BENCHić-lang: A Benchmark for Discriminating between Bosnian, Croatian, Montenegrin and Serbian

Peter Rupnik and **Taja Kuzman**
Jožef Stefan Institute, Slovenia
taja.kuzman@ijs.si,
peter.rupnik@ijs.si

Nikola Ljubešić
Jožef Stefan Institute, Slovenia
Center za jezikovne vire in tehnologije
Univerze v Ljubljani, Slovenia
nikola.ljubesic@ijs.si

Abstract

Automatic discrimination between Bosnian, Croatian, Montenegrin and Serbian is a hard task due to the mutual intelligibility of these South-Slavic languages. In this paper, we introduce the BENCHić-lang benchmark for discriminating between these four languages. The benchmark consists of two datasets from different domains – a Twitter and a news dataset – selected with the aim of fostering cross-dataset evaluation of different modelling approaches. We experiment with the baseline SVM models, based on character n-grams, which perform nicely in-dataset, but do not generalize well in cross-dataset experiments. Thus, we introduce another approach, exploiting only web-crawled data and the weak supervision signal coming from the respective country/language top-level domains. The resulting simple Naive Bayes model, based on less than a thousand word features extracted from web data, outperforms the baseline models in the cross-dataset scenario and achieves good levels of generalization across datasets.

1 Introduction

The status of "separate language" for Bosnian, Croatian, Montenegrin and Serbian is frequently discussed and is in academic circles mostly understood as related to the construction of identity (Alexander, 2013) and diverging and converging tendencies throughout history (Ljubešić et al., 2018). While each is an official language in the respective country, with a separate top-level Internet domain (Ljubešić and Klubička, 2014), their mutual intelligibility cannot be disputed. Regardless of the mutual intelligibility, differences do exist (Ljubešić et al., 2018). In this paper, we introduce a discrimination benchmark based on two datasets: a newspaper-based one, covering three out of four languages, and a Twitter-based one, covering all four languages. The publication of this benchmark coincides with the 10th anniversary of the VarDial workshop, in which this language group has been involved from the beginning.

The main contributions of this paper are the following. We introduce two datasets, based on previously collected data, that we now encode with maximal structure and publish in an academic data repository following the FAIR principles (Jacobsen et al., 2020). We introduce a benchmark based on the two datasets, and present baselines for the benchmark. Given the low performance of these competitive baselines on the benchmark, we introduce a new web-dataset-based method that shows to carry specificities of each language across the two datasets much better than any model directly trained on one of the two datasets. We hope that the availability of this benchmark, as well as the introduced strong competitors, will motivate further research in discriminating between similar languages.

2 Benchmark Datasets

The benchmark consists of two rather different datasets, whose selection was made with the aim of fostering cross-dataset evaluation of different modelling approaches. The first dataset is the parallel newspaper dataset from the "South-Eastern Times" (SETimes) website covering news in languages of South-Eastern Europe, including Bosnian, Croatian and Serbian. The dataset has been part of the VarDial shared task since 2014 (Zampieri et al., 2014) as part of the DSLCC collection (Tan et al., 2014), and was present in the following iteration of the shared task as well (Zampieri et al., 2015). Within VarDial, it was available in the form of 22 000 instances per language, each no longer than 100 tokens. We have now published all available content from the SETimes website in the form of 9 258 whole documents (Ljubešić and Rupnik, 2022a).[1] The documents are separated into a train, devel-

[1] http://hdl.handle.net/11356/1461

Tenth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2023), pages 113–120
May 5, 2023 ©2023 Association for Computational Linguistics

opment and test subset in an 8:1:1 ratio. While dividing the documents, we made sure that, given that the dataset consists of the same content in the three languages, there is no leakage of parallel data across these three subsets, especially given the mutual intelligibility of the languages covered. We assume that, given the parallel nature of this dataset, it could be very useful in learning the specifics in which the three close languages differ. The median length of instances (documents) is 627 words, while the arithmetic mean length is 849 words.

The second dataset is based on tweets, harvested with the TweetCat (Ljubešić et al., 2014) tool. This dataset was used as the out-of-domain testing data in the third iteration of the VarDial shared task (Malmasi et al., 2016), but in significantly smaller volume than what we included in this benchmark. We share tweets of 614 users, 394 of which are labeled as tweeting in Serbian, 89 in Croatian, 75 in Bosnian, and 56 in Montenegrin. Each user is represented with at least 200 tweets, merged in our experiments into one single text per each user. Single tweets were not filtered by the language they are written in, which allows for other languages besides the four languages of interest to occur in the dataset, such as infrequent tweets in English. With this decision we wanted to keep the dataset as natural and realistic as possible. The users were split into the train, development and test subset in a 3:1:1 ratio, so that the development and test splits would be large enough. The median length of instances (all tweets of one user) is 5,438 words, while the arithmetic mean length is 7,257 words. The dataset is published as a JSON file, each primary entry representing one user, the label denoting which language the user is tweeting in, and a list of the users' tweets (Ljubešić and Rupnik, 2022b).[2]

The benchmark allows for training on any of the two training datasets, as well as using external data, provided that it does not overlap with texts in the test split. Hyperparameters or model decisions can be chosen with the help of development data. The two official metrics of the benchmark are micro F1 and macro F1, both considered equally important.

The researchers are welcome to add to this benchmark the results achieved on any combination of training and testing datasets (in-domain or out-domain). However, the primary goal of this benchmark is to present results obtained in the cross-dataset scenario, that is, testing the model on test data from a dataset on which the model was not trained on, to prove the general applicability of the resulting model on the task of discriminating between the languages in question. The results of various models can be submitted via the GitHub repository[3] through a pull request.

3 Experiments

We experiment with two approaches: the baseline approach – a linear SVM model with character n-gram representation, described in Section 3.1, and our new approach, presented in Section 3.2: a Naive Bayes model using a text representation based on feature extraction from national web corpora. The classifier selection in each of the approaches is based on best results on the development data, and each of the two classifiers were considered in each of the approaches.

3.1 Baseline: SVM Model with Character N-Gram Text Representation

For the initial baseline of this benchmark, we used a simple approach that has been shown to be very competitive with even much more complex solutions (Malmasi et al., 2016; Zampieri et al., 2017) – a linear SVM model, used with the character n-gram text representation. We implemented the baseline solution inside the sklearn package (Pedregosa et al., 2011), and the only hyperparameter we tuned was the maximum length of the character n-gram, given that the shortest character n-gram is 3.

During hyperparameter tuning on the development data, we first selected the appropriate classifier, comparing the SVM and the Naive Bayes classifier while using character 3-grams as features. The results showed, as expected, that SVMs work better with the significant number of features produced with the character 3-gram feature generator. We next compared character 3-gram and 3–5-gram representations on our development data. The experiments showed that the character 3–5-grams perform slightly better in the in-dataset setup, reaching 1 to 6 points higher micro and macro F1 scores, while in the cross-dataset setup the 3-gram text representation provides slightly better results, outperforming the 3–5-gram representation by 1 to 4 points. This result does not come as a surprise as the character 3-gram model has a higher generaliz-

[2] http://hdl.handle.net/11356/1482

[3] https://github.com/clarinsi/benchich/tree/main/lang

Test data	Train data	micro F1	macro F1
SETimes	SETimes	0.995	0.995
	Twitter (3 class)	0.839	0.672
Twitter (3 class)	SETimes	0.743	0.747
	Twitter (3 class)	0.929	0.875

Table 1: Results of the linear SVM baseline with a character 3-gram text representation, trained either on the SETimes or the Twitter 3-class dataset, and tested in the in-dataset and the cross-dataset setup.

ability, important in the cross-dataset setup, while the 3–5-gram model has more capacity to learn the specifics of a dataset, preferred in the in-dataset setup. In further experiments, we use the character 3-gram text representation, as we are interested in a model which is able to generalize well to be applicable to different downstream datasets.

The method is tested on in-dataset and cross-dataset experiments, using the benchmark datasets: the SETimes and Twitter datasets. The in-dataset experiments consist of training and testing the model on the train and test split from the same dataset, while in the cross-dataset experiments, the model is trained on the train split from one dataset and tested on the test split of the other dataset. The cross-dataset setup was shown to be especially relevant for the task of discrimination between similar languages (Malmasi et al., 2016; Zampieri et al., 2017; Gaman et al., 2020), as well as document classification in general, because it shows the ability of the model to generalize across the datasets, and with that, its usefulness for the real-world applications.

Given that the SETimes dataset covers only three out of the four languages, while the Twitter dataset covers all four languages of interest, we used only languages that occur in both datasets for the baseline experiments, that is, the Bosnian, Croatian and the Serbian language.

Table 1 shows the results of the two baseline models, that is, the SVM model, trained on SETimes, and the SVM model, trained on the Twitter dataset. The models were tested on test splits from both datasets, showing their in-dataset and cross-dataset performance. The results show that, as expected, the in-dataset results are much higher than the cross-dataset results on both datasets. The in-dataset results reached up to 0.995 micro and macro F1 scores in the case of the SETimes model and 0.929 micro F1 and 0.875 macro F1 in the case of the Twitter model. As expected, in the in-domain setup, the SETimes model achieves higher results than the Twitter model. Somewhat unexpected, in the cross-dataset setup both combinations of training and evaluation data result in a very similar micro F1, showing a similar level of per-instance cross-dataset portability. However, on the macro F1 metric, the SETimes dataset shows to be a simpler evaluation dataset than the Twitter dataset, which is quite probably due to the fact that the SETimes dataset is more balanced, while the Twitter dataset is more challenging with its intensive skewness towards the Serbian language.

In the cross-dataset setup, the models scored for 9 up to 25 less points in micro and macro F1 points than in the in-dataset setup. This shows that models trained on any of the two datasets show to be rather incapable of generating predictions in the cross-dataset scenario that would be useful in the downstream, as around 25% of predictions are incorrect.

3.2 Our Approach: Naive Bayes Model and Web Corpora Feature Extraction

Given the rather low results of the proposed baselines in the cross-dataset setup on both datasets, we decided to propose a more robust approach to discriminating between the languages included in this benchmark. Since each of the four languages/countries has a top-level Internet domain (`.hr` for Croatian, `.ba` for Bosnian, `.me` for Montenegrin and `.rs` for Serbian), and since there are crawls of all four top-level domains available (Ljubešić, 2021), we are proposing a weak-supervised approach exploiting the information about the top-level domain from which a text came as our signal of weak supervision. That is, we regard texts from a specific top-level domain as being of the language related to the domain, e.g., texts from `.hr` as texts in Croatian language. Based on this, we perform a feature selection that identifies a small subset of words that are most specific for each language, i.e., top-level domain.

For the experiments, we use web corpora for the

	Feature Extraction		Training	
	paragraph #	word #	paragraph #	word #
Bosnian	943 515	18 503 316	2 102 489	37 681 981
Croatian	959 600	17 536 075	1 970 022	32 639 016
Montenegrin	864 921	35 684 637	999 997	35 677 096
Serbian	952 964	17 954 495	2 868 638	49 577 451

Table 2: Size of the parts of the web corpora used for feature extraction and model training.

four languages, available as part of the BERTić-data (Ljubešić, 2021), a text collection used for training the BERTić transformer model (Ljubešić and Lauc, 2021). We use part of the data for feature extraction and part of the data for training the Naive Bayes model, using the obtained features. Similarly to the baselines presented in the previous step, we have considered both the linear SVM and the Naive Bayes model, but the latter proved to be better performing on this task. The amount of data used for the feature extraction and for the model training is shown in Table 2. We used between 17 and 35 million words for the feature extraction, while we trained the classifier on 100 000 documents from each of the four top-level domains, each consisting of between 33 and 50 million words.

The feature extraction is based on comparing pairs of web corpora: for each pair, we identify features (words) that are the most specific for one language given another language. The weighting function for each language pair is the odds ratio, i.e., how much more probable it is for a word to appear in one language (or web corpus) in comparison to another language. As possible features, we consider words of three or more characters, consisting only of letters.

One hyperparameter that has to be tuned in our approach is the number of features per ordered language pair to be included in the feature set. Our experiments on the development data of both the SETimes and the Twitter dataset showed that using around 100 most prominent features per ordered language pair gives the best results on both test datasets. Since we obtain from each ordered language pair a list of 100 features, we have to calculate a union of 12 lists of 100 features, resulting in 819 final features, due to expected feature repetition. When training a model, texts are represented as vectors based on the 819 features, created with the CountVectorizer tool, available inside the sklearn package (Pedregosa et al., 2011). Preliminary experiments on the development set showed that among various quantifications of feature occurrence (frequency, TF-IDF, binary), the binary values regularly provided the best results.

Our next step is to train our model on web texts, classified into languages based on the top-level domain they are published on. As already reported, preliminary results showed that the Naive Bayes classifier performs better than the linear SVM classifier. It was shown to be much more stable across datasets, which does not come as a surprise given the low number of selected features. This is exactly the opposite from our baseline method, relying on many character n-gram features, where the SVM method showed to perform better. Interestingly, for both classifiers, the optimal number of features per ordered language pair showed to be around 100 features.

SETimes test data		
model	micro F1	macro F1
NB Web	0.957	0.957
SVM SETimes	**0.995**	**0.995**
SVM Twitter	0.839	0.672

Twitter 3-class test data		
model	micro F1	macro F1
NB Web	**0.946**	**0.897**
SVM Twitter	0.929	0.875
SVM SETimes	0.743	0.747

Twitter 4-class test data		
model	micro F1	macro F1
NB Web	**0.870**	0.682
SVM Twitter	**0.870**	**0.732**

Table 3: Results of our Naive Bayes model with web feature-based text representation and trained on web corpora (NB Web), compared to the baseline models: SVM model, trained on SETimes (SVM SETimes), and SVM model, trained on Twitter (SVM Twitter), on test splits of various datasets. The best results are in bold.

We compare our method, hereinafter referred to as "NB Web", with the in-dataset and cross-dataset baseline results, described in the previous section, both on the SETimes and Twitter test splits. The results are shown in Table 3. When the models are applied to the SETimes test dataset, the baseline SVM model, trained on the SETimes dataset, does perform best, reaching almost perfect scores of 0.995 micro and macro F1. This does not come as a surprise given the narrowness of the SETimes dataset (single-source news dataset). However, the NB Web model performs also rather well, micro and macro F1 scores lagging behind only for 4 points. Most importantly, the NB Web model performs drastically better than the baseline SVM model which was trained on the Twitter 3-class training dataset and used here in a cross-dataset setup.

The second section of Table 3 reports on the results on the Twitter 3-class test set where we used only instances from the three classes that are available in the SETimes dataset, that is, instances of Bosnian, Croatian and Serbian. In this setup, our model slightly outperforms even the in-dataset baseline results, i.e., the SVM model trained on Twitter data. It also performs drastically better than the baseline SVM model trained on SETimes, with a difference of more than 20 points.

This model finally allows also for some comparison to the 4-class baseline experiments, results of which we did not show in the previous section, given that the SVM classifier, trained on the SETimes dataset, only contains three out of four classes. The results on the Twitter 4-class test dataset are presented in the final section of Table 3. In this scenario, the NB Web model did not significantly outperform the baseline SVM model, trained on the Twitter dataset, as was the case on the 3-class Twitter test set. On the micro F1 metric, we obtained an equally good result with both methods – micro F1 of 0.87, while on the macro F1 metric, the SVM model, applied in an in-dataset setup, performs better, reaching the score of 0.73, while the NB Web model obtained 0.68 macro F1. However, given the equal result on the micro F1 metric, we assume that the edge of the (in-dataset) SVM Twitter model here is just the knowledge of the class distribution in the test set, information to which the NB web model was not exposed. Given this result, we can even assume that, with a class distribution far from the Twitter 4-class dataset, the NB web model should result in a better per-category performance than the in-domain method, and comparably on the per-instance level.

3.2.1 Impact of Amount of Training Data

Given that we have used a significant amount of data for training the web model (100 000 documents per class), we perform an additional analysis of the dependence of the performance of the NB Web model to the amount of web training data. We investigate how the model performs on all three test datasets (SETimes, Twitter 3-class and Twitter 4-class) if we are to train it on 25%, 50%, or 100% of our training data. The results are presented in Figure 1. The experiments show that we obtain very similar results to the previously presented ones even if we perform parameter estimation on one fourth of the training data. The only argument for using as much data as we are is the stability of the results, especially in the case of the 4-class Twitter problem, while on the SETimes dataset the results on less training data do not vary much.

What we have not explored, and what we leave for future work, is the impact of the amount of data used for feature selection. Given that best results were obtained with only 100 features selected from millions of words of text, we have to assume that these 100 features could have been similarly well extracted on a portion of the text used in our case.

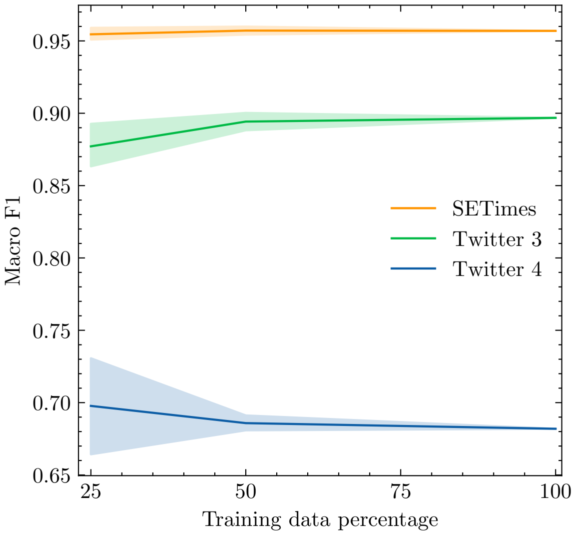

Figure 1: Impact of the size of the training dataset of the NB Web model on its performance on the SETimes and Twitter test datasets. Variation in the results is represented through the standard deviation.

3.2.2 Per-Category Performance

We conclude the results section with an analysis of the per-category performance of both models that are able to discriminate between all four languages, which are the baseline SVM Twitter in-

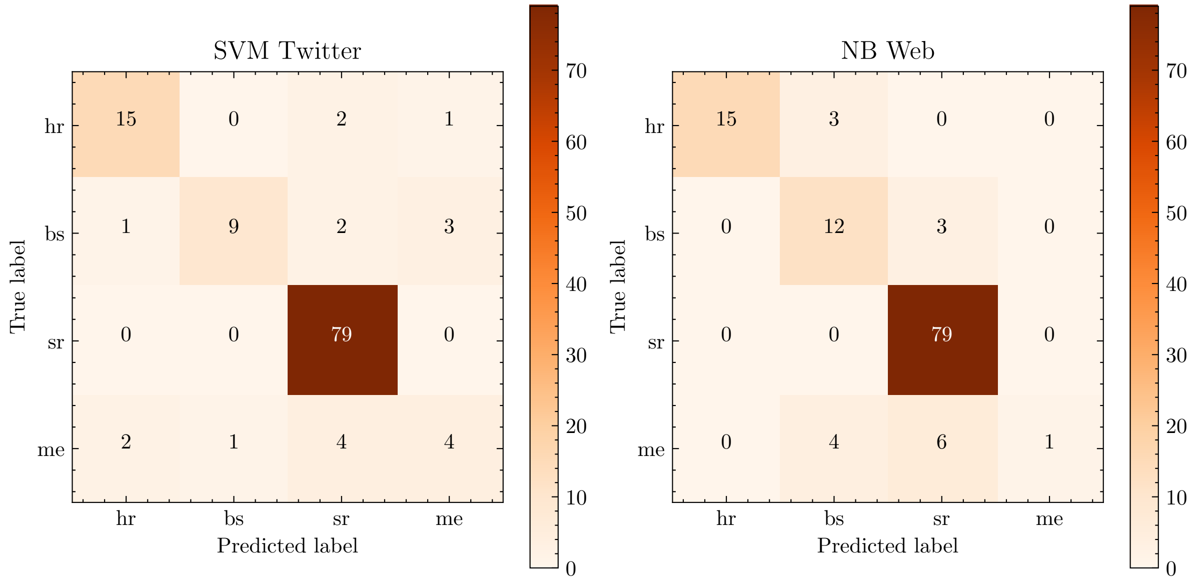

Figure 2: Confusion matrices for the baseline in-dataset SVM Twitter model and the NB Web model. The models are evaluated on the test split of the 4-class Twitter dataset.

dataset model, and the NB Web model. We present the performance via confusion matrices on the Twitter 4-class test split in Figure 2.

We can observe a good performance of both models on Croatian and Serbian, with a decent performance on Bosnian, especially with the NB Web model. However, the performance on Montenegrin is very unsatisfactory in case of both models. While the SVM Twitter in-dataset model correctly classifies only 4 out of 11 test instances in the Montenegrin category, the situation with the NB Web model is even worse. It classifies correctly only one out of 11 instances, others being taken primarily by Serbian and Bosnian. This analysis shows the limitation of our current results – while we do have a robust dataset-independent way of discriminating between Bosnian, Croatian and Serbian, the problem of identifying Montenegrin cannot be considered solved to a satisfactory level.

4 Conclusion

In this paper, we introduce the BENCHić-lang benchmark for discriminating between four very similar languages: Bosnian, Croatian, Montenegrin and Serbian. The benchmark consists of two rather different datasets, providing a good test bed for beyond-model generalizability.

We introduce two methods for discriminating between the languages. The first, a baseline, is a linear SVM model using character n-gram features, showing to perform well in-dataset, but not having generalization power to perform well in the cross-dataset setup. For that reason, we introduce another approach, exploiting only web-crawled data and the weak supervision signal coming from the country/language respective top-level domains. We perform heavy feature selection of less than 1000 word features on one subset of the web data, and train a Naive Bayes model on the remainder of the web data. We show that this model performs much better than the character n-gram models in the cross-dataset setting. What is more, it even outperforms the in-dataset results of the SVM model on one of the Twitter test sets. While we obtain very stable results on Bosnian, Croatian and Serbian, we must put forward that neither the in-dataset SVM Twitter, nor the NB Web model perform satisfactory on discriminating Montenegrin from the three other languages, which is a task to be tackled in future work.

Besides improving the identification of Montenegrin, there are many other directions we hope the community will investigate. One direction is exploiting linguistic features known to vary between the four languages (Ljubešić et al., 2018) and base the classification decision on these features. Another is to investigate transformer models, fine-tuning them either on the training data, or on the weak-supervision web data. We have performed an initial experiment on the latter, fine-tuning the BERTić model (Ljubešić and Lauc, 2021) for one epoch on the 400 000 web documents. During this first epoch we consistently obtained low results, with no tendency of improvement. Additional experimentation, potentially with lower learning rates or more complex loss functions, could be performed here. Finally, additional datasets should be added to the benchmark, especially such datasets that cover all of the four languages of interest.

Limitations

The two datasets included in the benchmark are by no means representative for the four languages we focus in this work. However, the datasets are different enough to serve as an initial test bed for robust discrimination between the four languages through a cross-dataset setup. Furthermore, the definition of these four languages is also rather problematic due to their similarity, and a potentially more viable option would be a linguistically-motivated multidimensional description of the variation among these languages, rather than aiming at the single-dimension 4-level description. The linguistically-motivated methods might be also more reliable, as they would be based on rules and lexicons, defined by linguists, rather than training corpora with unknown biases. We are aware that training the models with our method might introduce some bias to the results, because it is based on identifying words that are specific for each language by comparing the web corpora content. Consequently, some of the identified words might be more connected to topic differences between the corpora than variety differences. For instance, one of the words, specific for Croatian, is "kuna", a former Croatian currency, which is more of a culture-specific than a language-specific word. However, by extracting many features from very large numbers of documents, and then training the model on thousands of texts, we hope that such topic biases are minimized by the massive amounts of texts used.

Finally, using top-level domain information for assuming language labels is a weak-supervision method and is less reliable than manual annotation. With this approach, we presume that the majority of texts, published on the top-level web domain, are written by native speakers of the language that is associated with the respective country and its top-level domain. However, we are aware that it is possible that some texts are mislabeled and actually written in another language. We cannot be sure that the authors of these texts are native speakers, live in the respective country related to the national web domain, or that the text is not a republication from another source in another language, as was shown to be the case for the British-American English dataset in the Discriminating between Similar Languages (DSL) shared task 2014 (Zampieri et al., 2014).

Ethics Statement

We are aware that using web data is inevitably connected with questions of respecting the intellectual property and privacy rights of the original authors of the texts. In this paper, we used web corpora that have been collected by crawling the national top-level web domains. Only freely accessible texts were included in the corpora to avoid inclusion of sensitive data. Since the datasets were collected automatically and are too large to review manually, it is possible that the datasets include some texts whose authors do not consent to be included. However, in our paper, we only use the overall characteristics of the texts by extracting the most frequent language-specific words and do not examine the texts more closely or produce systems that could abuse personal information or intellectual property rights.

Secondly, as mentioned in Limitations, when training our NB Web model on web data, we presume that all texts from a specific national top-level domain are written in the main official language of the country to which the domain is connected. However, we are aware that there are national minorities of each of the analyzed languages that live across the borders of the country where the language is officially spoken, and that we can, for example, find a Serbian minority living in Bosnia and speaking Serbian on the Bosnian national web. By labeling all web texts from the Bosnian domain as Bosnian language, the resulting model could discriminate towards the minorities, equating their language with the language of the majority, publishing on the national domain. We are aware that our weak-supervision approach is a bit simplistic in regards to this issue, and while this is out of the scope of this paper, we plan to analyze this issue further in the future.

Acknowledgements

This work has received funding from the European Union's Connecting Europe Facility 2014-2020 - CEF Telecom, under Grant Agreement No. INEA/CEF/ICT/A2020/2278341. This communication reflects only the author's view. The Agency is not responsible for any use that may be made of the information it contains. This work was also funded by the Slovenian Research Agency within the Slovenian-Flemish bilateral basic research project "Linguistic landscape of hate speech on social media" (N06-0099 and FWO-G070619N, 2019–2023), the research project "Basic Research for the Development of Spoken Language Resources and Speech Technologies for the Slovenian Language" (J7-4642), and the research programme "Language resources and technologies for Slovene" (P6-0411).

References

Ronelle Alexander. 2013. Language and identity: The fate of Serbo-Croatian. In *Entangled Histories of the Balkans-Volume One*, pages 341–417. Brill.

Mihaela Gaman, Dirk Hovy, Radu Tudor Ionescu, Heidi Jauhiainen, Tommi Jauhiainen, Krister Lindén, Nikola Ljubešić, Niko Partanen, Christoph Purschke, Yves Scherrer, and Marcos Zampieri. 2020. A report on the VarDial evaluation campaign 2020. In *Proceedings of the 7th Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 1–14, Barcelona, Spain (Online). International Committee on Computational Linguistics (ICCL).

Annika Jacobsen, Ricardo de Miranda Azevedo, Nick Juty, Dominique Batista, Simon Coles, Ronald Cornet, Mélanie Courtot, Mercè Crosas, Michel Dumontier, Chris T Evelo, et al. 2020. FAIR principles: interpretations and implementation considerations.

Nikola Ljubešić. 2021. *Text collection for training the BERTić transformer model BERTić-data*. Slovenian language resource repository CLARIN.SI.

Nikola Ljubešić, Darja Fišer, and Tomaž Erjavec. 2014. TweetCaT: a tool for building Twitter corpora of smaller languages. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 2279–2283, Reykjavik, Iceland. European Language Resources Association (ELRA).

Nikola Ljubešić and Filip Klubička. 2014. {bs, hr, sr} wac-web corpora of Bosnian, Croatian and Serbian. In *Proceedings of the 9th web as corpus workshop (WaC-9)*, pages 29–35.

Nikola Ljubešić and Davor Lauc. 2021. BERTić - the transformer language model for Bosnian, Croatian, Montenegrin and Serbian. In *Proceedings of the 8th Workshop on Balto-Slavic Natural Language Processing*, pages 37–42, Kiyv, Ukraine. Association for Computational Linguistics.

Nikola Ljubešić, Maja Miličević Petrović, and Tanja Samardžić. 2018. Borders and boundaries in Bosnian, Croatian, Montenegrin and Serbian: Twitter data to the rescue. *Journal of Linguistic Geography*, 6(2):100–124.

Nikola Ljubešić and Peter Rupnik. 2022a. *The news dataset for discriminating between Bosnian, Croatian and Serbian SETimes.HBS 1.0*. Slovenian language resource repository CLARIN.SI.

Nikola Ljubešić and Peter Rupnik. 2022b. *The Twitter user dataset for discriminating between Bosnian, Croatian, Montenegrin and Serbian Twitter-HBS 1.0*. Slovenian language resource repository CLARIN.SI.

Shervin Malmasi, Marcos Zampieri, Nikola Ljubešić, Preslav Nakov, Ahmed Ali, and Jörg Tiedemann. 2016. Discriminating between similar languages and Arabic dialect identification: A report on the third DSL shared task. In *Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3)*, pages 1–14, Osaka, Japan. The COLING 2016 Organizing Committee.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in Python. *the Journal of machine Learning research*, 12:2825–2830.

Liling Tan, Marcos Zampieri, Nikola Ljubešic, and Jörg Tiedemann. 2014. Merging comparable data sources for the discrimination of similar languages: The DSL corpus collection. In *Proceedings of the 7th Workshop on Building and Using Comparable Corpora (BUCC)*, pages 11–15.

Marcos Zampieri, Shervin Malmasi, Nikola Ljubešić, Preslav Nakov, Ahmed Ali, Jörg Tiedemann, Yves Scherrer, and Noëmi Aepli. 2017. Findings of the VarDial evaluation campaign 2017. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, pages 1–15, Valencia, Spain. Association for Computational Linguistics.

Marcos Zampieri, Liling Tan, Nikola Ljubešić, and Jörg Tiedemann. 2014. A report on the DSL shared task 2014. In *Proceedings of the first workshop on applying NLP tools to similar languages, varieties and dialects*, pages 58–67.

Marcos Zampieri, Liling Tan, Nikola Ljubešić, Jörg Tiedemann, and Preslav Nakov. 2015. Overview of the DSL shared task 2015. In *Proceedings of the Joint Workshop on Language Technology for Closely Related Languages, Varieties and Dialects*, pages 1–9, Hissar, Bulgaria. Association for Computational Linguistics.