Language		
			Eng	*Ger*	*Span*
7	*1*	*accuracy*	**0.426**	**0.371**	0.448
		correlation	**0.860**	0.879	0.646
	2	*accuracy*	**0.426**	**0.371**	**0.449**
		correlation	0.858	0.879	0. 648
5	*1*	*accuracy*	0.423	0.368	0.444
		correlation	0.858	0.878	0.649
	2	*accuracy*	0.424	0.368	0.444
		correlation	0.858	0.879	0.651
3	*1*	*accuracy*	0.412	0.351	0.431
		correlation	0.859	**0.880**	**0.685**
	2	*accuracy*	0.413	0.352	0.432
		correlation	0.859	0.879	**0.685**

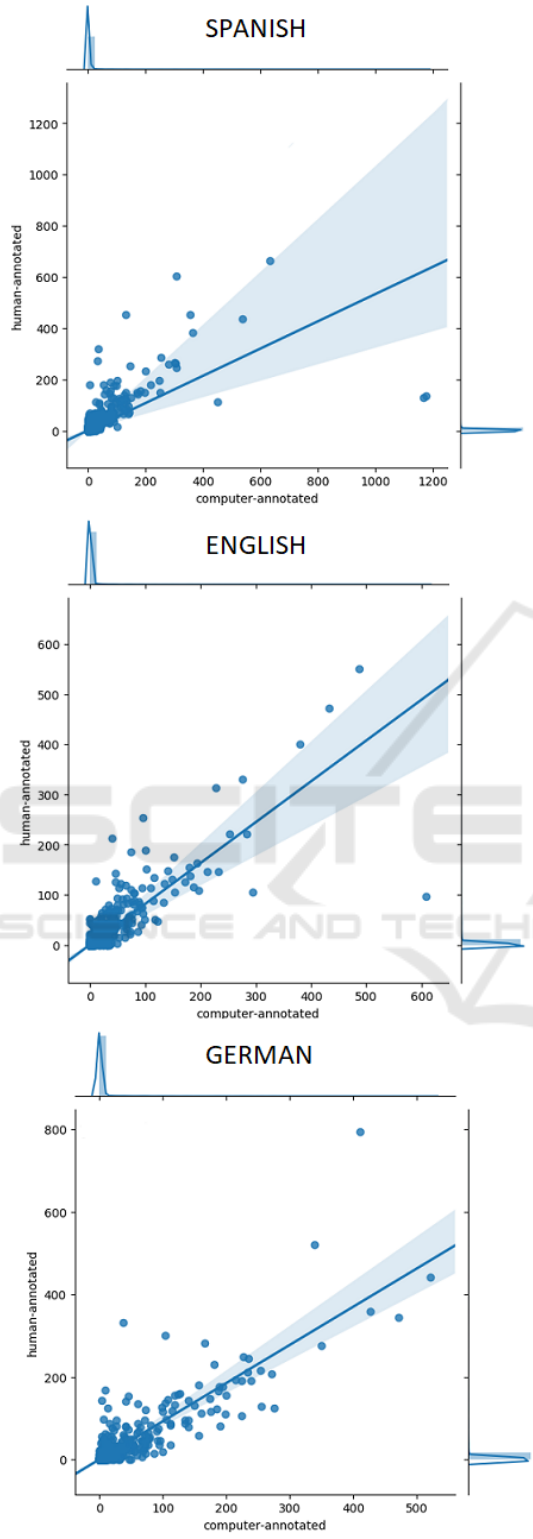

Figure 2: Comparison of code frequencies of 57 categories, trained only on the training part with the word embeddings for the out-of-vocabulary words, in six electoral programs by human and semi-automatic coding for Spanish, English, German texts respectively.

Table 5: Overview of the results for the algorithms that use FastText nearest neighbours instead of the words out-of-vocabulary. Performance varies across the languages.

Language	Accuracy without O-o-V	Change with O-o-V repl.	# of O-o-V in test	Vocab. size
English	0.426	−0.9%	3 485	52 949
German	0.372	+1.1%	3 266	24 227
Spanish	0.449	+3.2%	8 018	49 969

neighbors that are included in the training dataset can partially address the problem. Moreover, the more out-of-vocabulary words there are in the test dataset, the better such replacement performs. Indeed, Table 5 shows that the accuracy significantly improves for Spanish that has twice as many words out-of-vocabulary in the test set. In contrast, for German and English, the performance varied within one percentage point (and is even weaker for English than for the model that omits O-o-V words altogether).

However, experiments clearly show that there is a need to analyze a more extended context for better label classification. With the current amount of monolingual data, there is little one can do to broaden the context used by the models. We believe that further accuracy improvements could be achieved with multilingual models with the attention that could leverage varying importance of the words within different topics.

4 DISCUSSION

The achieved results are promising, concerning the complexity of the category scheme. Indeed, human coders' agreement level is only about 50%, comparing to a master copy (Lacewell and Werner(2013)). However, this level of accuracy does not allow to automation the real-world task completely.

It is also important to note here that some semi-sentences may contain more than one category. For coarse-grained ones, it is not a common problem, because the labels already include a variety of topics, but for the small labels, it is a real challenge. In this case, the current labeling scheme is difficult to reproduce by machine learning algorithms.

One possible way to modify the annotation process is to assign more than one label to the sample if it is needed. It should decrease ambiguity in the human coding process and, therefore, increase the machine-classification quality. Another idea is to change the structure of labels themselves to decrease overlapping.

5 CONCLUSION

This paper implements a classification algorithm for electoral programs from the Manifesto Project Database. A new approach is proposed to overcome the problem of the words that are out of the training vocabulary. The algorithm demonstrates the accuracy comparable to the state-of-the-art benchmarks for multi-label classification. The algorithm is tested on different languages, showing its applicability, and on different sizes of the kernel (weighting scheme of (Merz et al. (2016))). The experiments show that longer textual context is useful for the classification accuracy.

ACKNOWLEDGEMENTS

Authors are grateful to Oleg Gluhih, Maxim Gnativ and Alexei Postnikov for their help and constructive discussions.

REFERENCES

Chen T. and Guestrin C. (2016) *Xgboost: A scalable tree boosting system.* Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, 785–794.

Vapnik V. (1998) *The support vector method of function estimation.* Nonlinear Modeling, title=The support vector method of function estimation, author=Vapnik, Vladimir, booktitle=Nonlinear Modeling, 55–85, Springer.

Lazer D., Pentland A., Adamic L., Aral S., Barabási A.L., Brewer D., Christakis N., Contractor N., Fowler J., Gutmann M., Jebara T., King G., Macy M., Roy D., Alstyne M. (2009) *Computational Social Science.* Science, vol. 323, issue 5915, 721-723

Koehn P. (2005) *Europarl: A Parallel Corpus for Statistical Machine Translation.* MT Summit. http://www.statmt.org/europarl/

Gentzkow M., Shapiro J.M., Taddy M. (2018) *Parsed Speeches and Phrase Counts. Congressional Record for the 43rd-114th Congresses* Palo Alto, CA: Stanford Libraries. https://data.stanford.edu/congress_text

Lehmann P., Lewandowski J., Matthieß T., Merz N., Regel S., Werner A. (2018) *Manifesto Corpus. Version: 2018-1.* Berlin: WZB Berlin Social Science Center.

Volkens A., Krause W., Lehmann P., Matthieß T., Merz N., Regel S., Weßels B. (2019) *The Manifesto Data Collection. Manifesto Project (MRG/CMP/MARPOR). Version 2019a.* WZB. https://doi.org/10.25522/manifesto.mpds.2019a

Mikhaylov S., Laver M., Benoit K.R. (2012) *Coder Reliability and misclassification in the human coding of party manifestos.* Political Analysis 20(1), 78–91.

Glavaš G., Nanni F., Ponzetto S.P. (2017) *Cross-lingual classification of topics in political texts*

Zirn C., Glavaš G., Nanni F., Eichorts J., Stuckenschmidt H. (2016) *Classifying topics and detecting topic shifts in political manifestos.* In PolText.

Subramanian S., Cohn T., Baldwin T. (2017) *Hierarchical Structured Model for Fine-to-coarse Manifesto Text Analysis* Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, vol. 1

Subramanian S., Cohn T., Baldwin T., Brooke J. (2018) *Joint Sentence–Document Model for Manifesto Text Analysis* Proceedings of Australasian Language Technology Association Workshop: 25-33.

Merz N., Regel S., Lewandowski J. (2016) *The Manifesto Corpus: A new resource for research on political parties and quantitative text analysis*, Research and Politics. DOI: 10.1177/2053168016643346

Lan M., Tan C.L., Su J. (2007) *Supervised and Traditional Term Weighting Methods for Automatic Text Categorization.* Journal of IEEE PAMI, vol. 10, No. 10

Lacewell O.P. and Werner A. (2013) *Coder training: Key to enhancing coding reliability and estimate validity.* In: Volkens A., Bara J., Budge I., et al. (eds) Mapping Policy Preferences from Texts. Statistical Solutions for Manifesto Analysts. Oxford: Oxford University Press.

Bojanovski P., Grave E., Joulin A., Mikolov T. (2016) *Enriching Word Vectors with Subword Information.*

Grave E, Bojanowski P., Gupta P., Joulin A., Mikolov T. (2018) *Learning Word Vectors for 157 Languages.* Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)

Verberne S., D'hondt E., van den Bosch A., Marx M. (2014). Automatic thematic classification of election manifestos. Information Processing & Management, 50(4), 554-567.