Hierarchical Deep Learning for Arabic Dialect Identification

Gaël de Francony
LSE, EPITA, France
gael.de-francony@epita.fr

Victor Guichard
LSE, EPITA, France
victor.guichard@epita.fr

Praveen Joshi
CIT, Ireland
praveen.joshi@mycit.ie

Haithem Afli
ADAPT Centre, CIT, Ireland
haithem.afli@adaptcentre.ie

Abdessalam Bouchekif
LSE, EPITA, France
abdessalam.bouchekif@epita.fr

Abstract

In this paper, we present two approaches for Arabic Fine-Grained Dialect Identification. The first approach is based on Recurrent Neural Networks (BLSTM, BGRU) using hierarchical classification. The main idea is to separate the classification process for a sentence from a given text in two stages. We start with a higher level of classification (8 classes) and then the finer-grained classification (26 classes). The second approach is given by a voting system based on Naive Bayes and Random Forest. Our system achieves an $F1$ score of 63.02% on the subtask evaluation dataset.

1 Introduction

Online platforms such as Social Media have become the default channel for people to actively participate in the generation of online content in different languages and dialects. Arabic is one of the fastest growing languages used on these platforms. There are many differences between Dialectal Arabic and Modern Standard Arabic which cause many challenges for Arabic language processing. Therefore, identifying the dialect in which posts are written is very important for understanding what has been written over these online platforms.

Shoufan and Alameri (2015) presents a wide literature review of natural language processing for dialectical Arabic. The authors highlighted the huge lack of freely available dialectal corpora which was mentioned in (Zaghouani, 2014).

Although Arabic dialects are related but there are some lexical, phonological and morphological differences between them (Habash et al., 2013; Azab et al., 2013; Attia et al., 2012). Most recently, (Bouamor et al., 2018; Salameh et al., 2018; AL-Walaie and Khan, 2017) started to investigate the problem of the Arabic Dialect Identification with different classification methods.

In this paper, we are describing our work in the same research direction using the MADAR shared task corpus described in (Bouamor et al., 2019). The goal of this task is to classify a given text into one of 26 classes, corresponding to various dialects of Arabic language.

The remainder of this paper is organized as follows. In section 2, we describe the different techniques used in this work. In Section 3, we present our experimental setup and discuss the models and features used as well as our results. Finally, in Section 4 we conclude and give our future directions.

2 System Description

In the next few paragraphs, we will describe the two main methods we used in the MADAR shared task. The first one is based on deep learning with a hierarchical classification of dialects. The second one is based on the combination of Naive Bayes and Random Forest.

2.1 Hierarchical Deep Learning

We address the fine-grained identification of 25 dialects and the Modern Standard Arabic (MSA). Given the number of different dialects and the small size of the data set provided, deep learning algorithms didn't perform well. Our proposed method will aim to handle this problem by decreasing the number of classes the models need to predict. This is achieved using a hierarchical classification similar to the work described in Kowsari et al. (2017).

The classes are separated geographically and represent the dialects of 25 Arabic cities. Some of these dialects are remarkably similar, in particular for cities of the same country/region (Salameh et al., 2018).Some dialects can be clustered to form a larger group. These groups are determined by the geographical distribution of the cities and the similarities between each dialect. This distribution is shown in table 1.

Proceedings of the Fourth Arabic Natural Language Processing Workshop, pages 249–253
Florence, Italy, August 1, 2019. ©2019 Association for Computational Linguistics

Group 1	Group 2	Group 3	Group 4	Group 5	Group 6	Group 7	Group 8
Tunis Sfax	Rabat Fes Algiers	Tripoli Benghazi	Cairo Alexandria Aswan Khartoum	Doha Muscat Riyadh Jeddah Sana'a	Mosul Baghdad Basra	Jerusalem Amman Salt Beirut Damascus Aleppo	MSA

Table 1: Dialect distribution in groups.

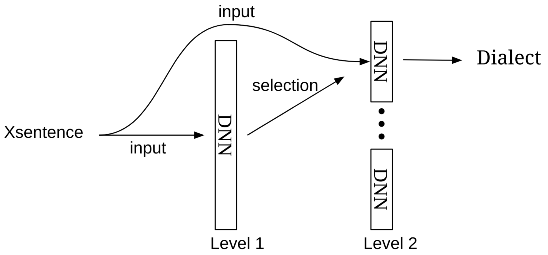

Figure 1: Hierarchical deep learning architecture.

A deep neural network (DNN) is trained to predict a group given a sentence. This model serves as the base for our system. Then for each dialect, a different model is trained. These models make predictions on their respective subset of dialects. Following this technique, two levels of DNNs are defined. First a base whose predictions are used to choose from a set of DNNs. The chosen one is then used to identify the dialect. The system architecture is presented in figure 1.

2.2 Vote Based Probabilistic Classifier

The low size of our data set made statistical models perform much better than the deep learning methods. Our proposed method will take into account the large number of classes by creating two different pipelines. The first one uses a Multinomial Naive Bayes. The second model uses a Random Forest Classifier. These models were implemented using the package scikit-learn (Pedregosa et al., 2011). The pipelines are pre-trained before they are given to the voting classifier. Then, the whole system is trained again to maximize the model performance for the dialects classification task. The data is first given into a count vectorizer then into a TF-IDF tranformer to extract meaningful information on word level. The voting classifier uses a hard voting method to select the model with the correct prediction.

3 Experiments and Results

3.1 Data

We used the data set provided by the MADAR Shared Task. The corpus covers the dialects of 25 Arab cities and the MSA. It is the same data set described in Bouamor et al. (2019) and Salameh et al. (2018). This corpus is composed of 2000 sentences translated to each dialect, with a total of 52000 sentences. We refer to this set as the MADAR corpus. We split this data set evenly between dialects in three parts: 80% constitutes the Train set, 10% the Dev and the last 10% the Test set. In our experiment, we limit the length of the sequences to 40 words and pad the sequences with zeros. For preprocessing we remove all non Arabic characters with the exception of Arabic numbers. To maximize the precision of the hierarchical deep learning system the input of the models is produced by a word2vec. The word2vec we used was trained separately using a database of over 32 million tweets. This data was downloaded using keywords extracted from the MADAR corpus. We used the score of a TF-IDF to find the most relevant words from each dialect. Tweets containing one of these words were downloaded and added to this data set. This way we could ensure a dialectal weight on the word embeddings.

3.2 Hierarchical Deep Learning

In our models, we used Bidirectional Long Short-Term Memory networks (B-LSTM) (Schuster and Paliwal, 1997). It consists of two LSTM networks running in parallel in different directions. Each LSTM generates a hidden representation: the first is generated by reading the input sequence from left to right and the second form right to left. This representations are then combined to compute the output sequence.

The architecture of the hierarchical system is composed of two levels (see the figure 1). The level one is a DNN with three layers: A B-LSTM

Dialect	*Precision*	*Recall*	*F1-sore*
ALE	0.58	0.68	0.63
ALG	0.82	0.75	0.78
ALX	0.78	0.73	0.75
AMM	0.53	0.49	0.51
ASW	0.59	0.55	0.57
BAG	0.58	0.75	0.66
BAS	0.68	0.68	0.68
BEI	0.60	0.69	0.64
BEN	0.70	0.67	0.68
CAI	0.47	0.66	0.55
DAM	0.54	0.56	0.55
DOH	0.64	0.60	0.62
FES	0.70	0.63	0.66
JED	0.61	0.57	0.59
JER	0.61	0.45	0.52
KHA	0.69	0.53	0.60
MOS	0.78	**0.83**	**0.80**
MSA	**0.84**	0.61	0.71
MUS	0.42	0.60	0.49
RAB	0.55	0.75	0.63
RIY	0.61	0.53	0.57
SAL	0.49	0.59	0.53
SAN	0.70	0.81	0.75
SFX	0.77	0.68	0.72
TRI	0.76	0.76	0.76
TUN	0.64	0.77	0.70
ALL	0.64	0.65	0.64

Table 2: Macro average of precision, recall and F1-score for vote based approach (Higher is better).

Model	F1
Deep learning	0.56
Hierarchical Deep Learning	0.58
Voting Classifier	**0.64**

Table 3: F1-score summary (higher is better).

Model	Precision	Recall	F1-score
G1	0.78	0.88	0.83
G2	0.82	0.89	0.85
G3	0.69	0.80	0.74
G4	0.82	0.79	0.81
G5	0.66	0.74	0.70
G6	0.84	0.80	0.82
G7	0.89	0.77	0.83
MSA	0.76	0.71	0.73
Avg.	0.79	0.80	0.78

Table 4: Hierarchical system level 1 precision.

of 128 neurons followed by a fully-connected layer of size 64 and a fully-connected layer of size 8 with softmax activation for the output. The level two is a set of 7 DNNs. For each of this models the size of the layers and the type of Recurrent Neural Network (RNN) units used is different. This is done in order to adapt each model to the number of classes it has to handle as well as to have a proportional number of parameters with the size of the groups data set. The models utilize the following pattern: They are composed of three layers. The first is a RNN layer, either B-LSTM or a B-GRU with a size ranging between 32 and 64 units. Then a fully-connected layer of size ranging between 32 and 64. Finally a fully-connected layer with softmax activation for the output.

All models were trained using the following parameters: batch size = 100, learning rate = 0.001, $\beta 1 = 0.9$, $\beta 2 = 0.999$, decay = 0. The cost function used was the cross entropy. Two gradient descent optimizers where used for training: the RMSProp and the Adamax. To metric the possible improvement of this system we compare the results with a baseline. This baseline is a deep neural network with a similar architecture as the ones found in the hierarchical system.

3.3 Vote Based Probabilistic Classifier

The statistical method performed much better than the Deep learning method. In this section we describe the pipeline using different parameters. To define the accuracy we used the F-1 macro average score. By changing parameters of each pipeline, our results change drastically. We found that for the Naive Bayes the $alpha$ at 0.3 was giving the best performance. For the Random Forest Classifier (RFC), random states set to 2 was also giving the best results. Using 250 estimators and a 200 depth, the RFC was performing the best, leading up to a 4% increase in F1-score.

3.4 Results

The table 5 shows the result of each DNN in the hierarchical system. We notice good performance for some groups such as G3 and G2. However, the improvement in accuracy is not as substantial in most of the groups. Notably the performance of the seventh group only reaching a score of 0.55. This translates to a poor performance on the overall system. We see in table 3 that the hierarchical separation of dialects outperforms the simpler DNN by only 1.4%. Both models can have trou-

Model	Class	Precision	Recall	F1-score	Model	Class	Precision	Recall	F1-score
G1	TUN	.65	.76	.70	G5	DOH	.74	.68	.71
	SFX	.80	.69	.74		MUS	.69	.73	.71
	Avg.	.72	.73	.72		RIY	.61	.60	.61
G2	RAB	.69	.70	.70		JED	.69	.73	.70
	FES	.65	.68	.66		SAN	.77	.75	76
	ALG	.89	.83	.86		Avg.	.70	.70	.70
	Avg.	.74	.74	.74	G6	MOS	.84	.85	.85
G3	TRI	.89	.86	.87		BAG	.73	.65	.69
	BEN	.86	.88	.87		BAS	.60	.66	.63
	Avg.	.87	.87	.87		Avg.	.72	.72	.72
G4	CAI	.58	.64	.61	G7	JER	.55	.47	.51
	ALX	.79	.71	.75		AMM	.53	.49	.51
	ASW	.60	.63	.61		SAL	.58	.61	.59
	KHA	.86	.84	.85		BEI	.58	.70	.63
	Avg.	.71	.70	.71		DAM	.50	.46	.48
						ALE	.58	.64	.61
						Avg.	.56	.55	.55

Table 5: Hierarchical system level 2 precision.

Sentence	Prediction	Label
نحب كاس ماء باش نشرب الدواء ، يعيشك I want a glass of water to take my medicine, please.	SFX	TUN
نحب كاس ماء باش ناخذ الدواء ، يعيشك I want a glass of water to drink my medicine, please.	SFX	SFX

Table 6: Voting classifier predictions on close sentence of similar dialects.

ble on similar dialects. For example, the first two sentences of table 6, are very similar. The first one is from *Tunis* whereas the second one is from *Sfax* which both belong to the same group. Because of this similarity, the models cannot make a correct distinction and often miss predict the correct label. Nonetheless, the statistical method provides good result when dialects are very close. Tunis and *Sfax* have both a good F1-score, even with some confusion due to similar sentences. However it struggles to identify dialects such as *Mosul* (MOS), *Cairo* (CAI) and *Salt* (SAL) which have a very low precision (table 2). The results can be explained by the fact that the amount of data available was very low which can lead to an overfitting of the deep learning model. The voting classifier perform 9% better (table 3).

4 Conclusion and Future Work

In this paper, we propose to use two different methods for Arabic dialect identification: the Hierarchical Deep Neural Network and the Hard Voting Classifier. The hierarchical model uses two levels of DNNs where the first one predicts the group of a dialect, and the second one predicts the dialect according to the previous prediction. The method based on a statistical model is composed of a Multinomial Naive Bayes and a Random Forest Classifier connected by a Hard Voting Classifier. This model outperformed the F1-score results of the Hierarchical Deep Neural Network.

In the future, we plan to work on the combination of two neural networks. The output of the first model will be a vector composed of probabilities for each group. The second one, will take as input the sentence as well as the output of the previous model as a new feature.

Acknowledgments

This research is partially supported by EPITA Systems Laboratory (www.lse.epita.fr) and Science Foundation Ireland through ADAPT Centre (Grant 13/RC/2106) (www.adaptcentre.ie) at Cork Institute of Technology.

References

Mona Abdullah AL-Walaie and Muhammad Badruddin Khan. 2017. Arabic dialects classification using text mining techniques . In *Proceedings of the International Conference on Computer and Applications (ICCA)*, pages 325–329, Dubai, United Arab Emirates.

Mohammed Attia, Pavel Pecina, Younes Samih, Khaled Shaalan, and Josef van Genabith. 2012. Improved spelling error detection and correction for arabic. In *Proceedings of COLING 2012*, pages 103–112.

Mahmoud Azab, Houda Bouamor, Behrang Mohit, and Kemal Oflazer. 2013. Dudley north visits north london: Learning when to transliterate to arabic. In *Proceedings of NAACL-HLT 2013*, pages 439–444.

Houda Bouamor, Nizar Habash, Mohammad Salameh, Wajdi Zaghouani, Owen Rambow, Dana Abdulrahim, Ossama Obeid, Salam Khalifa, Fadhl Eryani, Alexander Erdmann, and Kemal Oflazer. 2018. The MADAR Arabic Dialect Corpus and Lexicon. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan.

Houda Bouamor, Sabit Hassan, and Nizar Habash. 2019. The MADAR Shared Task on Arabic Fine-Grained Dialect Identification. In *Proceedings of the Fourth Arabic Natural Language Processing Workshop (WANLP19)*, Florence, Italy.

Nizar Habash, Ryan Roth, Owen Rambow, Ramy Eskander, and Nadi Tomeh. 2013. Morphological analysis and disambiguation for dialectal arabic. In *Proceedings of NAACL-HLT 2013*, pages 426–432.

Kamran Kowsari, Donald E. Brown, Mojtaba Heidarysafa, Kiana Jafari Meimandi, Matthew S. Gerber, and Laura E. Barnes. 2017. Hdltex: Hierarchical deep learning for text classification. In *16th IEEE International Conference on Machine Learning and Applications, ICMLA 2017, Cancun, Mexico, December 18-21, 2017*, pages 364–371.

Fabian Pedregosa, Gael 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 Edouard Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*.

Mohammad Salameh, Houda Bouamor, and Nizar Habash. 2018. Fine-Grained Arabic Dialect Identification. In *Proceedings of the 27th International Conference on Computational Linguistic*, page 13321344, Santa Fe, New Mexico, USA.

Mike Schuster and Kuldip K. Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*.

Abdulhadi Shoufan and Sumaya Alameri. 2015. Natural language processing for dialectical arabic: A survey. In *ANLP@ACL*.

Wajdi Zaghouani. 2014. Critical survey of the freely available arabic corpora. In *International Conference on Language Resources and Evaluation (LREC'2014), OSACT Workshop*.