The University of Tripoli at NADI 2023 shared task: Automatic Arabic Dialect Identification is Made Possible

Abdusalam F A Nwesri
University of Tripoli
Tripoli - Libya
a.nwesri@uot.edu.ly

Nabila A S Shinbir
Tripoli College of Sci. & Tech.
Tripoli - Libya
shinbir@tcst.edu.ly

Hassan A H Ebrahem
University of Tripoli
Tripoli - Libya
h.ebrahem@uot.edu.ly

Abstract

In this paper we present our approach towards Arabic Dialect identification which was part of the The Fourth Nuanced Arabic Dialect Identification Shared Task (NADI 2023). We tested several techniques to identify Arabic dialects. We obtained the best result by fine-tuning the pre-trained MARBERTv2 model with a modified training dataset. The training set was expanded by sorting tweets based on dialects, concatenating every two adjacent tweets, and adding them to the original dataset as new tweets. We achieved 82.87 on F1 score and we were at the seventh position among 16 participants.

1 Introduction

Arabic dialects are different spoken versions of Modern Standard Arabic (MSA) which become to increasingly emerge in a written format recently. Although Arabic dialects have common linguistic features with MSA, they have different features where NLP tools used for MSA fail to work properly. There are more than 27 Arabic dialects (Elgabou and Kazakov, 2017) which need different NLP techniques than those used for MSA. It was proven that NLP tools for MSA is less efficient with Arabic dialects (Khalifa et al., 2016). As such, it is crucial to identify a dialect version in order to properly apply proper NLP techniques on it.

Arabic dialect identification is very challenging for several reasons. First, Arabic dialects are all originating from MSA and share common features and words. As MSA is the formal language of writing across Arabic countries, writing dialectal phrases are usually mixed with MSA complete phrases. Furthermore, dialectal Arabic has no official spelling standards and usually written differently by different people (Darwish et al., 2021).

Second, With the absence of short vowels (diacritics) in Arabic text, it is hard to know the phrase dialect, for example, the word إنتِ /enti/ (you) in Tunisian dialect is used to address both a Masculine or a feminine third person, while إنتَ /enta/ is used to address a masculine and إنتِ /enti/ to address a feminine third person in several other dialects, while in MSA أنتَ /anta/ and أنتِ /anti/ are used respectively for the same purpose.

Third, tweets are usually short and in many cases it is hard not only for a learning model, but for an Arabic reader to guess the dialect of the tweet based on its words.

Previous work on Arabic dialect identification were mostly carried out through the Nuanced Arabic Dialect Identification (NADI) shared tasks series (Abdul-Mageed et al., 2020, 2021b, 2022). The goal of these shared tasks is to improve dialect identification and other dialect processing tasks such as sentiment analysis and machine translation from dialects to MSA. The organizers provide required resources such as datasets to participants who carry research on those tasks. The forth Nuanced Arabic Dialect Identification (Abdul-Mageed et al., 2023) has three subtasks:

- Subtask 1 (Closed Country-level Dialect ID): dialect identification using provided datasets only. No External datasets should be used.
- Subtask 2 (Closed Dialect to MSA MT): Sentence-level machine translation from Egyptian, Emirati, Jordanian, and Palestinian dialects to MSA using only provided training data.
- Subtask 3 (Open Dialect to MSA MT): Sentence-level machine translation from Egyptian, Emirati, Jordanian, and Palestinian dialects to MSA using provided training data and any publicly available datasets.

We participated in Subtask 1 only. We tested several machine learning and deep learning models which we report in this paper.

Proceedings of the The First Arabic Natural Language Processing Conference (ArabicNLP 2023), pages 620–624
December 7, 2023 ©2023 Association for Computational Linguistics

Dataset	Type	Dialects	Tweets
MADAR-2018	Imbalanced	15	40K
NADI-2020	Imbalanced	17	19.3K
NADI-2021	Imbalanced	17	19.7K
NADI-2023	balanced	18	18K

Table 1: Subtask 1 training datasets provided by NADI 2023.

The remaining part of this paper is structured as follows: Section 2 describes the data used in our experiments, Section 3 describes our experiments and proposed systems, and in Section 4 we present our results proceeded by our discussions and conclusion.

2 Data

For Subtask 1, the organizers provided a 23.4k tweets dataset that covers 18 dialects. the dataset is split into 18k training set, 1.8k development set and 3.6k test set. Extra datasets was also provided and can be used by participants. Particularly, data used in previous NADI competitions plus the MADAR dataset (Bouamor et al., 2018). As a closed-country subtask, participants were not allowed to use other external data to train their systems. Datasets and their size are presented in Table 1.

3 Experiments

We run several experiments using both machine learning and deep learning models. We determined our baseline and officially submit the best three outputs of our systems to be scored on the leaderboard.

3.1 Machine Learning Models

We tested several Machine Learning classifiers, namely: Multi-layer perceptron classifier (MLPClassifier), Support Vector Machines (SVC), Naive Bayes classifier for multivariate Bernoulli models (BernoulliNB), and Naive Bayes classifier for multinomial models (MultinomialNB) (Pedregosa et al., 2011). For each model, we calculate the Accuracy (A), Precision (P), Recall (R), and the normal F1-measure. We obtained best results on the original training dataset after normalizing text and removing non-Arabic characters. Results are shown in Table 2.

We also removed a list of known stopwords in Arabic and used the Snowball stemmer [1] on the original datasets, but results dropped down in both cases. Table 3 shows the results when using stopwords and Table 4 shows results using both stopwords and stemming.

Classifier	F1	A	P	R
SVC	0.60	0.61	0.61	0.60
MLPClassifier	0.62	0.63	0.62	0.62
MultinomialNB	**0.63**	**0.64**	0.64	**0.64**
BernoulliNB	0.58	0.56	**0.67**	0.56

Table 2: Results obtained using Machine Learning classifiers on the training datasets.

Classifier	F1	A	P	R
SVC	0.59	0.58	0.60	0.58
MLPClassifier	0.61	0.61	0.61	0.61
MultinomialNB	**0.62**	**0.62**	0.63	lbf 0.62
BernoulliNB	0.55	0.53	**0.66**	0.53

Table 3: Results obtained using Machine Learning classifiers on the training datasets when removing stopwords.

3.2 Transformer Based Models

It was reported that deep learning techniques are superior to machine learning models. The introduction of transformers based approaches have significantly improved results of NLP tasks such as text classification (Chang et al., 2020). Transformers allow building proficient language models that can be fine-tuned for a specific task. The introduction of Bidirectional Encoder Representations from Transformers (BERT) model (Devlin et al., 2019) by google AI Language resulted in the stat-of-the-art results in a wide variety of NLP tasks. Several versions based on this model have been developed for Arabic Language including AraBERT (Antoun et al., 2021) and MARBERT (Abdul-Mageed et al., 2021a). Results in (Abdul-Mageed et al., 2021a) show that MARBERTv2 was superior to ARBERT, and AraBERT in an Arabic dialect iden-

Classifier	F1	A	P	R
SVC	0.57	0.56	0.58	0.56
MLPClassifier	0.55	0.55	0.56	0.55
MultinomialNB	**0.59**	**0.59**	0.60	**0.59**
BernoulliNB	0.55	0.53	**0.62**	0.53

Table 4: Results obtained using Machine Learning classifiers on the training datasets when removing stopwords and using the Snowball stemmer.

[1] https://pypi.org/project/snowballstemmer/

Model	F1	A	P	R
MARBERTv2	**84.47**	**84.39**	**84.87**	**80.39**
arabertv02	79.31	79.22	79.62	79.22

Table 5: Baseline results by fine-tuning both MARBERTv2 and bert-base-arabertv02-twitter models using the training and the development datasets.

Model	F1	A	P	R
MARBERTv2	**80.12**	**80.00**	**80.60**	**80.00**
arabertv02	76.44	76.44	76.75	76.44

Table 6: The effects of pre-processing tweets on the Baseline results.

tification task. We decided to use MARBERTv2 and AraBERT as our baseline models since they were trained on different datasets and were reported to achieve better result than other models.

3.3 Baseline

We run the script provided by the organizers and fine-tuned the "UBC-NLP/MARBERTv2" and the "aubmindlab/bert-base-arabertv02-twitter" models with the initial following parameters: maximum sequence length is set to 256, training batch is set to 32, learning rate is set to 1e-5 ,and number of epochs is set to 3. We used the training dataset for training and the development set for testing. Scores are then calculated using Accuracy (A), macro average precision (P), macro average recall (R), and macro average F1 (F1) using the provided script. The identification scores is shown in Table 5.

We run several experiments using the baseline models in order to obtain better results than the baseline.

3.4 Pre-processing

We pre-processed the training and the development datasets by removing any non-Arabic characters including emojis, and URLs from tweets; reducing repeated characters to two occurrences; and normalizing the different shapes of Arabic letters such as "آاأإ", "ي", and "ة" to "ا", "ى", and "ه" respectively.

The pre-processed datasets are used to fine-tune our baseline models. This step negatively affected our baseline. Results are shown in Table 6.

3.5 Stop-words Removal

Based on the idea that dialects share the same words originated from MSA, we calculated the frequency of the top 50 words in the training dataset and considered them as our stopwords list. Applying stopwords removal on our baseline decreased our scores as shown in Table 7.

Model	F1	A	P	R
MARBERTv2	**83.19**	**83.11**	**83.36**	**83.11**
arabertv02	78.12	78.06	78.28	78.06

Table 7: The effects of stopwords removal on the Baseline results.

3.6 Tweets Expansion

Our officially reported results came by increasing the tweets length. The idea comes from the fact that a human who reads one sentence, might not be able to recognize a writer's dialect until reading another one. As tweets are usually short with a minimum of three words in the case of our dataset, we made new longer tweets by sorting tweets based on their dialect, and then combining every two adjacent tweets belong to the same dialect together adding the combination to the dataset. The new dataset contains 35898 tweets with a maximum tweet length of 540.

We fine-tuned the pre-trained "UBC-NLP/MARBERTv2" model using the new generated dataset. We set the maximum sequence length to 512, the training batch to 32, and number of epochs to 3. We used the default values for the learning rate. The model was first fine-tuned on a 16GB RAM with core i5 processor. It took around 6 hours to complete. However, using the google Colab T4 GPU (Bisong, 2019), it only took 30 minutes to finish. This technique achieved the best score that was above our baseline. The results are shown in Table 8 as UoT-1 (UoT stands for the University of Tripoli, the name of our team).

We have also run the same experiment (labeled UoT-2) on the same dataset, however, we applied the above mentioned pre-processing technique on the new dataset. This action caused scores to drop below the baseline.

The third run we submitted (Uot-3) is similar to UoT-1, however, the fine-tuning was done using the "aubmindlab/bert-base-arabertv02-twitter" pre-trained model.

We finally run the unlabeled testset against our models and submitted our predictions to leaderboard. Table 9 shows the results of our system using the testset as officially reported by the organisers.

Run	F1	A	P	R
UoT-1	**84.70**	**84.67**	**85.01**	**84.67**
UoT-2	80.64	80.61	80.93	80.61
UoT-3	80.38	80.39	80.54	80.39

Table 8: Results obtained using tweet expansion using the training and Development datasets.

Run	F1	A	P	R
UoT-1	82.87	82.86	83.17	82.68
UoT-2	80.70	80.69	81.18	80.69
UoT-3	74.45	74.44	75.01	74.44

Table 9: Official results in the leaderboard using the output of our systems with the unlabeled testset.

Table 10 shows our best result among the participating teams.

Team	F1	A	P	R
NLPeople	87.27	87.22	87.37	87.22
rematchka	86.18	86.17	86.29	86.17
Arabitools	85.86	85.81	86.10	85.81
SANA	85.43	85.39	85.60	85.39
Frank	84.76	84.75	84.95	84.75
ISL-AAST	83.73	83.67	83.87	83.67
UoT	82.87	82.86	83.17	82.86
AIC	82.37	82.42	82.57	82.42
Cordyceps	82.17	82.14	82.57	82.14
DialectNLU	80.56	80.50	80.92	80.50
Mavericks	76.65	76.47	77.43	76.47
exa	70.72	71.03	72.26	71.03
IUNADI	70.22	70.78	71.32	70.78
NAYEL	63.09	63.39	63.30	63.39
ustdb	62.51	62.17	63.07	62.17
Frau. IAIS	29.91	33.14	38.47	31.39

Table 10: The leaderboard showing our scores in the seventh position (UoT) among participating teams.

4 Discussion

Dialect Identification of a written text is uneasy task. By going through tweets in the development dataset, We found a considerable overlap between regional dialects which is natural, for example Gulf dialects usually overlap and are miss judged by language models. for example, Saudi-Arabian dialect overlaps with Qatar, UAE, and Omani dialects. And Maghrebi dialects such as Tunisian are falsely judged as Algeria and Libyan tweets only; and Levantine dialects such as Syrian are falsely judged as Lebanese, Jordanian, and Palestinian tweets. The best judgement was achieved on Moroccan dialect with only 3 tweets judged as Tunisian and one as Palestinian. False predicted tweets are usually short and are hard for a human to judge. For instance, "سكر الباب وراك" meaning "close the door behind you", is a Kuwaiti tweet which is falsely judged as Egyptian. This tweet can also be Libyan and it is hard to detect its origin dialect. That is why our approach was beneficial in clarifying such tweets. Expanding tweets should be explored further. for instance expanding the dataset with a combination of only shorter tweets within the same dialect.

We expected that pre-processing would improve identification as it cleans text, however, for dialects it did not. After deep analysis of the training dataset, we realized that removing none Arabic characters and normalization should be handled carefully as there are several Arabic tweets written in Farsi characters which fall out of the range of Arabic characters. For example removing characters such as "گ" which is used to represent "ك" in the word "ملگت" would leave the word "مل ت" in the tweet. Such mistake should be corrected by normalizing the letter "گ" to "ك" in the tweets.

5 Conclusions

We used several machine learning classifiers and pre-trained language models to identify Arabic dialects. We also showed the affects of pre-processing, stemming and sotpwords removal on the identification results. our best results are obtained using two pre-trained Models namely: the MARBERTv2 Model and the AraBERT model. We fine-tuned those models with an expanded version of the training dataset. This approach resulted in improving our baseline and put us in the seventh position among 16 participating teams in the Fourth Nuanced Arabic Dialect Identification Shared Task.

6 Limitations

Identifying Arabic dialects is a hard task as dialects follow no standards in their structure. They also share MSA phrases due to the fact that MSA is the formal written language in the Arabic world. Our approach of extending tweets improves dialect detection, however, long tweets on a large dataset requires large memory and computing power. For example, when changing the setting of the maximum sequence length to 512 and using the combination of all datasets provided by the organizers for training, our models crashed due to memory shortage. This was overcome by limiting the tweets length to 256 to allow the model to run without crashing.

References

Muhammad Abdul-Mageed, AbdelRahim Elmadany, and El Moatez Billah Nagoudi. 2021a. ARBERT & MARBERT: Deep bidirectional transformers for Arabic. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7088–7105, Online. Association for Computational Linguistics.

Muhammad Abdul-Mageed, AbdelRahim Elmadany, Chiyu Zhang, ElMoatez Billah Nagoudi, Houda Bouamor, and Nizar Habash. 2023. NADI 2023: The Fourth Nuanced Arabic Dialect Identification Shared Task. In *Proceedings of The First Arabic Natural Language Processing Conference (ArabicNLP 2023)*.

Muhammad Abdul-Mageed, Chiyu Zhang, Houda Bouamor, and Nizar Habash. 2020. NADI 2020: The first nuanced Arabic dialect identification shared task. In *Proceedings of the Fifth Arabic Natural Language Processing Workshop*, pages 97–110, Barcelona, Spain (Online). Association for Computational Linguistics.

Muhammad Abdul-Mageed, Chiyu Zhang, AbdelRahim Elmadany, Houda Bouamor, and Nizar Habash. 2021b. NADI 2021: The second nuanced Arabic dialect identification shared task. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 244–259, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.

Muhammad Abdul-Mageed, Chiyu Zhang, AbdelRahim Elmadany, Houda Bouamor, and Nizar Habash. 2022. NADI 2022: The third nuanced Arabic dialect identification shared task. In *Proceedings of the The Seventh Arabic Natural Language Processing Workshop (WANLP)*, pages 85–97, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Wissam Antoun, Fady Baly, and Hazem Hajj. 2021. Arabert: Transformer-based model for arabic language understanding.

Ekaba Bisong. 2019. *Google Colaboratory*, pages 59–64. Apress, Berkeley, CA.

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. European Language Resources Association (ELRA).

Wei-Cheng Chang, Hsiang-Fu Yu, Kai Zhong, Yiming Yang, and Inderjit Dhillon. 2020. Taming pretrained transformers for extreme multi-label text classification.

Kareem Darwish, Nizar Habash, Mourad Abbas, Hend Al-Khalifa, Huseein T. Al-Natsheh, Samhaa R. El-Beltagy, Houda Bouamor, Karim Bouzoubaa, Violetta Cavalli-Sforza, Wassim El-Hajj, Mustafa Jarrar, and Hamdy Mubarak. 2021. A panoramic survey of natural language processing in the arab world.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Hani Elgabou and Dimitar Kazakov. 2017. Building dialectal Arabic corpora. In *Proceedings of the Workshop Human-Informed Translation and Interpreting Technology*, pages 52–57, Varna, Bulgaria. Association for Computational Linguistics, Shoumen, Bulgaria.

Salam Khalifa, Nizar Habash, Dana Abdulrahim, and Sara Hassan. 2016. A large scale corpus of Gulf Arabic. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 4282–4289, Portorož, Slovenia. European Language Resources Association (ELRA).

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