Language Discrimination and Transfer Learning for Similar Languages: Experiments with Feature Combinations and Adaptation

Nianheng Wu Eric DeMattos Kwok Him So Pin-zhen Chen Çağrı Çöltekin
University of Tübingen
Department of Linguistics
{nianheng.wu|eric.demattos|kwok-him.so|pinzhen.chen}@student.uni-tuebingen.de
ccoltekin@sfs.uni-tuebingen.de

Abstract

This paper describes the work done by team tearsofjoy participating in the VarDial 2019 Evaluation Campaign. We developed two systems based on Support Vector Machines: SVM with a flat combination of features and SVM ensembles. We participated in all language/dialect identification tasks, as well as the Moldavian vs. Romanian cross-dialect topic identification (MRC) task. Our team achieved first place in German Dialect identification (GDI) and MRC subtasks 2 and 3, second place in the simplified variant of Discriminating between Mainland and Taiwan variation of Mandarin Chinese (DMT) as well as Cuneiform Language Identification (CLI), and third and fifth place in DMT traditional and MRC subtask 1 respectively. In most cases, the SVM with a flat combination of features performed better than SVM ensembles. Besides describing the systems and the results obtained by them, we provide a tentative comparison between the feature combination methods, and present additional experiments with a method of adaptation to the test set, which may indicate potential pitfalls with some of the data sets.

1 Introduction

Language identification is a text classification task that has been studied extensively in the field of Natural Language Processing. The general concept and common implementations are described in the recent survey by Jauhiainen et al. (2018c). A more challenging task is discerning closely related languages or dialects of the same language. In recent years, the VarDial Evaluation Campaign has organized a multitude of shared tasks on classifying these with textual and spoken data (Malmasi et al., 2016; Zampieri et al., 2017, 2018). This year's VarDial evaluation campaign (Zampieri et al., 2019) featured one rerun (Swiss German dialect identification) and three new closely-related language identification tasks (Mainland vs. Taiwan varieties of Mandarin, Romanian vs. Moldavian, and cuneiform language identification, with the latter covering seven related languages within a wide historical time frame). Our focus has been German dialect identification (GDI) and discriminating between mainland and Taiwan varieties of Mandarin (DMT). However, we submitted predictions for all language identification tasks.

While closely-related languages (or dialects) pose a challenge for language identification, they also provide opportunities for cross-lingual transfer where available resource and tools in one language is adapted to another, similar language variety. This year's evaluation campaign also features two cross-lingual transfer tasks. Namely, cross-lingual morphological analysis (CMA), and cross-lingual topic identification between Romanian and Moldavian (MRC). The CMA is a substantially different task than language identification. However, the MRC subtasks on cross-lingual topic identification can be solved by the very same text classification models used for language identification. Hence, we also participated in the cross-lingual classification subtasks of the MRC.

Our base model is a linear support vector machine (SVM) classifier with sparse character and word n-gram features. These models have been found to be successful in earlier instances of VarDial language identification tasks; in fact, they were found to be more effective than more recent neural classifiers (Çöltekin and Rama, 2016; Clematide and Makarov, 2017; Medvedeva et al., 2017). A successful variation of these linear classifiers is an ensemble of classifiers with different n-gram orders used both for language discrimination (Malmasi and Zampieri, 2017b,a), and native language identification (Malmasi and Dras, 2018). Besides the simple, 'flat' concatenation of

Proceedings of VarDial, pages 54–63
Minneapolis, MN, June 7, 2019 ©2019 Association for Computational Linguistics

the overlapping n-gram features, we also used an ensemble approach in some of the tasks, providing a tentative comparison between these two related methods.

An interesting result of last year's VarDial evaluation campaign was SUKI team's success on Indo-Aryan language identification (Jauhiainen et al., 2018b) and GDI (Jauhiainen et al., 2018a) tasks with a rather large margin, which was likely because of the adaptation mechanism they used at prediction time. We adopted a similar adaptation approach to our SVM systems. Besides the curious difference in the GDI data set last year, the adaptation idea is also a good fit for the cross-lingual topic identification task (MRC).

The remainder of this paper introduces the tasks and data sets, describes our systems, and presents the results obtained followed by a brief discussion.

2 Tasks and Data

2.1 CLI: Cuneiform Language Identification

The provided datasets for Cuneiform Language Identification (Jauhiainen et al., 2019) consisted of a training set and a development set. The training data contained cuneiform texts written in Sumerian (SUX) and six Akkadian dialects: Old Babylonian (OLB), Middle Babylonian peripheral (MPB), Standard Babylonian (STB), Neo Babylonian (NEB), Late Babylonian (LTB), and Neo Assyrian (NEA). The data for the shared task contained only Unicode transcriptions of the documents without token boundaries or any other visual features. The data set exhibited a large class imbalance, ranging from 3 803 instances for Old Babylonian to 53 673 instances for Sumerian. The training data contained a total of 139 421 text samples, while the development set contained 668 lines for each language or dialect.

2.2 DMT: Discriminating between Mainland and Taiwan variation of Mandarin

The Discriminating between Mainland and Taiwan variation of Mandarin Chinese (DMT) task consisted of classifying sentences extracted from news articles into classes of two major Mandarin variations: *Putonghua* (Mainland China) and *Guoyu* (Taiwan). The task has two tracks: traditional and simplified.

In Mandarin Chinese, there are many mutually intelligible regional variations. *Putonghua* and *Guoyu* are more distinguishable in spoken language due to systematic phonetic differences, while they are more ambiguous in written text with no overt morphological, syntactic, and lexical preferences in language use, especially in formal text. It is considered challenging even for native speakers to distinguish between them, and since the shared task data offered only textual information with no phonetic transcription, it was particularly interesting to explore possible solutions to the problem.

In contemporary written Chinese, there are two scripts: traditional and simplified. The only distinction between the two writing systems is the visual form of the characters. As the name suggests, characters in simplified Chinese usually appear simpler than their traditional counterparts, while some are identical which may lead to performance variations based on different system designs. A text in traditional Chinese can always be transformed verbatim into its simplified counterpart without any content change and vice versa. Two corpora, one using traditional script and one simplified, were provided to investigate the performance of the discrimination task on the two different scripts, which will be further discussed in Section 5.

The DMT data comes from the news domain for both varieties. The datasets contained a training and development set for both simplified Chinese (McEnery and Xiao, 2003) and traditional Chinese (Chen et al., 1996). The training set consisted of 18 770 samples for both Chinese varieties, whereas the development set contained 2 000 samples each. The texts contained no punctuation and were (automatically) segmented by the task organizers.

2.3 GDI: German Dialect Identification

As in previous years, the GDI data set is based on the corpus introduced in Samardžić et al. (2016), consisting of samples from four regions around Bern (BE), Basel (BS), Lucerne (LU) and Zurich (ZH). Besides transcriptions of the audio recordings, we were also provided with 400-dimensional i-vectors representing the acoustic features of each sample, and automatically obtained normalization data where words are paired with their standard German spelling. In our submissions, we used the text transcripts and i-vectors.

There were 14 279 training and 4 530 development instances. Both training and development sets included a fair amount of class imbalance.

2.4 MRC: Moldavian vs. Romanian Cross-dialect Topic identification

The MRC task involved discrimination between two closely written language varieties, Romanian and Moldavian, and cross-variety topic classification. The first subtask was a binary classification problem, discriminating between the two language varieties. The second and third tasks required classifying the documents in one variety using training data from the other variety into six topics: culture, finance, politics, science, sports and technology. The second subtask used Moldavian as the source language and Romanian as the target language in the transfer task. Task 3 had the same setup, but the source and target languages were swapped. Topic classification tasks are formulated as multi-class problems (in contrast to multi-label classification common in the field), where each text is assigned to only one class. Named entities in the data set were anonymized.

The training data for subtask 1 consisted of 21 701 texts with a slight class imbalance (11 740 Romanian, 9 961 Moldavian), with a development set of 11 834 instances approximately following the same class distribution. Training sets for subtasks 1 and 2 included 9 961 and 11 740 texts, and development sets included 5 432 and 6 402 texts, respectively. All subtasks shared a test set of 5 918 texts, although subtasks 2 and 3 were evaluated on subsets of the test set. Further information on the data can be found in Butnaru and Ionescu (2019).

3 Methods and experimental setting

Our main submissions were based on two SVM systems that differ in the way they combine the n-gram features: SVM with flat feature combinations and SVM ensembles. We employed both character and word n-gram features. Depending on the task, the character n-grams varied between 1 to 9 and the word n-grams varied from 1 to 3. The features were weighted with either tf-idf or BM25 (Robertson et al., 2009) weighting schemes. The flat combination is similar to Çöltekin and Rama (2016) and the ensemble approach is similar to Malmasi and Dras (2015). Both methods were implemented in Python using the scikit-learn library (Pedregosa et al., 2011).

We also experimented with recurrent neural classifiers and considered a system similar to HeLi (Jauhiainen et al., 2016), which was also used in earlier VarDial evaluation campaigns. However, we only submitted results with the SVM classifiers described in more detail below, and we will limit our discussion to the results obtained by the SVM classifiers.

3.1 SVM with flat combinations of features

For all tasks, we submitted predictions generated by SVM classifiers where a range of overlapping character and word n-grams are combined into a single feature matrix. The features are weighted using BM25, although a plain tf-idf weighing scheme produced similar results on the development set. In all tasks, we optimized the model hyperparameters through random search, using 5-fold cross validation on combined training and development sets. Random search was stopped after approximately 1 000 draws from the space of random parameters, and picking the best average F1-score over the 5 folds. This is simply the same approach taken in a series of earlier VarDial evaluation campaigns (Çöltekin and Rama, 2016; Rama and Çöltekin, 2017; Çöltekin and Rama, 2017; Çöltekin et al., 2018).

Following the adaptation idea used by Jauhiainen et al. (2018a,b) in last year's VarDial evaluation campaign, we also employed an adaptation approach in some of the tasks. At test time, we produced a set of first-level predictions based on the best model tuned for the task on the training/development set, and retrained the model after adding the predictions with high-confidence to the training set. In our case, predictions with high-confidence means the test instances that are farther than a threshold — in this case, 0.50 — from the decision boundary for binary classification, and the instances that are claimed by only one of the one-vs-rest classifiers for the multi-class problems. Intuitively, this is useful for the adaptation subtasks of MRC, and in case the distribution of the test instances diverge from the distribution in the training/development sets.

All tasks we participated in involved text classification. However, the GDI data set also included features extracted from audio samples (i-vectors), as well as normalized spellings of the dialectal words. We did not make use of the normalized spellings, however in our GDI contribution, we used audio features by simply concatenating the i-vectors with the n-gram vectors weighted by BM25, before feeding them to the SVM classifiers. As SVMs are sensitive to the scale of the data, we introduced a weight parameter and searched for its optimum value during tuning.

3.2 SVM ensembles

SVM ensembles are generally considered more robust than single classifiers (Oza and Tumer, 2008). An ensemble system makes use of decisions from multiple classifiers on every input entity. The decisions are congregated through a fusion method, reevaluated, and a final decision is made. There are various fusion methods (Malmasi and Dras, 2015), but the one we chose was *mean probability rule*, an approach that is considered stable and simple (Kuncheva, 2004) as well as resistant to estimation errors (Kittler, 1998). Each classifier returns a prediction with the probability of each test instance belonging to each label. The final decision is the label with the highest average probability.

Each classifier was trained on the standard training set using single n-gram order. We performed binary search on the DMT simplified training development set in the range of [0, 1000] in order to determine the ideal penalty value C. The F1-score increased with increasing C value, and plateaued when $C \geq 100$, so we adopted $C = 100$ as the optimal value. Table 2 lists the score of each classifier using the DMT simplified development set.

Since SVMs separate classes by maximizing the margin from items to the hyperplane (Burges, 1998), there is no natural probabilistic interpretation of the decision function of an SVM classifier. Therefore, we applied the technique of calibration suggested by Platt et al. (1999), a method that maps the outputs of SVM to probabilities, as implemented in the scikit-learn library.

We used grid search to find the optimal combination of n-gram features for each task. For DMT simplified, the final ensemble system we selected utilized five parallel classifiers, each of them generated with different parameters: character-based bigrams, trigrams, 4-grams, 5-grams, and word-based unigrams. For DMT traditional, the combination additionally included character-based unigrams. For GDI, we used character-based bigrams, trigrams, 4-grams, 5-grams, word-based unigrams, and the audio i-vectors.

task (model)	F1-macro	rank	F1-diff
DMT-S (flat)	87.38	2	−1.91
DMT-S (ens.)	84.45	NA	−4.84
DMT-T (flat)	88.44	3	−2.41
DMT-T (ens.)	85.61	NA	−5.24
GDI (flat)	75.93	1	0.52
GDI (ens.)	65.17	NA	−10.76
MRC 1 (flat)	75.73	5	−13.92
MRC 2 (flat)	61.15	1	5.26
MRC 3 (flat)	55.33	1	13.23
MRC 1 (flat)*	96.20	NA	6.70
MRC 2 (flat)*	69.08	NA	7.93
MRC 3 (flat)*	81.93	NA	26.60
CLI (flat)	76.32	2	−0.63

Table 1: Official results obtained by our models on all tasks we participated. The column F1-diff indicates the macro F1-score difference from the top score if the result is not the top score, or the difference from the second best scores otherwise. Our submissions in the MRC task had an error, causing a shift of labels after a certain index. The scores marked with * are post-evaluation results with the gold labels released by the organizers after the evaluation period.

4 Results

We list the results obtained by our systems on the official test sets in Table 1. The results clearly show that the simple linear classifiers we used are competitive with other (best) participating systems. Furthermore, in our experiments, the flat combination often worked better than the ensemble method. However, we do not provide a more conclusive, systematic comparison at this time. In the remainder of this section, we will first describe some of the interesting results in each task, and also present a series of additional experiments with the adaptation method described above.

4.1 DMT

For both DMT tasks, we submitted at least one classifier with a combined feature matrix (flat) and at least one model with parallel classifiers (ensemble). Our submissions with a combined feature matrix using character n-grams of order 1 to 4 combined with word unigrams and bigrams consistently outperformed the parallel classifiers.

In order to improve accuracy for the ensemble, multiple trials were conducted on the development set to determine the best possible combination of features. Most combinations performed similarly, on the order of approximately 87–89 % accuracy with no significant jump in accuracy using any particular combination. However, the most gains were observed when combining a large number of character n-grams with $1 \leq n \leq 5$ and word unigrams. Word bigrams already resulted in a significant loss of accuracy in the SVM ensemble (possibly overfitting due to large number of features, and large C value selected in the earlier step).

Feature Types	n	F1 macro
character	1	77.41
character	2	83.77
character	3	87.19
character	4	86.99
character	5	83.75
word	1	76.63
word	2	33.33

Table 2: F1 scores achieved by SVM with single features, tested on development set (Simplified Chinese)

During development, we observed that training and testing our model on traditional Chinese consistently performed slightly better than training and testing on simplified Chinese. Combining the traditional training set with the simplified training set did not yield any significant gains and in fact slightly hindered the model's performance.

Our flat SVM model placed second for simplified and third for traditional. Other teams also saw higher F1-scores for traditional compared to simplified which suggests that the traditional script carries more information that proves useful in distinguishing between the two dialects. Despite this, our model misidentified the Taiwanese variant roughly twice as often as its Mainland counterpart using both scripts (simplified: 166 vs. 88, traditional: 151 vs. 80).

4.2 GDI

The same models used for DMT were slightly modified for the German Dialect Identification task. Our flat model using character n-grams of order 1 to 5, word unigrams and bigrams, and the i-vector features achieved first place with an F1-score of 75.93, which was very closely followed by the second and third place entries.

The confusion matrix presented in Figure 1 demonstrates that Basel was most easily identified (recall: 91.99). Lucerne was the dialect most often misclassified (recall: 62.41), usually confused with Bern. Consequently, Bern had the lowest precision (69.39) while Basel and Zurich enjoyed the highest (tied with 80.81). This distribution mirrors the results of last year's GDI task (Ciobanu et al., 2018; Ali, 2018; Benites et al., 2018; Barbaresi, 2018).

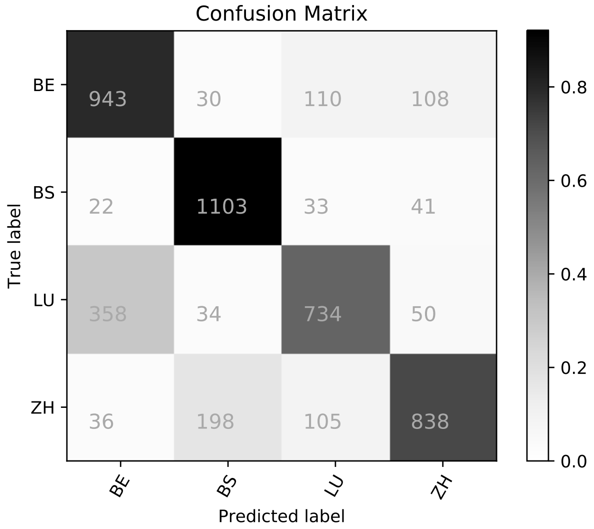

Figure 1: Confusion matrix for GDI. Abbreviation key: Bern (BE), Basel (BS), Lucerne (LU), Zurich (ZH).

In the development set, the SVM ensemble with character n-grams of $2 \leq n \leq 5$, word unigrams, and audio i-vectors outperformed the flat feature combination. The ensemble system yielded an F1-score of 65.35 in comparison to a 44.24 F1-score obtained by the flat combination. This is likely due to the fact that ensemble systems are particularly effective when the individual classifiers are independent, and features from text and audio provide more independent predictions in comparison to the overlapping n-gram features.[1]

4.3 CLI

We submitted predictions using only the flat feature combination for the cuneiform language identification task. Our submission with adaptation came in a close second with an F1-score of 76.32. Since the data did not include any word boundaries, our system combines only character n-grams (of order 1 to 5). We also experimented with two unsupervised segmentation methods (Çöltekin and Nerbonne, 2014; Virpioja et al., 2013). However,

[1] Our official score on the test with the flat combination is higher than the ensemble submission. A potential reason for this discrepancy is an error in our submission identified post-evaluation.

using tokens obtained through both segmentation methods as (additional) features did not improve the results on the development set.

On the CLI data, the adaptation method is highly effective. Our submission with no adaptation performed much worse (53.18 F1-score). We will present more results with adaptation in Section 4.5 and discuss it further in Section 5.

The confusion matrix from our official submission is presented in Figure 2, which depicts some effects of the historical proximity of the languages.

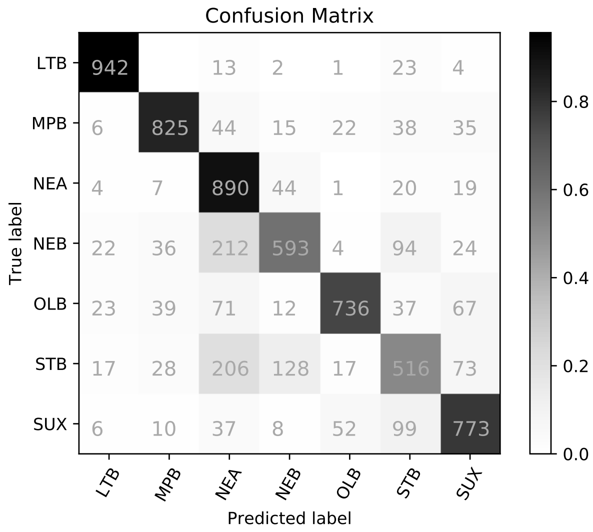

Figure 2: Confusion matrix for CLI. Abbreviations: Late Babylonian (LTB), Middle Babylonian peripheral (MPB), Neo Assyrian (NEA), Neo Babylonian (NEB), Old Babylonian (OLB), Standard Babylonian (STB), Sumerian (SUX).

4.4 MRC

We submitted predictions with only the flat combination for the MRC tasks. Our submissions in this task had an error, causing a shift of labels after a certain index. Despite this shift (with some effort from the organizers to guess the location of the missing predictions) our submissions obtained first rank in subtasks 2 and 3. After rectifying the problem post-evaluation, F1-macro scores increased by up to 30%, reaching 96.20 for subtask 1, 69.08 in subtask 2, and 81.93 in subtask 3. The high rate of success in discriminating between such close linguistic varieties is interesting. However, the primary objective in MRC was cross-lingual learning in the last two subtasks which we discuss further in Section 4.5.

4.5 Adaptation to target data

In this instance of the VarDial evaluation campaign, we employed a method of adaptation to the test data. Among the tasks in which we participated, the clear cases for adaptation are MRC subtasks 2 and 3. These tasks are transfer learning tasks, hence some sort of adaptation is expected to help. In other cases, we do not expect substantial gains from adaptation unless test sets diverge from the training substantially and systematically.

Our official submissions did not always include results from the identical models with and without adaptation, and as such does not clearly indicate the utility of it. Here, we present results from more systematic experiments conducted on the development sets using our SVM model with flat combinations of features. The intuition here is that if the distribution of the test instances diverge from the training set, we can adapt to the test set either by using a small amount of data with gold-labels, or predictions with high confidence at prediction time. The first method (adding gold target data) is not an option during the shared task evaluation. Therefore, we tested both options on the designated development sets. For the second method (adaptation at prediction time), our method is similar to, but simpler than, the method of Jauhiainen et al. (2018a,b). We trained a base classifier on the training data, and re-trained the system after adding the test instances predicted with high-confidence to the training data. For binary tasks, we picked the training instances with a distance greater than 0.50 to the decision boundary. For multi-class classification problems, we picked the instances that are claimed by only one of the one-vs-rest classifiers as confident predictions.

Figure 3 presents five sets of results on all (sub)tasks that we worked on. The first bar in each group represents the average F1-scores obtained with 5-fold cross validation on each training data set. For the rest of the experiments, we split the development set into two equal-sized data sets (after shuffling). The first part is treated as development set, and the second part is treated as test set. The second set of bars (no adapt) represents the F1-scores on the test set (the second part of the respective development sets), after training the model on the training set. The third bar (add) is the first case of adaptation. We add first half of the development set to the training data, and test on the second half. This is compatible in the scenario where we have

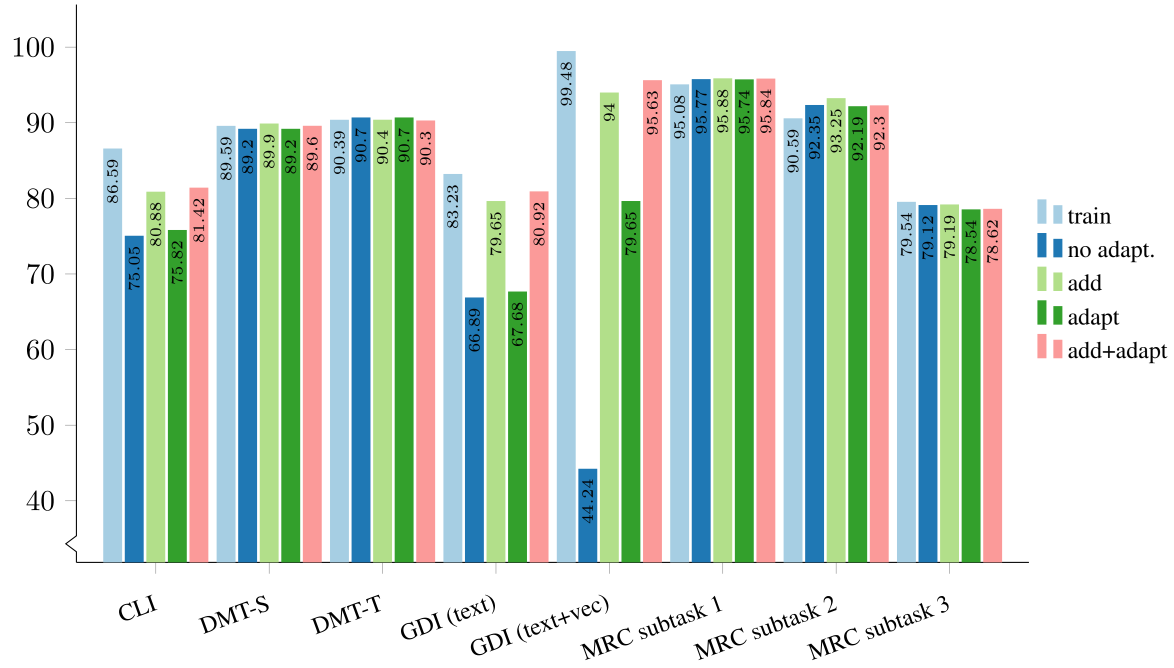

Figure 3: Results of adaptation experiments. The graph presents macro averaged F1-scores of five experiments on each task. 'train' indicates average of 5-fold CV on training set; 'no adapt.' indicates no adaptation, train only on the training set; 'add' indicates adding half of the gold-labeled data from the development set, and testing on the other half; 'adapt' is adaptation during training by adding predictions with high-confidence to the training set and re-training the model; and 'add+adapt' combines the last two options.

a large amount of data from the source domain, with a small amount of data from the target domain. The training instances from the source and target are equally weighted in our experiments. In the fourth set of experiments (adapt), the base classifier is trained on the training data, and testing is done adaptively on the second part of the development set. The final bar (add+adapt) combines the last two. The base system is trained with the combination of the training set and the first half of the development set, and tested on the second part of the development set using adaptation.

The scores illustrated in Figure 3 for both DMT tasks and MRC subtask 1 (language identification) are as expected. The cross-validation scores on the training set are slightly better than scores on the test (part of official development) set, and adaptation options give a slight boost in most cases. In MRC subtask 2, the F1-score on the test set is better than the training set. This is particularly interesting, as this is a language transfer task where the test set is expected to diverge. All scores we obtained in this subtask are also much higher than the (corrected) official test set score (69.08) presented in Table 1. Adaptation, however, seems to help if data with gold labels are added. In MRC subtask 3, which reverses the languages in MRC subtask 2, adaptation does not seem to be useful either.

The results of CLI and, especially, GDI tasks are particularly surprising. In these tasks, adaptation, and especially the addition of gold-labeled data, seem to improve the results drastically. The difference likely indicates a systematic difference between the training and development sets (and possibly test). We provide further discussion of these results for the GDI, in Section 5.

5 Summary and Discussion

Thus far we have described our participation in the VarDial 2019 evaluation campaign, where we participated in all text classification tasks using two variants of linear SVM classifiers. Our systems ranked well among other participants, obtaining first place in some tasks, or following the top result with small differences in others. The results show that simple linear classifiers work well in language identification and cross-dialect topic classification. In most of our experiments, a flat combination of features performed better than ensembles. Furthermore, the adaptation system we used seems to be effective, particularly in some of the tasks. In this section, we present our observations on the

DMT task, and discuss the potential reasons for the effectiveness of adaptation methods.

Observations on the DMT task. The relationship from traditional Mandarin to simplified is generally bijective, but there are some cases where the relation is many-to-one. Thus, a machine is better able to predict using traditional over simplified. Consequently, this explains why our model always produced 1-2% better results with the traditional script. To illustrate this, consider the following example: 「雲」‘cloud’ and 「云」‘speak’ in traditional Mandarin are both written as 「云」in simplified, which indicates that the simplified character「云」carries the meaning of both ‘cloud’ and ‘speak’. In other words, simplified Mandarin has more homonyms, which makes it more difficult for the model to make an accurate prediction.

The texts converted from simplified to traditional are different from traditional to simplified. In traditional Mandarin, both「后」and 「後」can be converted to 「后」in simplified Mandarin. If we convert the word「后面」‘in the back’ from simplified to traditional, it could be either「后面」or「後面」. Hence, we might erroneously select「后面」‘the face of a queen’, where「後面」would be the semantically correct answer. Converting data from traditional to simplified would prevent this type of noise.

Chinese is a language with many compound words, whose tokenization require special attention. Some compounds are used only in Mainland China, but not in Taiwan. However, when split, the individual tokens might all be used in Taiwan, but not the original compound word. Therefore, this would be detrimental to its discrimination accuracy. For example, the word ‘microeconomics’ is 「個體經濟學」in Taiwan, but 「微觀經濟學」on the mainland. It is a compound word composed of 「個體」and「經濟學」in Taiwan and 「微觀」「經濟學」in Mainland China. But we should not categorize 「微觀」and「經濟學」as Mainland Chinese, because when they are treated as two tokens, they are two words that are commonly used in Taiwan. This is not a unique example, and similar cases of segmentations of compounds are likely to have detrimental effects on identification.

Adaptation to test set. Another interesting finding in this work is the impact of adaptation in the CLI and GDI tasks, especially when using the i-vectors. A potential explanation for this is the existence of other systematic variation in the data. For the GDI task, our hypothesis is that the second systematic variation is the (limited number of) speakers. Since the data contains multiple utterances from each speaker (and each speaker speaks only one dialect), a classifier relying on speaker specific features in the training set will also do well on identifying his/her dialect. Such a classifier, then, will have difficulty classifying the utterances from different speakers in the test set.

As a result, the scores of the models with no adaptation in Figure 3 drop drastically when they are trained on the training set, and tested on a test set with utterances from different speakers. On the GDI data, this is true of models using text-only and text and i-vector features. However, it becomes more striking when i-vectors are included, as they are well-known for their ability of speaker identification. Although the model can achieve almost perfect dialect identification on the training data, the F1-score drops to 44.24 when tested on different speakers. The success on the training set and the drop on the test set is less drastic for text-only data. In both cases, the models perform clearly better than random. Hence, the models learn something about the dialects as well. However, the success of our (and other participants') adaptation methods, are likely not (only) finding dialectal differences, but rely more on speaker-specific features by incorporating features of otherwise unknown speakers into the training set.

The experiments presented in Figure 3 also indicate a likely additional source of variation in the CLI data as well. Without more information about the data and its division, the source of this variation is not clear. On the other hand, ineffectiveness of the adaptation method on MRC subtasks 2 and 3 is unexpected. However, we are not able to offer a potential explanation at this time.

Future work. Although the flat feature combination worked better in our experiments here, our experiments are far from conclusive. We intend to extend our work on ensemble models to cover different combination methods and more diverse architectures.

Acknowledgments

Some of the experiments reported were run on a Titan Xp donated by the NVIDIA Corporation.

References

Mohamed Ali. 2018. Character Level Convolutional Neural Network for German Dialect Identification. In *Proceedings of the Fifth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, pages 174–175.

Adrien Barbaresi. 2018. Computationally efficient discrimination between language varieties with large feature vectors and regularized classifiers. In *Proceedings of the Fifth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, pages 167–168.

Fernando Benites, Ralf Grubenmann, Pius von Daniken, Dirk von Grunigen, Jan Deriu1, and Mark Cieliebak. 2018. Twist Bytes - German Dialect Identification with Data Mining Optimization. In *Proceedings of the Fifth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, page 224.

Christopher JC Burges. 1998. A tutorial on support vector machines for pattern recognition. *Data mining and knowledge discovery*, 2(2):121–167.

Andrei Butnaru and Radu Tudor Ionescu. 2019. MOROCO: The Moldavian and Romanian Dialectal Corpus. *arXiv preprint arXiv:1901.06543*.

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.

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.

Keh-Jiann Chen, Chu-Ren Huang, Li-Ping Chang, and Hui-Li Hsu. 1996. SINICA CORPUS : Design Methodology for Balanced Corpora. In *Language, Information and Computation : Selected Papers from the 11th Pacific Asia Conference on Language, Information and Computation : 20-22 December 1996, Seoul*, pages 167–176, Seoul, Korea. Kyung Hee University.

Alina Ciobanu, Shervin Malmasi, and Liviu P. Dinu. 2018. German Dialect Identification Using Classifier Ensembles. In *Proceedings of the Fifth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, page 291.

Simon Clematide and Peter Makarov. 2017. CLUZH at VarDial GDI 2017: Testing a variety of machine learning tools for the classification of Swiss German dialects. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, pages 170–177, Valencia, Spain.

Tommi Jauhiainen, Heidi Jauhiainen, Tero Alstola, and Krister Lindén. 2019. Language and dialect identification of cuneiform texts. In *Proceedings of the Sixth 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. 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. Association for Computational Linguistics.

Tommi Jauhiainen, Krister Lindén, and Heidi Jauhiainen. 2016. HeLI, a Word-Based Backoff Method for Language Identification. In *Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3)*, pages 153–162, Osaka, Japan.

Tommi Jauhiainen, Marco Lui, Marcos Zampieri, Timothy Baldwin, and Krister Lindén. 2018c. Automatic language identification in texts: A survey. *arXiv preprint arXiv:1804.08186*.

Josef Kittler. 1998. Combining classifiers: A theoretical framework. *Pattern analysis and Applications*, 1(1):18–27.

Ludmila I Kuncheva. 2004. *Combining pattern classifiers: methods and algorithms*. John Wiley & Sons.

Shervin Malmasi and Mark Dras. 2015. Language identification using classifier ensembles. In *Proceedings of the Joint Workshop on Language Technology for Closely Related Languages, Varieties and Dialects (LT4VarDial)*, pages 35–43, Hissar, Bulgaria.

Shervin Malmasi and Mark Dras. 2018. Native language identification with classifier stacking and ensembles. *Computational Linguistics*, 44(3):403–446.

Shervin Malmasi and Marcos Zampieri. 2017a. Arabic dialect identification using iVectors and ASR transcripts. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, pages 178–183, Valencia, Spain.

Shervin Malmasi and Marcos Zampieri. 2017b. German dialect identification in interview transcriptions. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, pages 164–169, Valencia, Spain.

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 3rd Workshop on Language Technology for Closely Related Languages, Varieties and Dialects (VarDial)*, Osaka, Japan.

A. M. McEnery and R. Z. Xiao. 2003. The Lancaster corpus of Mandarin Chinese.

Maria Medvedeva, Martin Kroon, and Barbara Plank. 2017. When sparse traditional models outperform dense neural networks: the curious case of discriminating between similar languages. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, pages 156–163, Valencia, Spain.

Nikunj C Oza and Kagan Tumer. 2008. Classifier ensembles: Select real-world applications. *Information Fusion*, 9(1):4–20.

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. *Journal of machine learning research*, 12(Oct):2825–2830.

John Platt et al. 1999. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74.

Taraka Rama and Çağrı Çöltekin. 2017. Fewer features perform well at native language identification task. In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 255–260, Copenhagen, Denmark.

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

Tanja Samardžić, Yves Scherrer, and Elvira Glaser. 2016. ArchiMob – a corpus of spoken Swiss German. In *Proceedings of LREC*.

Sami Virpioja, Peter Smit, Stig-Arne Grönroos, Mikko Kurimo, et al. 2013. Morfessor 2.0: Python implementation and extensions for Morfessor baseline. Technical Report 25/2013.

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)*, Valencia, Spain.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Ahmed Ali, Suwon Shuon, James Glass, Yves Scherrer, Tanja Samardžić, Nikola Ljubešić, Jörg Tiedemann, Chris van der Lee, Stefan Grondelaers, Nelleke Oostdijk, 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)*, Santa Fe, USA.

Marcos Zampieri, Shervin Malmasi, Yves Scherrer, Tanja Samardžić, Francis Tyers, Miikka Silfverberg, Natalia Klyueva, Tung-Le Pan, Chu-Ren Huang, Radu Tudor Ionescu, Andrei 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 (VarDial)*. Association for Computational Linguistics.

Çağrı Çöltekin and John Nerbonne. 2014. An explicit statistical model of learning lexical segmentation using multiple cues. In *Proceedings of EACL 2014 Workshop on Cognitive Aspects of Computational Language Learning*, pages 19–28.