Brandeis at VarDial 2024 DSL-ML Shared Task: Multilingual Models, Simple Baselines and Data Augmentation

Jonne Sälevä and **Chester Palen-Michel**
Mitchom School of Computer Science
Brandeis University
{jonnesaleva, cpalenmichel}@brandeis.edu

Abstract

This paper describes the Brandeis University submission to VarDial 2024 DSL-ML Shared Task on multilabel classification for discriminating between similar languages. Our submission consists of three entries per language to the closed track, where no additional data was permitted. Our approach involves a set of simple non-neural baselines using logistic regression, random forests and support vector machines. We follow this by experimenting with finetuning multilingual BERT, either on a single language or all the languages concatenated together. In addition to benchmarking the model architectures against one another on the development set, we perform extensive hyperparameter tuning, which is afforded by the small size of the training data. Our experiments on the development set suggest that finetuned mBERT systems significantly benefit most languages compared to the baseline. However, on the test set, our results indicate that simple models based on scikit-learn can perform surprisingly well and even outperform pretrained language models, as we see with BCMS. Our submissions achieve the best performance on all languages as reported by the organizers. Except for Spanish and French, our non-neural baseline also ranks in the top 3 for all other languages.

1 Introduction

Language identification (LID) is the task of determining which language a piece of text is written in (Jauhiainen et al., 2019). While robust LID software already exists (e.g. Google's CLD3[1]), there are still several unsolved problems that plague current state-of-the-art LID models. One of the most pressing issues is lack of proper language coverage, which recent work has fortunately started to address as more data becomes available for more languages (e.g. Adebara et al., 2022; Burchell et al., 2023a; Kargaran et al., 2023).

Despite these promising developments, detection of lower-resourced languages, variants, and dialects still poses problems for modern NLP. The lack of resources also generally correlates with poor quality of the resources that are available which can lead to, for instance, datasets with unusually short sentences which may make the task difficult (Baldwin and Lui, 2010). To make matters worse, low-resource language variants tend to also be deceptively similar to other languages or dialects which makes differentiating between them accurately all the more challenging (Jauhiainen et al., 2019).

In the last ten years, the NLP for Similar Languages, Varieties, and Dialects workshop (VarDial) has emerged as the principal venue for discussion around these problems (e.g. Aepli et al., 2023, 2022; Chakravarthi et al., 2021). The workshop also features an annual shared task on discriminating between similar languages (DSL). The first VarDial DSL shared task DSL was organized with the purpose of better understanding the difficulties faced by state-of-the-art systems when differentiating between similar languages and varieties (Zampieri et al., 2014). Since then, multiple DSL shared tasks have been organized, leading to the development of a robust research community (Zampieri et al., 2014, 2015; Malmasi et al., 2016; Zampieri et al., 2017).

In the most recent VarDial DSL shared task, annotated datasets were added (Aepli et al., 2023). In the current iteration of the task, the labels were treated as a multi-label classification problem as proposed in Bernier-colborne et al. (2023).

In this paper, we describe our submission to the most recent VarDial shared task. For our submission, we experimented with simple non-neural baselines using scikit-learn, extensive hyperparameter tuning, data augmentation, and concatenating the

[1] https://github.com/google/cld3

Proceedings of the Eleventh Workshop on NLP for Similar Languages, Varieties, and Dialects (VarDial 2024), pages 241–251
June 20, 2024

Language	Split	Total Documents	Mean Sentences per doc	Mean Tokens per doc
EN	train	2,097	1.5	38.3
EN	dev	599	1.4	34.8
EN	test	300	1.4	35.3
BCMS	train	368	428.7	6,540.3
BCMS	dev	122	429.0	6,672.8
BCMS	test	123	465.1	6,999.1
FR	train	340,363	9.0	80.4
FR	dev	17,090	7.7	78.4
FR	test	12,000	12.0	96.7
PT	train	3,467	1.8	44.3
PT	dev	991	1.8	44.0
PT	test	495	1.8	43.7
ES	train	3,467	1.9	58.7
ES	dev	989	1.9	58.6
ES	test	495	1.9	60.2

Table 1: Counts of documents, average sentences per document, and tokens per document for each dataset.

datasets in an attempt at enhancing multilingual transfer. Ultimately, we found the best performing models for all languages tended to be fine-tuned mBERT variants (Devlin et al., 2018), except BCMS whose best performing model was a non-neural random forest model implemented in `scikit-learn` (Pedregosa et al., 2011).

2 Task Description

The shared task (Chifu et al., 2024) consisted of distinguishing between different varieties of a macro-language. There were 5 macro-language groups in the shared task. Some datasets differ notably in the size of a single classification instance, which we refer to as documents. In Table 1, the number of total documents for each of the splits is shown along with the mean sentences and tokens per document. The tokens and sentences are obtained by using the `spaCy` library and the `*_core_small` models for each language. For BCMS, we used the Croatian model, since it was the only language explicitly supported by `spaCy`. It can be seen that the French dataset is much larger than the others and that the BCMS dataset contains much longer documents in terms of sentences and tokens than any of the other datasets.

Data Sources The English, Spanish, and Portuguese data is from DSL-TL (Zampieri et al., 2024), which is manually annotated labels from the Discriminating Similar Languages Corpus Collection (DSLCC) (Tan et al., 2014). The French data partially comes from FreCDo (Găman et al., 2023) and DSLCC. French is also the only language whose dataset has named entities masked out. The Bosnian, Croatian, Montenegrin, and Serbian (BCMS) data comes from BENCHić-lang (Rupnik et al., 2023) and Twitter HBS 1.0 (Ljubešić and Rupnik, 2022) as well as Miletić and Miletić (2024). Given that much of the BCMS data is derived from Twitter, it is fairly different than the other datasets in terms of content. Details regarding the origins of the datasets and how they were annotated are summarized in Table 2.

3 System Descriptions

We made three submissions for the closed track. The three submissions consisted of our best performing models for scikit learn based classifiers, our best performing models using fine-tuning of mBERT, and a fine-tuned mBERT model using the concatenation of all datasets.

3.1 Run 1: `scikit-learn` Baselines

For Run 1, we submitted our best model from testing a series of scikit-learn classifiers: logistic regression models, linear-kernel SVMs and random forest models. For all models, we used bag-of-n-grams-style features where the n-grams were defined over (a) space-separated tokens (`analyzer=word`) or (b) characters (`analyzer=char`). In addition to integer counts (`CountVectorizer`), we also experimented with real-valued tf-idf weights (`TfidfVectorizer`) as an alternative representation. To prevent overfitting, we did not consider n-grams beyond $n = 2$. The full set of hyperparameters is shown in Table 3. The best performing configurations can be found in Table 4.

3.2 Run 2: Per-language mBERT Models

For our second run, we experimented with fine-tuning multilingual BERT (Devlin et al., 2018) independently on each language. We used `bert-base-multilingual-cased` for each submission[2]. The multilingual BERT model is pre-trained on masked language modeling and next sentence prediction. All macro-languages are included in mBERTs pre-training data. While the documentation of mBERT is less clear about variants of the macro-languages are included, for BCMS, individual languages are listed. All BCMS languages are

[2] `https://huggingface.co/google-bert/bert-base-multilingual-cased`

Lang.	Original data	Varieties	Train	Dev / Test	Annotation	Entities
English	DSL-TL	British English American English	Multi-label	Multi-label	Manually	Present
Spanish	DSL-TL	Castillian Spanish Argentinian	Multi-label	Multi-label	Manually	Present
Portuguese	DSL-TL	Brazilian, Portugal	Multi-label	Multi-label	Manually	Present
French	FreCDo, DSLCC	Canadian, Belgian Metropolitan French, Swiss	Multi-label	Multi-label	Automatically	Masked
BCMS	BENCHić-lang / Twitter HBS 1.0	Bosnian, Serbian, Montenegrin, Croatian	Single-label	Multi-label	Manually	Present

Table 2: Description of datasets included in the shared task.

Hyperparameter	Values
Architecture	Random forest, log. reg., SVM
Mode	multilabel, multiclass
Feature type	count, tf-idf
n-gram level	word, char
n-grams range	unigrams, bigrams, both
Solver	newton-cg, lbfgs, liblinear, sag, saga
Regularizer (C)	0.001, 0.01, 0.1, 1, 10, 100
Class weight	unadjusted, balanced
Max. iterations	off, 5000
Max. features	off, sqrt
No. of estimators	50, 100
Max. depth	30, 50

Table 3: Hyperparameter values used in non-neural `scikit-learn` experiments (Run 1).

represented in mBERTs pre-training data except for Montenegrin. We experimented with different hyperparameters for fine-tuning; the full set of values used can be seen in Table 5.

We adapt mBERT to multi-label classification by using a linear layer for classification, applying a sigmoid function to the logits and setting a threshold of 0.5 for the label to be included in the output. At inference time, if no output label meets the threshold, we relax the threshold to ensure each example is labeled first to .25, then .05. If after relaxing the threshold no label is assigned, we assign the most common label for the dataset.

Because the BCMS dataset had particularly longer documents with multiple sentences, we segmented each example first into sentences using spaCy (Honnibal et al., 2020). We then trained a model to predict on independent sentences. For inference we segment the documents first and classify each of their sentences. We then obtain final labels for the document by including labels that occur over a threshold of a proportion of the composite sentences. The threshold was set at 0.2 by adjusting to the development set.

All hyperparameters were tuned using an exhaustive grid search through all possible options. The hyperparameter configurations we experimented with for Run 2 can be found in Table 5.

3.3 Run 3: Finetuning All Languages at Once

For Run 3, we submitted mBERT fine-tuned on the concatenation of all the datasets. As we had already performed extensive hyperparameter tuning for Run 2, we opted to re-use well-performing hyperparameters from prior mBERT training runs for Run 2. Specifically, we used a learning rate of 2.0E-5, a batch size of 64, and 3 epochs to train the model with the concatenated dataset. We used a naive concatenation for this run and did not weight or sample the combined dataset in any special way. The motivation for this run is that it would provide a single model capable of distinguishing between similar languages for multiple macro-languages. As we discuss further in Section 5, this combined single model works decently well for most languages, but performs very poorly on the BCMS data.

4 Additional Experiments

In addition to the submitted systems, we conducted other experiments. These additional experiments included exploring data augmentation and segmentation of BCMS documents. Ultimately the BCMS segmentation was used for Run 2, but the data augmentation approaches did not appear to be useful enough to be included any of our submitted systems.

4.1 Segmenting BCMS

Noticing that performance was lower on BCMS and that the dataset had a much higher proportion of sentences per document compared with the datasets of other macro-languages, we compared

Language	BCMS	English	Spanish	Portuguese	French
Model	Random forest	Log. Reg. (OvR)	Random forest	SVM (OvR)	SVC (OvR)
Text features					
Count type	tf-idf	tf-idf	tf-idf	tf-idf	count
n-gram level	word	word	word	word	char
n-gram range	unigrams	unigrams	unigrams	both	bigrams
Common hyperparameters					
Solver	-	sag	-	-	-
Regularization (C)	-	10	-	10	100
Max iterations	-	100	-	5000	5000
Random forest params					
Bootstrap	False	-	False	-	-
Class weight	balanced	-	-	-	-
Max depth	50	-	50	-	-
Max features	-	-	sqrt	-	-
No. of estimators	50	-	100	-	-
F1 (macro)	71.33	79.75	82.99	72.01	55.00

Table 4: Best hyperparameters for scikit-learn models as computed on the development set.

Language	Batch Size	Learning Rate	Epochs
EN	16	2.0E-05	3
BCMS	16	2.0E-05	3
FR	16	2.0E-05	3
ES	64	3.0E-05	3
PT	16	2.0E-05	3

Table 5: Hyperparameters for individual mBERT models submission (Run 2).

	Orig. BCMS	Segmented BCMS
Macro F1	20.67	**72.2**
Weighted avg. F1	47.73	**79.8**

Table 6: Comparison of mBERT model on original BCMS dataset with segmented data.

performance from segmenting and not segmenting the data first. When segmenting the data into sentences, we used spaCy (Honnibal et al., 2020) with the Croatian model for all BCMS languages. In order to map back to the original examples, we label the example with any label that shows up in more than 20% of the composite sentences.

The results of this experiment are shown in Table 6. When applying segmentation and the strategy of classifying on each sentence individually, we saw a large gain of more than 50 points of macro F1 when segmenting first and then recombining.

4.2 Data Augmentation

Since some of the datasets had only a few thousand samples, we explored data augmentation as a way to obtain additional samples while still using only the datasets available for the closed track. Because the French and BCMS datasets contained hundreds of thousands of training sentences, we focused our data augmentation experiments on English, Spanish, and Portuguese. We attempted two simple data augmentation strategies.

First, since very simple word replacements have been shown to help model robustness (Wei and Zou, 2019; Kolomiyets et al., 2011) we tried naively splitting documents in half and recombined these half sentences with other half sentences of the same labels. The pieces from each sentence must have the same label. An example of this process is shown in Figure 1, where the label is EN-GB for all sentences in the example.

Second, similar to Zhang et al. (2022) or Andreas (2020), we attempted to replace segments based on spans from dependency trees with spans from other documents with the same labels. For the syntactic span augmentation, we use spaCy to get a dependency parse of each sentence. We then take a node and replace its children token span with another token span from a node of the same part of speech and parent dependency relation from a randomly sampled sentence with the same label. An example can be seen in Figure 2. In Figure 2, the label is EN-US for each sentence.

Unfortunately, neither of these approaches ended up providing a significant performance increase when evaluating on the development set.

We compare the naive augmentation, tree-based

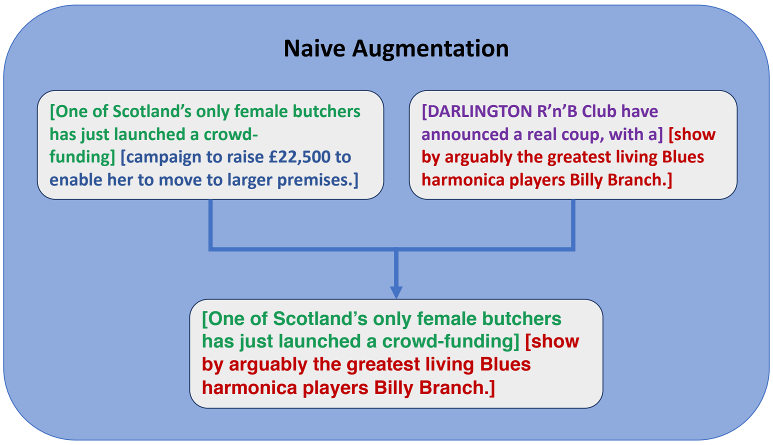

Figure 1: Naive augmentation approach.

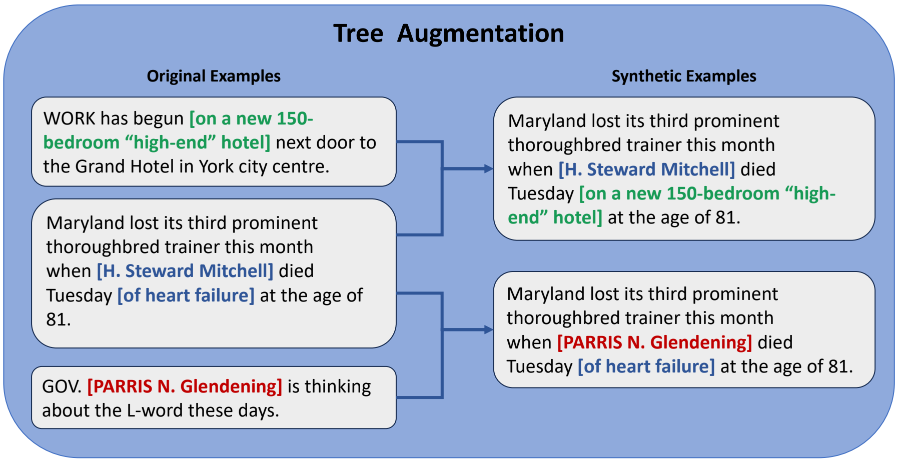

Figure 2: Tree augmentation approach.

Augmentation Strategy	EN	ES	PT
No Augmentation	**84.18**	**82.36**	74.45
Naive Aug.	82.47	82.09	**76.05**
Tree Aug.	81.8	81.19	73.69

Table 7: Results from data augmentation experiments. Scores are Macro-F1.

	EN	ES	FR	BCMS	PT
Run 1	79.75	74.49	54.26	69.32	72.01
Run 2	83.49	**83.50**	**96.58**	**72.20**	75.20
Run 3	**84.67**	82.75	68.40	20.67	**76.01**

Table 8: Macro F1 scores on the development set for each of our submissions on each language group.

augmentation, and no augmentation in Table 7 and find the macro-average F1 for each language is lower with the augmentations except for Portuguese. Since the Portuguese performance was only .04 higher than the concatenation model (run 3) and only seemed to benefit Portuguese, we decided not to submit any of the data augmentation approaches as part of our final submission.

5 Results

Based on performance on the development set as seen in Table 8, we expected Run 2 to perform best for Spanish, French, and BCMS and Run 3 to perform best for English and Portuguese.

Results from each submission are reported in Table 9. Run 3, the concatenated dataset with mBERT, does perform best for English and Por-

Language	Run	F1 (m.)	F1 (w.)
BCMS	Run 1: scikit-learn	**76.20**	**84.28**
BCMS	Run 2: mBERT	71.90	75.61
BCMS	Run 3: mBERT-all	19.85	45.30
EN	Run 1: scikit-learn	80.60	80.78
EN	Run 2: mBERT	85.27	85.56
EN	Run 3: mBERT-all	**85.48**	**85.62**
ES	Run 1: scikit-learn	74.59	75.31
ES	Run 2: mBERT	**82.27**	**82.68**
ES	Run 3: mBERT-all	82.09	82.31
PT	Run 1: scikit-learn	72.36	75.49
PT	Run 2: mBERT	71.40	74.10
PT	Run 3: mBERT-all	**75.21**	**77.71**
FR	Run 1: scikit-learn	27.03	27.03
FR	Run 2: mBERT	26.53	26.53
FR	Run 3: mBERT-all	**38.51**	**38.51**

Table 9: Test set results for all submitted runs. F1 (m.) and F1 (w.) refer to macro-F1 and weighted F1.

tuguese. However, for Run 1, Random Forest performed better on the test set for BCMS than mBERT-based models. Additionally, for Run 3, the concatenated dataset with mBERT, outperformed for French instead of Run 2 as seen on the development dataset.

To better understand the results, we created confusion matrices for our submitted runs for each dataset. Figure 3 shows the confusion matrix for Run 1 and 4 for Run 2. A confusion matrix for Run 3 is included in Appendix A.

Class imbalance appears to be a challenge, especially for BCMS and French. For Run 3, all predictions were for Serbian. Run 2 appears most capable for BCMS of making predictions that are ambiguous but still at least partially correct. Run 1 clearly performs well on BCMS, but seems to struggle with French class imbalance. For French, class imbalance seems to affect Run 1 the most with all varieties being mistaken for Metropolitan French at a higher rate than other runs. Run 3 appears to do better at correctly classifying Belgian and Swiss French.

For English, Run 2 predicts British English more often. All runs appear to struggle with ambiguous examples in English and Portuguese. It appears models are better able to correctly predict ambiguous examples in Spanish than in other macrolanguages.

6 Discussion and Conclusion

In this paper, we presented the Brandeis submissions to the VarDial 2024 DSL-ML Shared Task. We conclude by discussing some relevant aspects of our findings.

Baselines Perform Remarkably Well Somewhat contrary to our initial expectations, `scikit-learn`-based models seemed to perform well on both the development and test sets for many languages. On English, Portuguese and BCMS, the non-neural baselines underperformed mBERT by less than 4 macro-F1 points which is remarkable given the drastically smaller size of the baselines. This suggests that simple baselines may carry more utility than initially anticipated.

Further, the baseline performance on the test set shows stronger evidence of their utility. On French, Portuguese and BCMS, the baselines even outperform mBERT. While the differences in test set macro-F1 are less than 1 point in for both Portuguese and French, on BCMS the best baseline outperforms mBERT by more than 4.3 F1 points.

While this is a positive sign, we find the trend reversal somewhat perplexing. Since other trends, such as the universally low performance of Run 3 on BCMS, are replicated on both the test and development set, it stands to reason that this may not entirely be an issue of domain mismatch. Instead, we hypothesize that this may have to do with inherent noisiness in the kinds of low-resource data the shared task deals with.

Concatenation of Fine-Tuning Languages Contrary to the findings of Baldwin and Lui (2010), who showed that language identification becomes more difficult as the number of languages increases, we find that performance does not degrade significantly even after we increase the number of output labels from 2-4 per macrolanguage (independent mBERT models) to 14 (mBERT finetuned on all languages). One exception to this is BCMS, where mBERT-all underperforms even the official baseline. We hypothesize that with such a comparatively small number of languages (with other models like Burchell et al. (2023b) handling more than 200), increasing the number of languages to be classified does not degrade performance when the number of samples is comparable between languages. We speculate that BCMS languages may have underperformed with the concatenated model because there were drastically fewer examples. The majority class for BCMS is Serbian, and the minority classes are especially under-represented.

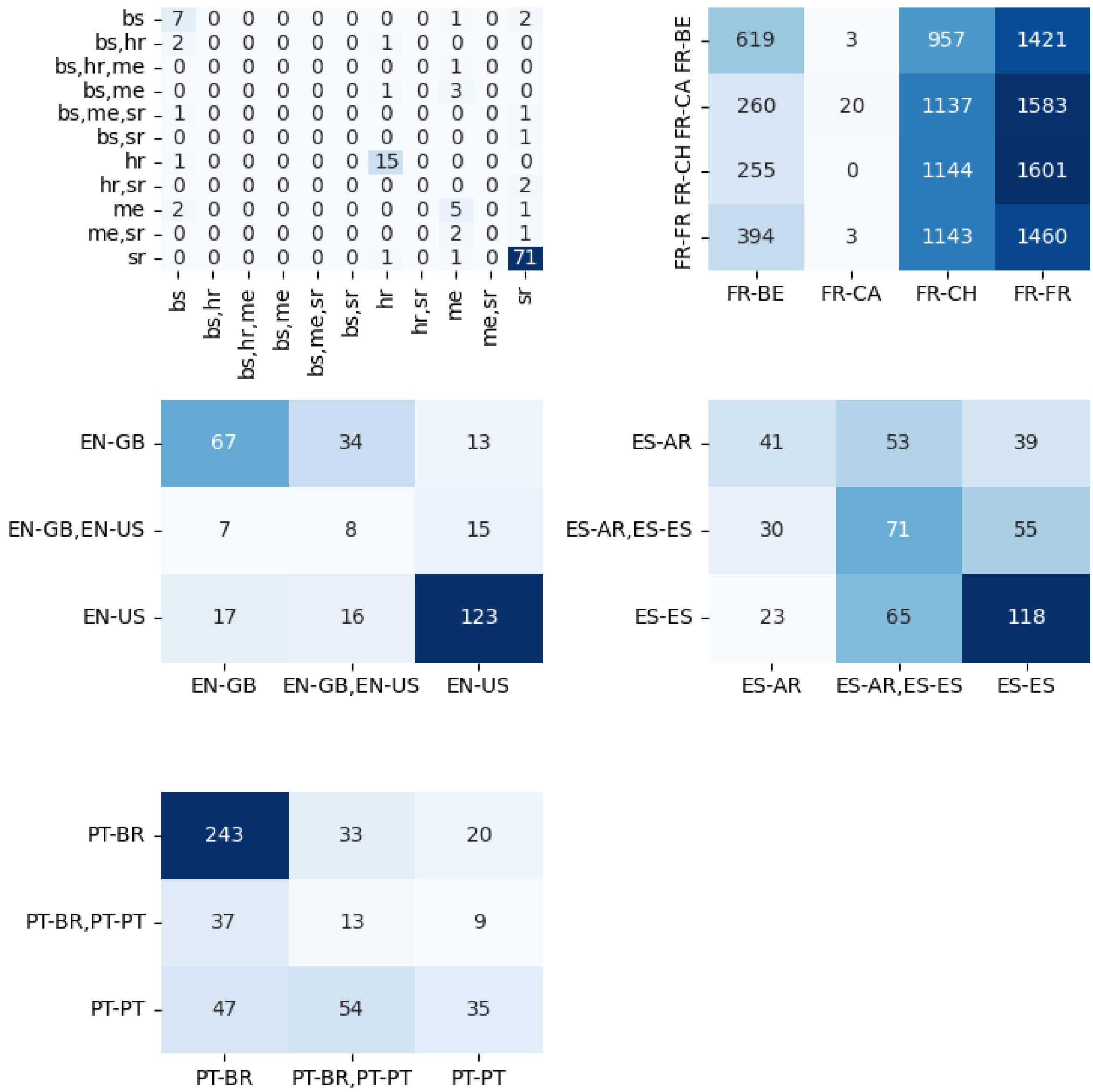

Figure 3: Confusion matrices for Run 1 on the test set. Correct labels are the x-axis and predicted are on the y-axis.

Simple Data Augmentation Does Not Help Much. We did not see improvement from fairly simple data augmentation approaches. It is possible that the models for discriminating similar models mostly rely on small spans of tokens that are already well represented in the original data. It is plausible that changing mixing spans of tokens into different contexts does not make much of a difference if those spans are already well weighted features and do not highly depend on what context they occur in. In future work, it may be worth attempting to better identify which spans are more informative features and experiment with data augmentation approaches that focus on the portion of the text that is most helpful in distinguishing the language variety.

Acknowledgements

The authors wish to thank the organizers of the VarDial 2024 DSL-ML Shared Task and the workshop in general. The authors also wish to thank Christopher Allison for providing technical support with the Brandeis Student Compute Cluster.

References

Ife Adebara, AbdelRahim Elmadany, Muhammad Abdul-Mageed, and Alcides Alcoba Inciarte. 2022. Afrolid: A neural language identification tool for african languages.

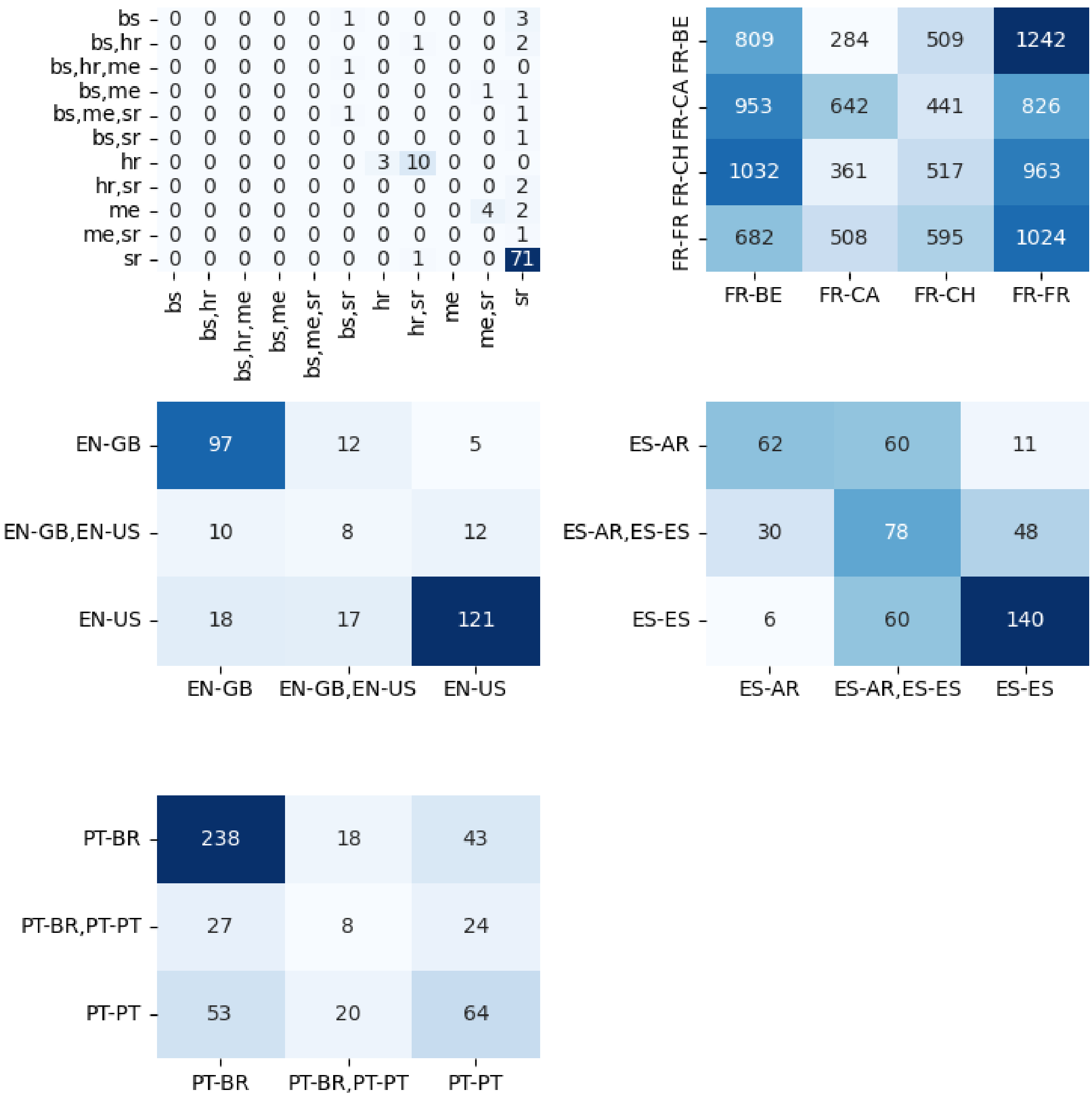

Figure 4: Confusion matrices for Run 2 on the test set. Correct labels are the x-axis and predicted are on the y-axis.

Noëmi Aepli, Antonios Anastasopoulos, Adrian-Gabriel Chifu, William Domingues, Fahim Faisal, Mihaela Gaman, Radu Tudor Ionescu, and Yves Scherrer. 2022. Findings of the VarDial evaluation campaign 2022. In *Proceedings of the Ninth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 1–13, Gyeongju, Republic of Korea. Association for Computational Linguistics.

Noëmi Aepli, Çağrı Çöltekin, Rob Van Der Goot, Tommi Jauhiainen, Mourhaf Kazzaz, Nikola Ljubešić, Kai North, Barbara Plank, Yves Scherrer, and Marcos Zampieri. 2023. Findings of the VarDial evaluation campaign 2023. In *Tenth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2023)*, pages 251–261, Dubrovnik, Croatia. Association for Computational Linguistics.

Jacob Andreas. 2020. Good-enough compositional data augmentation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7556–7566, Online. Association for Computational Linguistics.

Timothy Baldwin and Marco Lui. 2010. Language identification: The long and the short of the matter. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 229–237, Los Angeles, California. Association for Computational Linguistics.

Gabriel Bernier-colborne, Cyril Goutte, and Serge Leger. 2023. Dialect and variant identification as a multi-label classification task: A proposal based on near-duplicate analysis. In *Tenth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial*

2023), pages 142–151, Dubrovnik, Croatia. Association for Computational Linguistics.

Laurie Burchell, Alexandra Birch, Nikolay Bogoychev, and Kenneth Heafield. 2023a. An open dataset and model for language identification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 865–879, Toronto, Canada. Association for Computational Linguistics.

Laurie Burchell, Alexandra Birch, Nikolay Bogoychev, and Kenneth Heafield. 2023b. An open dataset and model for language identification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 865–879, Toronto, Canada. Association for Computational Linguistics.

Bharathi Raja Chakravarthi, Gaman Mihaela, Radu Tudor Ionescu, Heidi Jauhiainen, Tommi Jauhiainen, Krister Lindén, Nikola Ljubešić, Niko Partanen, Ruba Priyadharshini, Christoph Purschke, Eswari Rajagopal, Yves Scherrer, and Marcos Zampieri. 2021. Findings of the VarDial evaluation campaign 2021. In *Proceedings of the Eighth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 1–11, Kiyv, Ukraine. Association for Computational Linguistics.

Adrian Chifu, Goran Glavaš, Radu Ionescu, Nikola Ljubešić, Aleksandra Miletić, Filip Miletić, Yves Scherrer, and Ivan Vulić. 2024. VarDial evaluation campaign 2024: Commonsense reasoning in dialects and multi-label similar language identification. In *Eleventh Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2024)*, Mexico City, Mexico. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Mihaela Găman, Adrian-Gabriel Chifu, William Domingues, and Radu Tudor Ionescu. 2023. FreCDo: A large corpus for French cross-domain dialect identification. *Procedia Computer Science*, 225:366–373.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength natural language processing in python.

Tommi Jauhiainen, Marco Lui, Marcos Zampieri, Timothy Baldwin, and Krister Lindén. 2019. Automatic language identification in texts: A survey. *Journal of Artificial Intelligence Research*, 65:675–782.

Amir Hossein Kargaran, Ayyoob Imani, François Yvon, and Hinrich Schütze. 2023. Glotlid: Language identification for low-resource languages.

Oleksandr Kolomiyets, Steven Bethard, and Marie-Francine Moens. 2011. Model-portability experiments for textual temporal analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 271–276, Portland, Oregon, USA. Association for Computational Linguistics.

Nikola Ljubešić and Peter Rupnik. 2022. The news dataset for discriminating between bosnian, croatian and serbian SETimes.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.

Aleksandra Miletić and Filip Miletić. 2024. A gold standard with silver linings: Scaling up annotation for distinguishing Bosnian, Croatian, Montenegrin and Serbian. In *Proceedings of the 4th Workshop on Human Evaluation of NLP Systems*, Turin, Italy. European Language Resources Association.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Peter Rupnik, Taja Kuzman, and Nikola Ljubešić. 2023. BENCHić-lang: A benchmark for discriminating between Bosnian, Croatian, Montenegrin and Serbian. In *Tenth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2023)*, pages 113–120, Dubrovnik, Croatia. Association for Computational Linguistics.

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, Reykjavik, Iceland.

Jason Wei and Kai Zou. 2019. EDA: Easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6382–6388, Hong Kong, China. Association for Computational Linguistics.

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, Kai North, Tommi Jauhiainen, Mariano Felice, Neha Kumari, Nishant Nair, and Yash Bangera. 2024. Language variety identification with true labels. In *Proceedings of LREC-COLING*.

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, Dublin, Ireland. Association for Computational Linguistics and Dublin City University.

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.

Le Zhang, Zichao Yang, and Diyi Yang. 2022. TreeMix: Compositional constituency-based data augmentation for natural language understanding. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5243–5258, Seattle, United States. Association for Computational Linguistics.

A Run 3 Confusion Matrix

Figure 5 shows the confusion matrix for Run 3. Run 3 performs poorly on the BCMS dataset and only predicts Serbian for all examples. For French, Run 3 appears to do worse at predicting Metropolitan French, but better at Swiss and Belgian than Run 2.

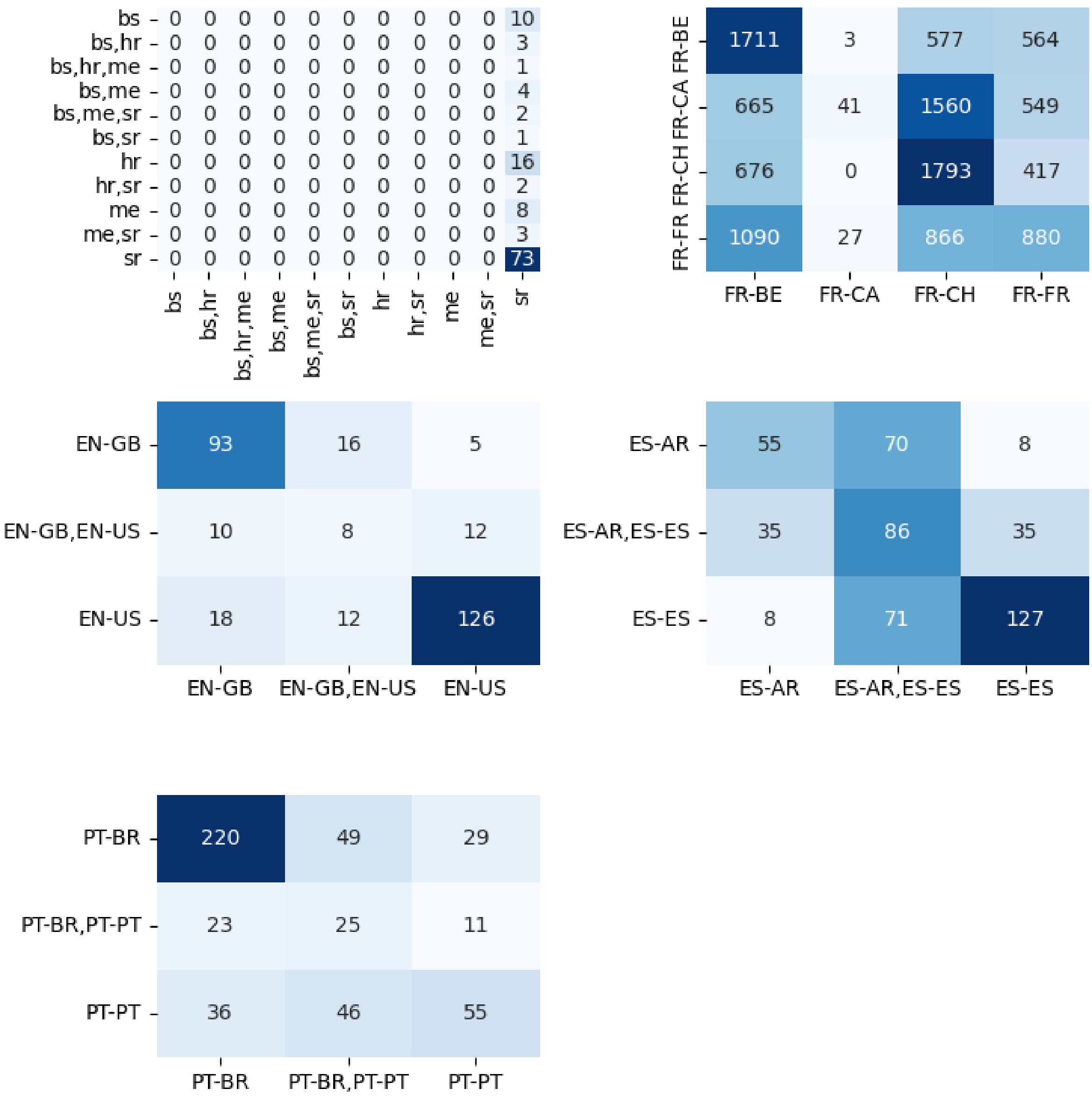

Figure 5: Confusion matrices for Run 3 on the test set. Correct labels are the x-axis and predicted are on the y-axis.