Neural Networks for Cross-domain Language Identification. Phlyers @Vardial 2022

Andrea Ceolin
Università di Modena e Reggio Emilia
ceolin@unimore.it

Abstract

We present our contribution to the Identification of Languages and Dialects of Italy shared task (ITDI) proposed in the VarDial Evaluation Campaign 2022 (Aepli et al., 2022), which asked participants to automatically identify the language of a text associated to one of the language varieties of Italy. The method that yielded the best results in our experiments was a Deep Feedforward Neural Network (DNN) trained on character ngram counts, which provided a better performance compared to Naïve Bayes methods and Convolutional Neural Networks (CNN). The system was among the best methods proposed for the ITDI shared task. The analysis of the results suggests that simple DNNs could be more efficient than CNNs to perform language identification of close varieties.

1 Introduction

In this paper, we present the submissions of Team Phlyers to the Identification of Languages and Dialects of Italy (ITDI) shared task of the VarDial Evaluation Campaign 2022 (Aepli et al., 2022). The campaign is part of a conference series, the Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial), which has reached its ninth edition, six of which have included several shared tasks (Zampieri et al., 2017, 2018, 2019; Găman et al., 2020; Chakravarthi et al., 2021; Aepli et al., 2022). The shared tasks involve the categorization of text documents according to their language variety, typically across different domains. Language identification has received attention in the literature because it is important in the context of machine translation and categorization of social media posts, and several approaches to perform it have been proposed (House and Neuburg, 1977; Dunning, 1994; Bergsma et al., 2012; Lui and Baldwin, 2014; Zubiaga et al., 2016; Jauhiainen et al., 2019c). Most of the VarDial shared tasks invite participants to develop language identification systems in contexts characterized by minimal diversification of the languages involved and low-resource settings, often with lack of data for the domain of interest.

Figure 1: A map of the language varieties of Italy, from Pellegrini (1977).

In the next sections, we briefly describe our submissions for the ITDI shared task.[1]

2 ITDI

The ITDI task involves the classification of sentences from eleven different language varieties from Italy. Five of these varieties - Piedmontese (pms), Lombard (lmo), Ligurian (lij), Emilian-Romagnol (eml), Venetian (vec) - are part of a Northern group composed of Gallo-Italic and Venetan varieties; they are represented in yellow in the *Carta dei Dialetti Italiani*, the reference map drawn by Pellegrini (1977) using a set of isoglosses that define the boundaries of certain morpho-phonological properties. Two of the varieties, Neapolitan (nap) and Tarantino (roa-tara), are part of the Southern group, in pink. Sicilian (scn)

[1]The material developed for this work is available at https://github.com/AndreaCeolin/VarDial2022.

Proceedings of the Ninth Workshop on NLP for Similar Languages, Varieties and Dialects, pages 99–108

is the only representative of the Extreme Southern group, in purple. Friulian (fur) and Ladin (lld) are part of the Northeastern Rhaeto-Romance group, although Pellegrini keeps them separate (in orange and dark green). Finally, Sardinian (sc) is represented in brown.

Training data is provided in the form of Wikipedia dumps containing a total of 233K sentences, while evaluation data is provided in the form of approximately 7K short sentences for seven out of the eleven languages. The test set contains sentences from a subset of the given language varieties, and the classifier is evaluated on sentence level.

An inspection of the development data clearly shows that the sentences are not taken from Wikipedia articles, but from other sources, like literary texts or folktales (see Table 1 for some examples). The sentences in the test dataset also appear to be clearly different from the kind of sentences one expects to find in Wikipedia articles, and we assume that they were taken from domains similar to those used to collect the development sentences.

The fact that the training and validation/testing data come from different domains implies that the task is essentially a cross-domain classification task.

3 Methods

The state-of-the-art methods for language identification are typically inspired by Support Vector Machines (SVM) models (Goutte et al., 2014; Çöltekin and Rama, 2017; Medvedeva et al., 2017; Kreutz and Daelemans, 2018; Benites de Azevedo e Souza et al., 2018; Wu et al., 2019; Çöltekin, 2020) and multinomial Naïve Bayes (NB) models (Barbaresi, 2016; Clematide and Makarov, 2017; Jauhiainen et al., 2019a, 2020; Ceolin and Zhang, 2020; Jauhiainen et al., 2021b), that are trained on features derived from word and character ngrams.

Deep learning methods have also been successfully applied to language identification tasks (Cianflone and Kosseim, 2016; Jaech et al., 2016; Butnaru and Ionescu, 2019; Hu et al., 2019; Tudoreanu, 2019), and in particular several of the most recent VarDial shared tasks have been addressed using transformer models (Bernier-Colborne et al., 2019; Popa and Stefănescu, 2020; Scherrer and Ljubešić, 2020; Zaharia et al., 2020; Jauhiainen et al., 2021b; Zaharia et al., 2021).

While last year we decided to use Convolutional Neural Networks (CNNs) to address the shared tasks (Ceolin, 2021), this year we decided to focus on Deep Feedforward Neural Networks (DNNs), since they represent an alternative approach to language identification.

The reason for this shift of focus is that while CNNs have been the most popular neural architecture used for language identification (Zhang et al., 2015; Conneau et al., 2016; Kim et al., 2016; Jaech et al., 2016), following their success in tasks like image classification and sequence processing, language identification is quite different from such tasks.

While in domains like image classification and sequence processing hard-coding features is not straightforward, in language identification the cues for discriminating among classes are usually words or orthographic/morpheme sequences, which can be directly extracted and used as input features for a simple DNN in the form of word and character ngrams of different size. A CNN instead performs feature extraction indirectly, using fixed-size filters applied to input sequences that have to be of the same length (which is rarely the case for texts), and therefore is less flexible.[2]

For these reasons, comparing these two different approaches can be informative to decide whether CNNs provide any advantage over regular DNNs for language identification.

3.1 DNN

The DNN we used has two hidden layers of size 50, and is trained on a term-frequency matrix of 20K character ngrams in the window [1-5] derived from the training sentences.[3] The DNN is trained with a learning rate of 0.0001 and a batch size of 4 for 20 epochs. The number of parameters is ≈1M. The hyper-parameters and the size of the network were manually selected based on the performance on the evaluation set across different runs. The architecture is visualized in Figure 2.

3.2 CNN

The CNN has two 1-D convolutional layers, one with 256 filters and one with 128 filters, both of size 3 with stride 1, each followed by a max pool layer

[2]Google's LID system, CLD3 (https://github.com/google/cld3), also uses a DNN trained on character ngrams rather than a CNN.

[3]The term-frequency matrix has been extracted using the CountVectorizer method in *sklearn*.

Dataset	Label	Text	Source
train	vec	El Yucatán el ze uno dei 31 Stati del Mèsego, situà inte el sud-est del teritòrio, inte ła parte nord de l'omònema penìzoła. El confina verso nord col Golfo del Mèsego, verso est col Stato de Quintana Roo e verso sud-òvest col Stato de Campeche.	vec.wikipedia.org, "Yucatán"
train	scn	Heaven For Everyone è na canzuni scritta ra Roger Taylor e pubbricatu nto 1988 da li The Cross comu singulu trattu ra l'album Shove It, ru stissu annu.	scn.wikipedia.org, "Heaven For Everyone"
dev	vec	Da seno a mi me par Che no ghe sia rason de barufar	Iliad (version by Luigi de Giorgi)
dev	scn	e Mirimì chi aiutava nnâ mandria na picuredda a figghiari, lassau l'opira a mezzu e si misi a curriri chî manu ntê capiddi, non sapennu chi fari	Storia di Pietracucca (Francesco Lanza)

Table 1: Example sentences from the training and evaluation data. We can see that while the training data contains Wikipedia articles which look like direct translations from other languages, the evaluation data contains sentences from other sources, like poetry or short stories.

(with a window of size 3). Then, it is followed by a fully connected layer of size 50, and is trained with a learning rate of 0.0001 and a batch size of 4 for 20 epochs. The number of parameters is ≈250K. The hyper-parameters and the size of the network were manually selected based on the performance on the evaluation set across different runs. The architecture is visualized in Figure 3. Each input sentences was truncated at 160 characters.

3.3 NB

We also decided to use a NB system as a baseline. The system is trained on the same term-frequency matrix of character ngrams that was used to train the DNN, with alpha=1.

All models were run on Google Colab, with 1 GPU, using the *sklearn* and *tensorflow* libraries.

4 Evaluation

This section summarizes our contributions to the ITDI shared task and the evaluation of our models.

4.1 In-domain Classification

One of the main challenges of the ITDI shared task was to find a proper way to evaluate the performance of the classifiers given that the evaluation set and the test set were not expected to contain the same languages. In a first experiment, we tried a simple in-domain classification task, using only the ≈7K sentences in the evaluation dataset for the seven languages represented in it (henceforth, 'gold' languages) divided in training/test sets using a 80:20 split. We applied minimum normalization: the text was converted to lowercase and numbers and punctuation were removed, with the exception

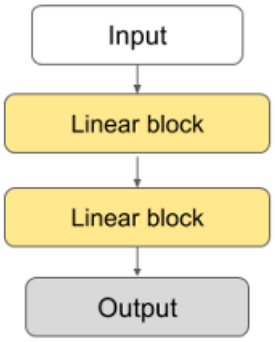

Figure 2: This is the architecture of the DNN model trained for the task. Learning rate: 0.0001, Batch: 4, Epochs: 20. Each hidden layer has size=50.

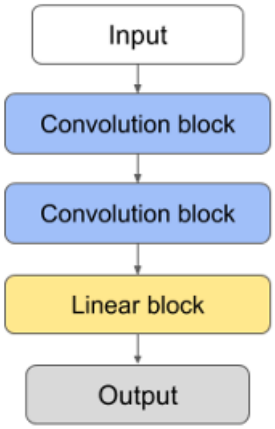

Figure 3: This is the architecture of the CNN model trained for the task. Learning rate: 0.0001, Batch: 4, Epochs: 20. The first convolutional layer has 256 filters of size 3x1, while the second one has 128 of them. Stride: 1. Each layer is followed by a max pool layer, with a window of size 3x1. The fully connected layer has size=50.

Model	Micro F_1 score	Macro F_1 score
DNN	0.994	0.994
Naïve Bayes	0.983	0.984
CNN + data aug. (10ep.)	0.982	0.983
CNN	0.977	0.978

Table 2: Performance of the models on the evaluation set, in-domain classification. The DNN is the model that yields the best performance when using the evaluation set for both training and testing.

Model	Micro F_1 score	Macro F_1 score
Naïve Bayes	0.861	0.554
DNN	0.791	0.520
CNN	0.718	0.471

Table 3: Performance of the models on the evaluation set, cross-domain classification. The Naïve Bayes system is the model that yields the best performance when using the training set for training, and the evaluation set for testing.

of "'", that in these varieties can represent elision of vowels or syllables, and thus is informative. As we can see from Table 2, all models yielded very good results, with the DNN performing best.

We also tried to improve the performance of the CNN by augmenting the training data. Two copies of each sentence were added to the training set with their words shuffled, following the strategy described in Ceolin (2021). Indeed, the strategy allows the network to reach convergence in just 10 epochs and slightly increase its accuracy.[4] Interestingly, increasing the number of parameters of the CNN or the number of epochs did not have the same effect.

These results suggest that the 'gold' languages are well distinguished, and that the amount of sentences in the evaluation set is sufficient to train a robust classifier, assuming that the sentences in the evaluation and test sets belong to the same domain.

4.2 Cross-domain Classification

The second experiment we attempted was a cross-domain classification task. For training, we used a balanced sample of 20K sentences from the 233K training sentences extracted from the Wikipedia dumps using the script recommended by the organizers (Attardi, 2015), while for testing we used the 7K evaluation sentences.[5] In this case, a heavier normalization was required, since the texts contained roman numerals, several proper names of cities/regions, and many different hyperlinks, which had to be removed. From Table 3, we can see that the performance dropped significantly, especially for the neural networks. In particular, many of the predictions (up to 10%, depending on the model and the run) contain one of the four languages which are not represented in the evaluation set (henceforth, 'silver' languages), and so the macro F_1 score is quite low.[6]

4.3 Combining Cross-domain and In-domain Classification

Since the cross-domain classification task turned out to be much harder than the in-domain task, we decided to run a third experiment that was similar to the first one, which relied on the evaluation set for both training and testing. However, after dividing the evaluation set into training/testing sets using a 80:20 split, we augmented the training set using the sentences from the Wikipedia dumps for the four 'silver' languages, in order to cover all languages in the training phase, and we retrained the models. The results are in Table 4.

In this setting, the performance is much better, which means that using the in-domain sentences from the evaluation set instead of the Wikipedia sentences (whenever possible) has a positive effect on the systems. In particular, the improvement in the macro F_1 score is caused by the fact that these systems are more conservative when it comes to the four 'silver' languages: only 1% of the test sentences are assigned to a label that is not part of the evaluation set in all models.

In particular, the DNN and NB systems turned out to be more reliable than the CNNs, both the regular one and the one trained with data augmentation. Interestingly, data augmentation had a clear positive effect on the CNN model (2% for the micro and 9% for the macro F_1 score), but it was still not sufficient to make the CNN reach the accuracy of the other systems.[7]

[4] We explained this behavior with the fact that this prevents the network from focusing on character sequences at word boundaries, i.e. involving space characters in the middle (Ceolin, 2021), which are not informative and can lead to overfitting.

[5] The only reason why we used a subset of the data was to avoid RAM issues. However, we noticed that using more training data did not have any noticeable effect on the results.

[6] In this case we did not try to augment the data for the CNN because the operation was not legitimate, given our access to more training data.

[7] We note that these effects could have been overestimated because, contrary to the in-domain experiment, variation in text length was higher with Wikipedia articles, and so shuffling sentences had the effect of exposing the network to words that

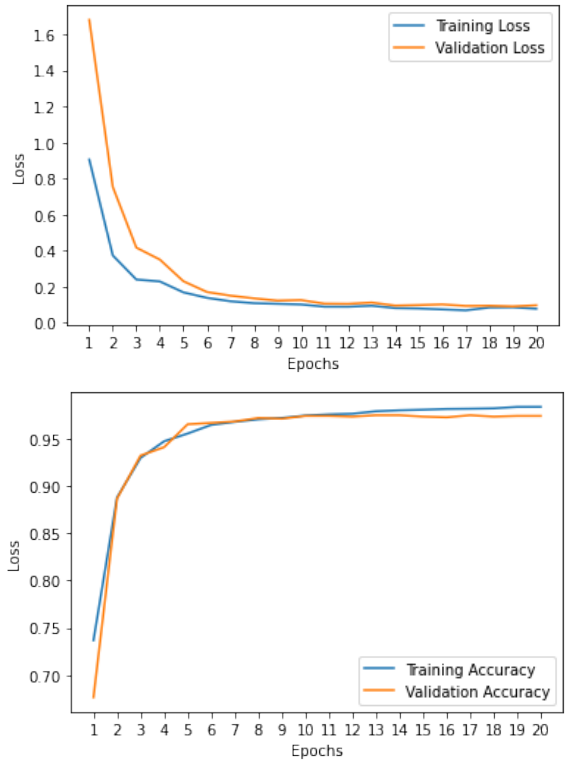

Figure 4: Evaluation of the DNN model on a training set composed of sentences from both the evaluation set (for the seven 'gold' languages) and the Wikipedia dumps (for the four 'silver' languages). Training and validation loss converge after 10 epochs and then decrease together. Accuracy improves up to the 13th/14th epoch, and then stays constant.

For these reasons, we decided to select the DNN as the model of choice for this task. In particular, its high precision, that was highlighted from the results of the in-domain experiment, gives us the option of using some of the sentences from the test set for which the network makes a confident prediction to augment the training data, a form of language model adaptation (Jauhiainen et al., 2018a,b, 2019b), as is explained in the next section. See Figure 4 for the loss and accuracy plots obtained during the evaluation of the DNN.

4.4 Predictions

Table 5 contains the predictions for the 11K sentences in the test set, made by the DNN model which was trained on the evaluation set for the seven 'gold' languages and on the Wikipedia sentences for the four 'silver' languages (in bold).

The most represented among the 'silver' languages is Neapolitan (nap), which is the second most common predicted label. This suggests that the language is present in the test set.

Ladin (lld) and Emilian-Romagnol (eml) are predicted to each represent about 6-7% of the sentences, a number which is not far from the number of sentences we expect to find a priori, especially given that we might expect 'silver' languages to be underpredicted.

The situation with the last 'silver' language, Tarantino (roa-tara) is tricky: the language appears to be quite rare in the test set (0.7%), and an examination of the logit scores associated with the predictions (Figure 5) revealed that Tarantino was the language whose average confidence was the lowest. All the other languages were associated with many more predictions and higher logit scores.

On the other end, Piedmontese (pms), a 'gold' language for which we have several sentences in the evaluation set, is also rare as a prediction, with an occurrence of 0.6%, which is compatible with the ratio of out-of-sample predictions detected in the evaluation experiments.

For these reasons, we decided to remove both Tarantino and Piedmontese, and re-train the classifier to predict only the remaining nine languages.

Model	Micro F_1 score	Macro F_1 score
DNN	0.978	0.761
Naïve Bayes	0.974	0.757
CNN + data aug. (10ep.)	0.951	0.740
CNN	0.929	0.651

Table 4: Performance of the models on the evaluation set, final model.

Label	Labels
vec	3127
nap	1519
scn	1365
fur	1325
lmo	1014
lld	751
eml	700
lij	585
sc	562
roa-tara	79
pms	63

Table 5: Predictions of the DNN for the test dataset. 'Silver' languages in bold.

5 Results

For our first submission, we simply re-trained the DNN excluding Piedmontese and Tarantino, and submitted the predictions on the test set obtained

would have otherwise been truncated. Truncation could thus be the main reason why CNNs underperform in this setting.

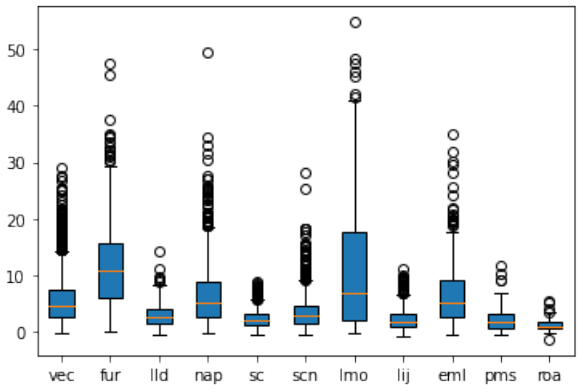

Figure 5: Logit scores associated to each prediction made by our DNN, divided per class.

Team	Model	Weighted F_1 score
SUKI	NB + language adaptation	0.901
Baseline	SVM Char-ngram TFIDF	0.773
Phlyers	**DNN**	**0.694**
ETHZ	Logistic Regression	0.688
Baseline	SVM Unigram TFIDF	0.490
ETHZ	BERT	0.576
Baseline	FastText	0.132

Table 6: Performance of the models on the evaluation of the ITDI task.

in this way. The second and third submission were similar, but we re-trained the network changing the way in which the 'silver' languages were represented: instead of the Wikipedia sentences, we used the label/sentences from the test set for which the predicted label was associated with a high likelihood, following a language model adaptation strategy similar to the one proposed by Jauhiainen et al. (2019b). The main difference is that instead of adding the new predictions, we used them to directly replace the training data for the 'silver' languages, with the aim of obtaining a better representation. We used different likelihood threshold to filter the predictions (>0.90 and >0.95, after transforming the logits into probabilities). On average, the number of predictions per class that were included was quite high, between 75-80%, which seemed a good balance in the trade-off between number of sentences and confidence associated with them.

The overall results of the ITDI shared task are summarized in Table 6.

The best system by far was the SUKI system (Jauhiainen et al., 2021a), a Naïve Bayes-like classifier which performs language adaptation. One of the baselines provided by the organizers, a SVM trained on character ngrams, provided the second best result, with an F_1 score of 0.773. Our best submission, the third one, completes the podium with an F_1 score of 0.694.

Label	Real	Predicted	F_1 score
vec	1139	1642	0.64
nap	2026	2296	0.78
scn	0	1003	0
fur	1323	1283	0.96
lmo	689	921	0.84
lld	2200	1937	0.85
eml	825	746	0.91
lij	2282	626	0.40
sc	0	636	0
roa-tara	603	0	0

Table 7: Predictions of the third submission, a DNN model trained on the evaluation set augmented with the test sentences that, according to the basic DNN model, belonged to the classes not represented in the evaluation set with probability >0.95.

Label	Sub-1	Sub-2	Sub-3
vec	1646 (0.63)	1205 (0.60)	1642 (0.64)
nap	2787 (0.73)	3229 (0.71)	2296 (0.78)
scn	638 (0)	654 (0)	1003 (0)
fur	1248 (0.96)	1299 (0.94)	1283 (0.96)
lmo	816 (0.89)	711 (0.93)	921 (0.84)
lld	2060 (0.86)	2513 (0.86)	1937 (0.85)
eml	1083 (0.80)	964 (0.86)	746 (0.91)
lij	459 (0.32)	268 (0.20)	626 (0.40)
sc	353 (0)	247 (0)	636 (0)
all	0.66	0.64	**0.69**

Table 8: Output of the models on the evaluation of our ITDI task submissions.

The organizers provided us with the results per class, in Table 7. It is apparent that our system overpredicted texts written in Sicilian (scn) and Sardinian (sc), which were actually absent from the data, and underpredicted texts written in Ligurian (lij) and in Tarantino (roa-tara), which was actually present in the test set, contrary to what we were expecting.

A comparison of the predictions of our three submissions, in Table 8, shows that the last submission led to improvements across the board, with one clear exception (Lombard, 'lmo') and a minor one (Ladin, 'lld'). This suggests that language adaptation had a positive impact on the system. However, it also led to the increase of sentences associated with the two languages absent from the test set, which had the effect of countering any substantial improvement, since their presence necessarily ended up hurting the performance of the other classes.

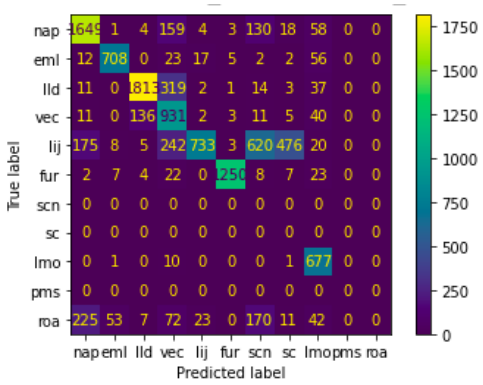

Figure 6: Confusion matrix with the predicted and the gold labels for our third submission.

6 Discussion

Comparing our class results with those of the other teams, the main weakness of our approach turned out not to be the underprediction of Tarantino (roa-tara), with which all the systems struggled, but that of Ligurian ('lij'), which was heavily weighted in the evaluation of the systems, since it was the most common language. An inspection of our evaluation results showed that Ligurian was not among the languages for which we were expecting underprediction. Moreover, Ligurian is a Gallo-Italic language like Lombard and Emilian-Romagnol, but both languages were associated with high F_1 scores in testing, and therefore cannot be responsible for this misclassification.

Since the organizers provided us with the gold labels, we were able to further investigate the behavior of our model by examining the confusion matrix (Figure 6). Some of the patterns were expected: most of the Tarantino (roa-tara) sentences were classified as Neapolitan (nap) or Sicilian (scn), the other two Southern varieties of the sample, and many of the predictions involving Venetian (vec) were instead sentences from Ladin (lld), which is spoken in the same region.

One pattern is instead very peculiar. Sicilian (scn) and Sardinian (sc) were the main responsible for the underprediction of Ligurian (lij), a result which was unexpected, given that the three languages belong to distinct groups, they were all represented in the evaluation set, and were well discriminated in the evaluation phase.

From a linguistic viewpoint, this outcome has an explanation: while Ligurian is a Gallo-Italic language, even classical works like the *Carta dei Dialetti Italiani* by Pellegrini (1977) noticed that there are at least two broad phenomena that the language shares with varieties spoken far from the Gallo-Italic area: the preservation of many word-final vowels, including -u, and the palatalization of [pl] and [bl] clusters. This means that even though Ligurian is clearly a Northern Italy language, an analysis limited to some of its phonological sequences or its morphology could well mistake it for languages spoken outside of the area.

In particular, the first phenomenon was the main responsible for the mistakes in this specific case. Table 9 shows some sentences that were misclassified, from the Ligurian version of Carlo Collodi's *The adventures of Pinocchio*, and in each of them we see morphemes which are typically associated with Southern varieties like Neapolitan and Sicilian and with Sardinian.

It is worth mentioning that the author of the translation published a second version of the text in which the orthographic conventions are different, and *u* is replaced by *o*, which is the case also in the sentences of the evaluation dataset. This variation in orthographic conventions explains why this ambiguity did not emerge in our evaluation phase. There are two reasons why the ambiguity could have affected our results more than those of the other teams. First, in our preprocessing we did not remove proper names from the test sentences because in the evaluation phase they did not seem to affect the results, but clearly having a name like *Pinocchiu* being strongly associated with Southern varieties (the only varieties in which the sequence *cchiu* was present in the training data) heavily affected the performance of our classifier. Second, our classifier was not able to learn that the letter *æ* was unambiguously associated with Northern varieties (only Ligurian and Emilian-Romagnol had it), a cue that should have corrected the mistake.

7 Conclusion

While in some of the previous VarDial evaluation campaigns neural networks yielded the best performance in language identification tasks, (Tudoreanu, 2019; Bernier-Colborne et al., 2019), it was not the case with this shared task, where traditional shallow models like Naïve Bayes and Support Vector Machines performed better, and the DNN model we devised failed to capture important cues like the presence of *æ* in the text.

Even though we were not able to present neural models that reach state-of-the-art performance, we

Target	Prediction	Text	Source
lij	nap	dumandò **u** Pino**cchiu** cun anscêtæ e aff**annu**	Pinocchio (version by Cino Peripateta)
lij	sc	E ti rendime a mæ, e f**emmu** paxe	Pinocchio (version by Cino Peripateta)
lij	scn	Mi suin mariun**ettu**	Pinocchio (version by Cino Peripateta)

Table 9: Sample of sentences written in Ligurian that were misclassified. The phonological sequences/morphemes that are strongly associated with other language varieties (Neapolitan, Sicilian, and Sardinian) are in bold.

still argue that this work makes two contributions.

First, data augmentation has proven to be an effective way to improve the performance of neural networks when the data is limited, a point that we also made last year (Ceolin, 2021) and which has been confirmed throughout the experiments conducted here. Data augmentation has had limited application in NLP (Coulombe, 2018; Kobayashi, 2018; Wei and Zou, 2019), but our experiments suggest that it can play an important role in adapting neural models to the task of language identification in low-resource settings.

Second, DNNs turned out to be more efficient than CNNs to handle language identification. They do not suffer from overfitting in the same way that CNNs do (Ceolin, 2021), they are more flexible, and they yield a better performance.

We hope that our results will encourage the exploration of neural architectures for low-resource language identification and more research in the automatic classification of languages varieties in Italy.

References

Noëmi Aepli, Antonios Anastasopoulos, Adrian Chifu, William Domingues, Fahim Faisal, Mihaela Găman, 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*, Gyeongju, Republic of Korea.

Giuseppe Attardi. 2015. Wikiextractor. https://github.com/attardi/wikiextractor.

Adrien Barbaresi. 2016. An Unsupervised Morphological Criterion for Discriminating Similar Languages. In *Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 212–220, Osaka, Japan.

Shane Bergsma, Paul McNamee, Mossaab Bagdouri, Clayton Fink, and Theresa Wilson. 2012. Language identification for creating language-specific Twitter collections. In *Proceedings of the Second Workshop on Language in Social Media*, pages 65–74, Montréal, Canada.

Gabriel Bernier-Colborne, Cyril Goutte, and Serge Léger. 2019. Improving cuneiform language identification with BERT. In *Proceedings of the Sixth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 17–25, Minneapolis, USA.

Andrei M. Butnaru and Radu Tudor Ionescu. 2019. MOROCO: The Moldavian and Romanian Dialectal Corpus. In *Proceedings of ACL 2019*, pages 688–698.

Andrea Ceolin. 2021. Comparing the performance of CNNs and shallow models for language identification. In *Proceedings of the Eighth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 102–112, Kiyv, Ukraine.

Andrea Ceolin and Hong Zhang. 2020. Discriminating between standard Romanian and Moldavian tweets using filtered character ngrams. In *Proceedings of the Seventh Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 265–272, Barcelona, Spain.

Bharathi Raja Chakravarthi, Mihaela Găman, 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.

Andre Cianflone and Leila Kosseim. 2016. N-gram and Neural Language Models for Discriminating Similar Languages. In *Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 243–250, Osaka, Japan.

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*, pages 170–177, Valencia, Spain.

Çağrı Çöltekin. 2020. Dialect identification under domain shift: Experiments with discriminating Romanian and Moldavian. In *Proceedings of the Seventh Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 186–192, Barcelona, Spain.

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, pages 146–155, Valencia, Spain.

Alexis Conneau, Holger Schwenk, Loïc Barrault, and Yann Lecun. 2016. Very deep convolutional networks for text classification. *arXiv preprint arXiv:1606.01781*.

Claude Coulombe. 2018. Text data augmentation made simple by leveraging NLP Cloud APIs. *arXiv preprint arXiv:1812.04718*.

Ted Dunning. 1994. *Statistical identification of language*. Computing Research Laboratory, New Mexico State University Las Cruces, NM, USA.

Cyril Goutte, Serge Léger, and Marine Carpuat. 2014. The NRC System for Discriminating Similar Languages. In *Proceedings of the First Workshop on Applying NLP Tools to Similar Languages, Varieties and Dialects*, pages 139–145, Dublin, Ireland.

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*, pages 1–14, Barcelona, Spain.

Arthur S House and Edward P Neuburg. 1977. Toward automatic identification of the language of an utterance. I. Preliminary methodological considerations. *The Journal of the Acoustical Society of America*, 62(3):708–713.

Hai Hu, Wen Li, He Zhou, Zuoyu Tian, Yiwen Zhang, and Liang Zou. 2019. Ensemble Methods to Distinguish Mainland and Taiwan Chinese. In *Proceedings of the Sixth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 165–171, Minneapolis, USA.

Aaron Jaech, George Mulcaire, Mari Ostendorf, and Noah A Smith. 2016. A neural model for language identification in code-switched tweets. In *Proceedings of The Second Workshop on Computational Approaches to Code Switching*, pages 60–64, Austin, USA.

Tommi Jauhiainen, Heidi Jauhiainen, and Krister Linden. 2018a. HeLI-based experiments in Swiss German dialect identification. In *Proceedings of the Fifth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 254–262, Santa Fe, USA.

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*, pages 66–75, Santa Fe, USA.

Tommi Jauhiainen, Heidi Jauhiainen, and Krister Lindén. 2019a. Discriminating between Mandarin Chinese and Swiss-German varieties using adaptive language models. In *Proceedings of the Sixth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 178–187, Minneapolis, USA.

Tommi Jauhiainen, Heidi Jauhiainen, and Krister Lindén. 2020. Experiments in language variety geolocation and dialect identification. In *Proceedings of the Seventh Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 220–231, Barcelona, Spain.

Tommi Jauhiainen, Heidi Jauhiainen, and Krister Lindén. 2021a. Naive Bayes-based Experiments in Romanian Dialect Identification. In *Proceedings of the Eighth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 76–83, Kiyv, Ukraine.

Tommi Jauhiainen, Krister Lindén, and Heidi Jauhiainen. 2019b. Language model adaptation for language and dialect identification of text. *Natural Language Engineering*, 25(5):561–583.

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

Tommi Jauhiainen, Tharindu Ranasinghe, and Marcos Zampieri. 2021b. Comparing approaches to Dravidian language identification. In *Proceedings of the Eighth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 120–127, Kiyv, Ukraine.

Yoon Kim, Yacine Jernite, David Sontag, and Alexander M Rush. 2016. Character-aware neural language models. In *Thirtieth AAAI conference on artificial intelligence*, Phoenix, USA.

Sosuke Kobayashi. 2018. Contextual augmentation: Data augmentation by words with paradigmatic relations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 452–457, New Orleans, Louisiana.

Tim Kreutz and Walter Daelemans. 2018. Exploring classifier combinations for language variety identification. In *Proceedings of the Fifth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 191–198, Santa Fe, USA.

Marco Lui and Timothy Baldwin. 2014. Accurate language identification of Twitter messages. In *Proceedings of the 5th workshop on language analysis for social media (LASM)*, pages 17–25, Gothenburg, Sweden.

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, pages 156–163, Valencia, Spain.

Giovan Battista Pellegrini. 1977. *Carta dei dialetti d'Italia*. Pisa: Pacini.

Cristian Popa and Vlad Stefănescu. 2020. Applying multilingual and monolingual transformer-based models for dialect identification. In *Proceedings of the Seventh Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 193–201, Barcelona, Spain.

Yves Scherrer and Nikola Ljubešić. 2020. HeLju@ VarDial 2020: Social media variety geolocation with BERT models. In *Proceedings of the Seventh Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 202–211, Barcelona, Spain.

Fernando Benites de Azevedo e Souza, Ralf Grubenmann, Pius von Däniken, Dirk Von Gruenigen, Jan Milan Deriu, 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*, pages 218–227, Santa Fe, USA.

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, Minneapolis, USA.

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.

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, Minneapolis, USA.

George-Eduard Zaharia, Andrei-Marius Avram, Dumitru-Clementin Cercel, and Traian Rebedea. 2020. Exploring the power of Romanian BERT for dialect identification. In *Proceedings of the Seventh Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 232–241, Barcelona, Spain.

George-Eduard Zaharia, Andrei-Marius Avram, Dumitru-Clementin Cercel, and Traian Rebedea. 2021. Dialect identification through adversarial learning and knowledge distillation on Romanian BERT. In *Proceedings of the Eighth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 113–119, Kiyv, Ukraine.

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*, pages 1–15, 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*, pages 1–17, 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*, pages 1–16, Minneapolis, USA.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in neural information processing systems*, pages 649–657.

Arkaitz Zubiaga, Inaki San Vicente, Pablo Gamallo, José Ramom Pichel, Inaki Alegria, Nora Aranberri, Aitzol Ezeiza, and Víctor Fresno. 2016. TweetLID: a benchmark for tweet language identification. *Language Resources and Evaluation*, 50(4):729–766.