Dialect Identification under Domain Shift: Experiments with Discriminating Romanian and Moldavian

Çağrı Çöltekin
University of Tübingen
Department of Linguistics
ccoltekin@sfs.uni-tuebingen.de

Abstract

This paper describes a set of experiments for discriminating between two closely related language varieties, Moldavian and Romanian, under a substantial domain shift. The experiments were conducted as part of the Romanian dialect identification task in the VarDial 2020 evaluation campaign. Our best system based on linear SVM classifier obtained the first position in the shared task with an F1 score of 0.79, supporting the earlier results showing (unexpected) success of machine learning systems in this task. The additional experiments reported in this paper also show that adapting to the test set is useful when the training data comes from another domain. However, the benefit of adaptation becomes doubtful even when a small amount of data from the target domain is available.

1 Introduction

Language identification can be performed with near-perfect accuracy from a short text in many settings (Jauhiainen et al., 2019, for a recent survey of the solutions). However, automatic discrimination of texts from closely related languages or dialects remains to be a challenging task. The successful discrimination of texts between closely related language varieties may improve language identification for practical applications, as well as providing further insights into the differences between these linguistic varieties. Recent VarDial evaluation campaigns have featured discrimination challenges between related languages (Zampieri et al., 2017; Zampieri et al., 2018; Zampieri et al., 2019). The present study is conducted within the scope of the VarDial 2020 (Găman et al., 2020) Romanian Dialect Identification (RDI) task.

Romanian and Moldavian are two closely related language varieties spoken in Romania and the Republic of Moldavia respectively. The languages, particularly in written form, are very similar – to the extent that discrimination by human annotators is barely above chance levels (Găman and Ionescu, 2020). However, as evidenced by the last year's evaluation campaign (Zampieri et al., 2019), the machine learning methods applied to the task seem to be more successful (Chifu, 2019; Onose et al., 2019; Tudoreanu, 2019; Wu et al., 2019). The MOROCO corpus (Butnaru and Ionescu, 2019) used in last year's RDI shared task consists of texts from online news. Although the data is fairly balanced with respect to the topics, and some of the obvious non-linguistic cues (e.g., named entities) are removed from the data, the data may still contain some unintended non-dialectal cues (e.g., style differences between the newspapers in two countries). A natural question that arises is whether the machine learning methods tap into such non-obvious cues not relevant to linguistic differences, or the data contains a strong signal for identifying the linguistic variation. To this end, the present shared task includes data from two different domains (or genres). The source domain is the MOROCO corpus of newspaper text, and the target domain texts consist of a newly collected data set gathered from Twitter.

The systems used in the current study are based on ensembles of linear SVM models with a simple adaptation mechanism that retrains the model(s) with the data augmented by the test instances that are classified by a base classifier with high confidence. Besides the adaptation to the test set at prediction time, we also experiment with a training set selection method based on reverse-prediction. This method is

based on training a classifier on a small target data (development set) and selecting the training instances which the classifier predicts with high confidence.

In the remainder of this paper, we describe the system, present the results, and provide a discussion with brief conclusions.

2 System Description

The main task at hand is predicting the language variety (Romanian or Moldavian) under domain shift. The participants were provided with a large annotated corpus from the source domain (newspaper text) and a small annotated development set from the target domain. The evaluation is based on the performance of the systems on the target domain.

All experiments reported in this study are performed using linear SVM classifiers with sparse character and word n-gram features. Overlapping character and word n-gram features (for all 'n' from 1 to a maximum value, determined during tuning) are combined into a single feature set, and weighted using BM25 (Robertson et al., 2009). The input is tokenized using a simple regular expression tokenizer that treats any contiguous alphabetic or non-space character sequence as a token. Except for (optional) case normalization (treated as a hyperparameter), and filtering based on low document frequency (also another hyperparameter), no preprocessing or filtering is performed. The same system has been used with minor differences for discriminating similar languages in the earlier VarDial evaluation campaigns (Çöltekin and Rama, 2016; Çöltekin and Rama, 2017; Çöltekin et al., 2018), and obtained top or near-top results. The detailed description of the approach can be found in these papers.

For some of the experiments, we use an ensemble of linear SVM classifiers described above trained on different, non-overlapping parts of the data. The predictions of the individual classifiers are combined using weighted majority voting where the distances from the decision boundary are used as weights.

Another interesting aspect of the system is an adaptation technique used during prediction. The adaptation method is similar to the adaptation method used in a few systems in earlier VarDial shared tasks (Jauhiainen et al., 2018a; Jauhiainen et al., 2018b; Wu et al., 2019). The method relies on a base classifier trained on the training data. During testing, the test instances for which the base classifier is confident in its decisions are added to the training set and the classifier is retrained with this augmented training set.

One of the differences between the source and the target domain is the average length of the documents. On average, the source domain newspaper texts are naturally longer than the tweets (target domain). Although the feature weighting method we use (BM25) counteracts the sensitivity to document size to some extent, we split the source domain documents to sentences. In all of the shared task submissions, the source domain documents were split before training the models.

Another aspect of some of our systems is filtering large source domain data based on what we call 'reverse-prediction'. With the assumption that the features relevant for the target domain can be captured well by a classifier trained on the small target domain development set, we first tune and train a classifier on the development set. We use this classifier to predict the labels of the large training set. We select the predictions with high confidence that match the gold-standard labels. In this paper, we consider the test instances with a distance of more that 1.0 from the decision boundary as confident predictions. The intuition is that, despite reduced data size, the selected training instances may include features better tuned to the target domain.

3 Experiments and Results

3.1 Data

The data for this task comes from two different sources. The first data set, the MOROCO (Butnaru and Ionescu, 2019) corpus, is used as the source domain (or genre) in this task. The target corpus is collected from Twitter by Găman and Ionescu (2020). The source corpus is provided with a training–development set division. In most of our experiments we combine training and development sets, and split the documents into sentences,[1] labeling each sentence with the label of the document. During the

[1]Version 1.4 of the Python `sentence-splitter` library was used with defaults for splitting the documents into sentences (`https://pypi.org/project/sentence-splitter/`).

Data set	instances	μ_{char}	σ_{char}	μ_{token}	σ_{token}	RO/MD
Source						
Development	5923	1718.55	1746.34	391.88	393.94	1.18
Training	33 564	1714.99	1847.48	390.11	410.18	1.18
Train+dev sentences	424 383	158.81	141.72	36.34	36.04	1.45
Selection (documents)	21 376	1853.36	2122.72	420.26	469.18	0.95
Selection (sentences)	243 904	161.40	144.75	36.83	36.19	1.30
Target						
Development	215	91.43	20.01	24.09	6.39	0.90
Test	5022	92.09	18.86	23.66	6.04	1.01

Table 1: Summary of the data. The columns μ_{char} and μ_{token} indicate average number of characters and tokens, σ_{char} and σ_{token} indicate standard deviations of respective measures. 'RO/MD' is the ratio of Romanian instances to Moldavian instances, provided as a measure of class imbalance.

competition, only a small development set from the target domain was released, and the test data with labels were provided after the competition. The statistics on the data sets are provided in Table 1.

The source domain contains a slight class imbalance. Also because of the fact that the Romanian documents are longer on average, sentence splitting amplifies this imbalance. The target domain is much more balanced, and contains shorter documents on average, even compared to sentence-split version. The document lengths of the target domain are less varied. The Twitter data set is also more balanced. We apply training instance selection with reverse-prediction to the whole documents, then use the sentence split version in the experiments reported below. The resulting data contains approximately half of the training and development instances. One notable aspect is that the resulting data sets have longer texts, likely because the classifier is more confident on longer texts. Furthermore, the selection process reverses the class imbalance on documents. However, since Romanian documents are longer on average, the balance is again in favor of Romanian in the sentence-split data.

3.2 Experimental setup

All classifiers we use were tuned on the respective data sets. We tune the following hyperparameters: SVM margin/regularization constant 'C' in range $[0.01, 4.0]$; maximum character n-gram order in range $[0, 7]$; maximum token n-gram order in range $[0, 4]$; document frequency cutoff in range $[1, 5]$; and whether to apply case normalization to tokens or not. The BM25 parameters were kept at their default values suggested by Robertson et al. (2009). For each classifier trained, we draw 1000 random hyperparameter combinations, train and test it using 10-fold cross validation and record the average macro-averaged F1 score over the cross validation folds.

For large data sets (of the source domain), during prediction, we combine the output of 20 classifiers trained on non-overlapping equal parts of the training set, and the use the majority vote weighted by the distance from the decision boundary as the final decision. For small data sets, we also employ a less-effective form of ensembling. We train 5 separate models on the same data set using the top-five hyperparameter settings, and combine their decisions the same way. The implementation is based on Python scikit-learn library (Pedregosa et al., 2011).

3.3 Shared Task Results

We submitted three runs to the competition. The first run used only the target development set as training data. We tune a model with random search over the hyperparameters listed above using 10-fold cross validation on the target development set, we re-train 5 classifiers with the best hyperparameter settings on the complete target development data, and use their combined (with weighted voting) decisions on the test set as final predictions.

For the second run, we used the complete source data (after sentence segmentation). We first tune the classifier on a random 1/20th sample of the whole data. Then, re-train 20 classifiers on the non-

Data set	Precision	Recall	F1 Score
Target dev	84.04 (8.50)	84.77 (8.28)	84.01 (8.52)
Target dev+test	89.29 (0.92)	89.31 (0.91)	89.29 (0.92)
Source sentences	86.36 (0.19)	85.73 (0.18)	86.01 (0.18)
Source documents	96.37 (0.27)	96.51 (0.26)	96.26 (0.28)

Table 2: In-domain results. All scores are macro-averaged, and presented as percentages. The values in the parentheses are the standard deviations of the scores over the folds of in 10-fold cross validation experiments.

overlapping parts of the complete source data set (development and training sets), and use the weighted vote as final predictions.

The third run is based on selection of the sentences from the source data which were predicted confidently by a classifier trained on the target development set. In particular, for run 3, we select the sentences from the source data whose distance to the decision boundary is 1.00 or larger. The tuning and prediction follows the same procedure as the second run.

All systems used the test set adaptation method, where all test instances with distances 0.50 or higher from the decision boundary of the base classifier were added to the training set, and the predictions are obtained from a classifier trained on this augmented data set.

Our first two runs obtained the first two ranks among 19 submissions in the competition with macro-averaged F1 scores of 0.788 and 0.784 respectively. The final run, with training data selection, obtained the fifth rank with an macro-averaged F1 score of 0.756.

The results indicate that, given a large test set to adapt to, even a small amount of target training data is effective. When shifting the domains, it seems crucial to have more data. Even a carefully selected subset of out-of-domain data leads to inferior performance in comparison to small in-domain data.

3.4 In-domain Experiments

Besides the experiments with the systems for the shared task participation, we present a set of in-domain experiments without adaptation. For the source domain, we present both results with and without sentence splitting. For the target domain, we present results obtained on the small development set (215 tweets), and combination of both development and test sets. All performance scores are average scores on 10-fold cross validation on the indicated data sets. For each setting, the SVM classifier was trained with 1000 random draws from the hyperparameter space indicated above, and the highest scores were reported.

Table 2 presents the results of the in-domain experiments. The in-domain performance of the system on the source data is in-line with the last year's competition. The scores on source documents are almost the same as the post-competition result reported by Wu et al. (2019), which was 6.70 percentage points higher than the official winner. Training and testing the system on source sentences causes a performance drop of approximately $10\,\%$. Presumably, the decrease of performance is due to increased ambiguity as a result of decreased length of the documents. In fact, tuning and training the classifier on sentences (of the official training set), and testing it on the documents (of the official development set) results in comparable scores (macro-averaged F1 score is 95.04) – despite the mismatch of text length between development splits and the test set.

The results on the target domain are also impressive. Using only 215 tweets in a cross-validation setup, the average F1 score over cross-validation folds is 84.01. And more data definitely helps, both for increasing the performance, and reducing the variance. Once we have about 5000 instances, the average F1-score on 10-fold cross validation is close to the scores obtained on the news domain. And, interestingly, despite the smaller data set and shorter texts (even in comparison to news sentences), the model is more successful on tweets than the news sentences. In fact, running the same experiments on the source domain with the 5000 instances reduces the F1 score of the classifier to 93.80 and 73.96 for source documents and sentences respectively.

	Precision		Recall		F1 Score	
Training set	-adapt	+adapt	-adapt	+adapt	-adapt	+adapt
Target dev	78.20	78.83	78.20	78.76	78.20	78.76
Source all	76.63	78.44	76.57	78.43	76.57	78.43
Source select	74.54	75.66	74.54	75.65	74.53	75.65

Table 3: Comparison of systems with (+adapt) and without (-adapt) adaptation to the test set. All scores are macro-averaged, and presented as percentages.

3.5 Adaptation to the Test Data

All of our official submissions included the test set adaptation method described in Section 2. To show the effects of the adaptation method, we present the scores both with and without adaptation in this section. Table 3 presents the scores on the test set with and without domain adaptation. The results with domain adaptation are the scores of the official submissions. The results without domain adaptation is calculated on the test set released by the organizers after the competition.

The adaptation has little effect when training data is in-domain. This is perhaps not surprising as training and test domains are identical. However, considering the small amount of training data (target dev set), one hopes to get additional benefits from increased training set because of domain adaptation. When training data comes from a different text type, domain adaptation seem to be more effective, increasing the scores close to two percentage points. The increase is less helpful for the training set selected through reverse-prediction. This may be because of the fact that the reverse-prediction already does the part of the job of the test set adaptation. Nevertheless, the most successful method is the one trained on small in-domain data with a small margin in comparison to full source data with adaptation. When trained with out-of-domain data, domain adaptation is clearly useful. Selecting training instances based on reverse-prediction does not seem to be helpful. This may be due to decrease in training data size, but the selection procedure may have also resulted in over-tuning to the development set. Indeed, the training documents selected with reverse-prediction has a 'RO/MD' ratio of 0.95, closer to the development set distribution (0.90) than the training set distribution (1.18).

4 General Discussion

This paper presented results from (ensembles of) linear SVM classifiers on the task of cross-domain discrimination of Moldavian and Romanian. Our systems obtained top positions on the official competition. The results indicate that linear SVMs are (still) one of the best solutions in certain settings. Furthermore, the cross-domain success of the system supports findings of Găman and Ionescu (2020), that better-than-human achievement of the machine learning models is based on dialect differences, and not due to some correlated hidden variable in the data set.

The in-domain experiments indicate that if the training sizes are similar, the discrimination is better on the Twitter data in comparison to the newspaper text. The in-domain experiment on Twitter corpus yield better discrimination than the in-domain experiment with with news sentences despite smaller training set size. This probably indicates that non-standard texts contain more cues to dialectal differences, since they do not necessarily follow common literary and stylistic traditions expected from more formal texts.

The adaptation method based on test set augmentation was found to be useful when training and test domains are different. The benefit of the test set adaptation method is not clear when the training and tests are from the same domain, even when the training set is very small.

The training data selection experiments resulted in worse results than expected. A potential reason for the failure is the fact that most training instances selected by the procedure contain the features that already occur in the target development set, hence, not providing additional information expected from a large data set. Although the training set selection method as used in this study seems to have failed, methods involving relevant, yet diverse instance may be helpful in adapting to a new domain.

References

Andrei Butnaru and Radu Tudor Ionescu. 2019. MOROCO: The Moldavian and Romanian dialectal corpus. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 688–698, Florence, Italy, July. Association for Computational Linguistics.

Adrian-Gabriel Chifu. 2019. The R2I_LIS team proposes majority vote for VarDial's MRC task. In *Proceedings of the Sixth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 138–143, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Çağrı Çöltekin and Taraka Rama. 2016. Discriminating similar languages with linear SVMs and neural networks. In *Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3)*, pages 15–24, Osaka, Japan.

Çağrı Çöltekin and Taraka Rama. 2017. Tübingen system in VarDial 2017 shared task: experiments with language identification and cross-lingual parsing. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, pages 146–155, Valencia, Spain.

Çağrı Çöltekin, Taraka Rama, and Verena Blaschke. 2018. Tübingen-Oslo team at the VarDial 2018 evaluation campaign: An analysis of n-gram features in language variety identification. In *Proceedings of the Fifth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2018)*, pages 55–65.

Mihaela Găman and Radu Tudor Ionescu. 2020. The unreasonable effectiveness of machine learning in Moldavian versus Romanian dialect identification. *arXiv preprint arXiv:2007.15700*.

Mihaela Găman, 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 Seventh Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*.

Tommi Jauhiainen, Heidi Jauhiainen, and Krister Lindén. 2018a. HeLI-based experiments in Swiss German dialect identification. In *Proceedings of the Fifth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2018)*, pages 254–262, Santa Fe, New Mexico, USA, August. Association for Computational Linguistics.

Tommi Jauhiainen, Heidi Jauhiainen, and Krister Lindén. 2018b. Iterative language model adaptation for Indo-Aryan language identification. In *Proceedings of the Fifth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2018)*, pages 66–75, Santa Fe, New Mexico, USA, August. Association for Computational Linguistics.

Tommi Sakari 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.

Cristian Onose, Dumitru-Clementin Cercel, and Stefan Trausan-Matu. 2019. SC-UPB at the VarDial 2019 evaluation campaign: Moldavian vs. Romanian cross-dialect topic identification. In *Proceedings of the Sixth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 172–177, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

Diana Tudoreanu. 2019. DTeam @ VarDial 2019: Ensemble based on skip-gram and triplet loss neural networks for Moldavian vs. Romanian cross-dialect topic identification. In *Proceedings of the Sixth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 202–208, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Nianheng Wu, Eric DeMattos, Kwok Him So, Pin-zhen Chen, and Çağrı Çöltekin. 2019. Language discrimination and transfer learning for similar languages: Experiments with feature combinations and adaptation. In *Proceedings of the Sixth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 54–63, Ann Arbor, Michigan, June. 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, April. Association for Computational Linguistics.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Ahmed Ali, Suwon Shon, James Glass, Yves Scherrer, Tanja Samardžić, Nikola Ljubešić, Jörg Tiedemann, Chris van der Lee, Stefan Grondelaers, Nelleke Oostdijk, Dirk Speelman, Antal van den Bosch, Ritesh Kumar, Bornini Lahiri, and Mayank Jain. 2018. Language identification and morphosyntactic tagging: The second VarDial evaluation campaign. In *Proceedings of the Fifth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2018)*, pages 1–17, Santa Fe, New Mexico, USA, August. Association for Computational Linguistics.

Marcos Zampieri, Shervin Malmasi, Yves Scherrer, Tanja Samardžić, Francis Tyers, Miikka Silfverberg, Natalia Klyueva, Tung-Le Pan, Chu-Ren Huang, Radu Tudor Ionescu, Andrei M. Butnaru, and Tommi Jauhiainen. 2019. A report on the third VarDial evaluation campaign. In *Proceedings of the Sixth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 1–16, Ann Arbor, Michigan, June. Association for Computational Linguistics.