No Army, No Navy: BERT Semi-Supervised Learning of Arabic Dialects [*]

Chiyu Zhang **Muhammad Abdul-Mageed**
Natural Language Processing Lab
The University of British Columbia
chiyu94@alumni.ubc.ca, muhammad.mageeed@ubc.ca

Abstract

We present our deep leaning system submitted to MADAR shared task 2 focused on twitter user dialect identification. We develop tweet-level identification models based on GRUs and BERT in supervised and semi-supervised settings. We then introduce a simple, yet effective, method of porting tweet-level labels at the level of users. Our system ranks top 1 in the competition, with 71.70% macro F_1 score and 77.40% accuracy.

1 Introduction

Language identification (LID) is an important NLP task that usually acts as an enabling technology in a pipeline involving another downstream task such as machine translation (Salloum et al., 2014) or sentiment analysis (Abdul-Mageed, 2017b,a). Although several works have focused on detecting languages in global settings (see Jauhiainen et al. (2018) for a survey), there has not been extensive research on teasing apart similar languages or language varieties (Zampieri et al., 2018). This is the case for Arabic, the term used to collectively refer to a large number of varieties with a vast population of native speakers (∼ 300 million). For this reason, we focus on detecting fine-grained Arabic dialect as part of our contribution to the MADAR shared task 2, twitter user dialect identification (Bouamor et al., 2019).

Previous works on Arabic (e.g., Zaidan and Callison-Burch (2011, 2014); Elfardy and Diab (2013); Cotterell and Callison-Burch (2014)) have primarily targeted cross-country regional varieties such as Egyptian, Gulf, and Levantine, in addition to Modern Standard Arabic (MSA). These works exploited social data from blogs (Diab et al., 2010; Elfardy and Diab, 2012; Al-Sabbagh and Girju, 2012; Sadat et al., 2014), the general Web (Al-Sabbagh and Girju, 2012), online news sites comments sections (Zaidan and Callison-Burch, 2011), and Twitter (Abdul-Mageed and Diab, 2012; Abdul-Mageed et al., 2014; Mubarak and Darwish, 2014; Qwaider et al., 2018). Other works have used translated data (e.g., Bouamor et al. (2018)), or speech transcripts (e.g., Malmasi and Zampieri (2016). More recently, other works reporting larger-scale datasets at the country-level were undertaken. These include data spanning 10-to-17 different countries (Zaghouani and Charfi, 2018; Abdul-Mageed et al., 2018).

To solve Arabic dialect identification, many researchers developed models based on computational linguistics and machine learning (Elfardy and Diab, 2013; Salloum et al., 2014; Cotterell and Callison-Burch, 2014), and deep learning (Elaraby and Abdul-Mageed, 2018). In this paper, we focus on using state-of-the-arts deep learning architectures to identify Arabic dialects of Twitter users at the country level. We use the MADAR twitter corpus (Bouamor et al., 2019), comprising 21 country-level dialect labels. Namely, we employ unidirectional Gated Recurrent Unit (GRU) (Cho et al., 2014) as our baseline and pre-trained Multilingual Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2018) to identify dialect classes for individual tweets (which we then port at user level). We also apply semi-supervised learning to augment our training data, with a goal to improve model performance. Our system ranks top 1 in the shared task. The rest of the paper is organized as follows: data are described in Section 2. Section 3 introduces our methods, follow by experiments in Section 4. We conclude in Section 5.

[*] The title is word play on the Yiddish linguist Max Weinreich much quoted metaphor (in Yiddish) "A language is a dialect with an army and navy". See: https://en.wikipedia.org/wiki/A_language_is_a_dialect_with_an_army_and_navy.

Proceedings of the Fourth Arabic Natural Language Processing Workshop, pages 279–284
Florence, Italy, August 1, 2019. ©2019 Association for Computational Linguistics

2 Data

Twitter user dialect identification is the second sub-task of 2019 MADAR shared task (Bouamor et al., 2019). This task is set up as fine-grained multi-class classification where corpus released by organizers are labeled with the tagset {*Algeria*, *Bahrain*, *Djibouti*, *Egypt*, *Iraq*, *Jordan*, *Kuwait*, *Lebanon*, *Libya*, *Mauritania*, *Morocco*, *Oman*, *Palestine*, *Qatar*, *Saudi_Arabia*, *Somalia*, *Sudan*, *Syria*, *Tunisia*, *United_Arab_Emirates*, *Yemen*}. The corpus is divided into train, dev and test (with the test set shared without labels). For each tweet, organizers released a user id and tweet id and participants needed to crawl the actual tweets. We were not able to crawl part of the data because of unavailability on the Twitter platform. The distribution of the data in our splits after crawling is as follows: 2,036 (TRAIN-A), 281 (DEV) and 466 (TEST). For our experiments, we also make use of the task 1 corpus (95,000 sentences (Bouamor et al., 2018)). More specifically, we concatenate the task 1 data to the training data of task 2, to create TRAIN-B. Note that both DEV and TEST across our experiments are exclusively the data released in task 2, as described above. TEST labels were only released to participants after the official task evaluation. Table 1 shows statistics of the data.

	# of tweets		
	TRAIN	**DEV**	**TEST**
TRAIN-A	193,086	26,588	43,909
TRAIN-B	288,086	–	–

Table 1: Distribution of classes within the MADAR twitter corpus.

3 Methods

3.1 Pre-processing & Architectures

With tweet ids at hand, we crawl users tweets via the Twitter API. We remove all usernames, URLs, and diacritics in the data. For evaluation, we use accuracy and macro $F_1 - score$. For modeling, we use two main deep learning architectures, Gated Recurrent Unit (GRU) and Bidirectional Encoder Representations from Transformers (BERT). For GRU, we tokenize tweets into word sequences by white-space. For BERT input, we apply WordPiece tokenization. We set the maximal sequence length to 50 words/WordPieces. A GRU (Cho et al., 2014; Chung et al., 2014) is a simplification of long-short term memory networks (LSTM), which in turn are a version of recurrent neural networks.

For BERT (Devlin et al., 2018), it was introduced to dispense with recurrence and convolution. Its model architecture is a multi-layer bidirectional Transformer encoder (Vaswani et al., 2017). It uses masked language models to enable pre-trained deep bidirectional representations, in addition to a binary *next sentence prediction* task. The pre-trained BERT can be easily fine-tuned on large suite of sentence-level and token-level tasks.We also use semi-supervised learning in our modeling, as we explain next.

3.2 Semi-supervise Learning

Supervised deep learning requires a large number of labeled data points. For this reason, we investigate augmenting our training data with automatically-predicted tweets using semi-supervised learning (SSL). More specifically, we use self-training. Self-training is a wrapper method for SSL (Triguero et al., 2015; Pavlinek and Podgorelec, 2017) where a classifier is initially trained on a small set of labeled samples $\boldsymbol{D}^l$. Then, the learned classifier is used to classify the unlabeled sample set $\boldsymbol{D}^u$. Based on the predication output, the most confident samples with their predicted labels are added to the labeled set. The classifier can then be re-trained on the new 'labeled' set. This process can be repeated until all the samples from $\boldsymbol{D}^u$ are added to $\boldsymbol{D}^l$ or a given stopping criteria is reached. We now introduce our experiments.

4 Experiments

We illustrate our four main sets of experiment. We present (i) our baseline model, GRU (Section 4.1), (ii) fine-tuning on BERT-Base, Multilingual Cased model for dialect identification (Section 4.2), (iii) semi-supervised learning with unlabeled data 4.3, (iv) user-level dialect identification (DID) 4.4.

4.1 GRU

We train a baseline GRU network with TRIAN-A. This network has one layer unidirectional GRU with 500 unites and a linear, output layer. The input word tokens are embedded by the trainable word vectors which are initialized with a standard normal distribution, with $\mu = 0$, and $\sigma = 1$, i.e., $W \sim N(0, 1)$. We apply Adam (Kingma and Ba, 2014) with a fixed learning rate of $1e - 3$ for optimization. For regularization, we use dropout (Srivastava et al., 2014) rate of 0.5 on the hidden layer. We set the maximal length of sequence in our GRU model to 50, and choose an arbitrary vocabulary size of 10,000 words. We employ batch training with a batch size of 8 on this model. We run the network for 10 epochs and save the model at the end of each epoch, choosing the model that performs highest on DEV as our best model. We report our best result on dev in Table 2. Our best result is acquired with 3 epochs. As Table 2 shows, the baseline obtains $accuracy = 46.81\%$ and $F_1 = 28.84$.

4.2 BERT

As mentioned earlier, we use the BERT-Base Multilingual Cased model released by the authors [1]. The model is trained on 104 languages (including Arabic) with 12 layer, 768 hidden units each, 12 attention heads, and has 110M parameters in entire model. The model has 119,547 shared WordPieces vocabulary, and was pre-trained on the entire Wikipedia for each language. For fine-tuning, we use a maximum sequence size of 50 tokens and a batch size of 32. We set the learning rate to $2e-5$ and train for 10 epochs. We use the same hyperparameters in all of our BERT models. We fine-tune BERT on TRAIN-A and TRAIN-B sets, and call these BERT-A and BERT-B respectively. As Table 2 shows, both BERT models acquire better performance than the GRU models. On accuracy, BERT-A is 1.69% better than the baseline, and BERT-B is 1.95% better than baseline. BERT-B obtains 34.87 F_1 which is 5.03 better than the baseline and 0.94 better than BERT-A. Our best model of above two sets of experiment is BERT-B which obtains the best accuracy and F_1. Hence, we use BERT-B in our following semi-supervised learning experiments.

Model	Acc.	F1
Baseline (GRU)	46.81	29.84
BERT-A	48.50	33.93
BERT-B	**48.76**	**34.87**

Table 2: Model performance. Baseline is a unidirectional 500-unit, one-layered GRU. Baseline and BERT-A are trained on TRAIN-A. BERT-B is trained on TRAIN-B.

	# of tweets	
	New	**Total**
5%_SEMI	499,102	787,188
10%_SEMI	998,196	1,286,282
25%_SEMI	2,495,491	2,783,577
5%_Class_SEMI	499,087	787,173
10%_Class_SEMI	998,186	1,286,272
25%_Class_SEMI	2,495,486	2,783,572

Table 3: Data splits for our emi-supervised learning experiments. ***New:*** The new dataset confidently predicted with semi-supervised learning that are added to TRAIN-B.

4.3 Semi-supervised Learning

As we mentioned earlier, we apply self-training in order to augment training set. For this purpose, we use an in-house unlabeled, Arabic dataset of 9,981,965 tweets. We refer to this unlabeled dataset as `unlabeled-10M`. We pre-process unlabeled-10M using the same method as the rest of our data. We use the best model from Section 4.2 (i.e. BERT-B, which is trained on TRAIN-B) to predict dialect labels for unlabeled-10M. To obtain the best performance, we investigate various settings to select the most reliable samples before adding such samples to our training data. These settings are based on the per-class value in the softmax/output layer, as follows: **(i) Top-N%:** We select samples which obtain top $n\%$ softmax values and add them with their predicted labels to TRAIN-B. We refer to the new training set as `N_SEMI`. **(ii) Top-N%_Class:** We also extract the samples which obtain top $n\%$ softmax value within each county class and add them to our training data, referring to the new train set as `N_Class_SEMI`. In our experiments, we choose n from the set $\{5\%, 10\%, 25\%\}$. Then, we fine-tune the BERT-Base, Multilingual Cased model on the resulting six new training sets (e.g., `5%_SEMI`, `5%_Class_SEMI`, `10%_SEMI`) with the same hyper-parameters as Section 4.2. We evaluate on DEV. For reference, BERT-*N* denotes the model which is trained on `N_SEMI`, and BERT-*N*Class_SEMI denotes the model which is trained on `N_Class_SEMI`. We present the description of these six train sets in Table 3. As Table 4 shows, most semi-supervised models outperform BERT-B. For accuracy, the best model is

Model	Acc.	F1
Baseline (GRU)	46.810	29.840
BERT-B	48.755	34.868
BERT-5%	48.958	**35.931**
BERT-10%	**49.394**	35.440
BERT-25%	48.751	35.305
BERT-5%_Class_SEMI	48.706	34.774
BERT-10%_Class_SEMI	48.842	33.835
BERT-25%_Class_SEMI	49.097	35.813

Table 4: Semi-supervised learning. All models are evaluated on DEV, with TRAIN-B as training data. Results higher than BERT-B are underlined. Best result is in **bold**.

BERT_10% ($acc = 49.34\%$) with 4 epochs. It is 0.639% higher than BERT-B. For F_1, the best model is BERT_5% ($F_1 = 35.931$) with 3 epochs. We use these two model in the following user-level DID. Since the official metric of the shared task is *macro* F_1 *score*, we also consider BERT-25%_Class_SEMI as a candidate model for user-level DID since it acquires better F_1 than BERT-10% as Table 4 shows.

4.4 User-level DID

Our aforementioned models identify dialect on the tweet-level, rather than directly detect the dialect of a user. Hence, we use tweet-level predicted labels (and associated softmax values) as a proxy for user-level labels. For each predicted label, we use the softmax value as a threshold for including only highest confidently predicted tweets. Since in some cases softmax values can be low, we try all values between 0.00 and 0.99 to take a softmax-based majority class as the user-level predicted label, fine-tuning on our DEV set. Figure 1 provides performance of the `BERT-25%_Class_SEMI` model on DEV using different softmax threshold values. Note that the shared task requires a maximum of three models submitted. For these, we chose the top 3 models in Table 4 (i.e., BERT-5%, BERT-10%, and BERT-25%_Class_SEMI). As a precauion, we also use the BERT-B when we fine-tune on the user-level on DEV. We then use only the 3 models that perform best on DEV as our official task submission. As Table 5 shows, the best three systems on DEV are BERT-B, BERT-5% and BERT-25%_Class_SEMI. For the 34 unavailable users, we assigned the majority class in TRAIN-A (i.e., '*Saudi_Arabia*'). According to 5, our best system on TEST set is BERT-5% with 77.04% accuracy and 71.70 F_1. It rank *top 1* in the shared task.

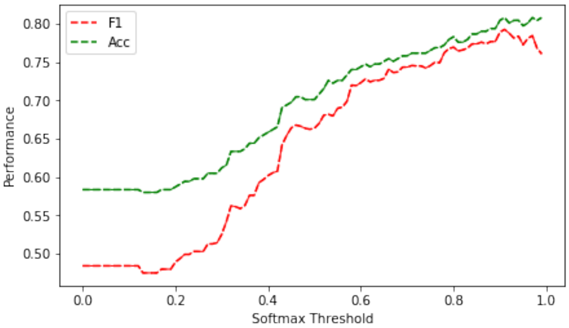

Figure 1: User-level Performance on DEV using different softmax value thresholds.

Model	Thresh	DEV		TEST	
		Acc.	F1	Acc.	F1
BERT-B	0.91	79.36	75.19	76.40	68.47
BERT-5%	0.89	79.36	76.05	**77.40**	**71.70**
BERT-10%	0.92	77.94	74.47	-	-
B-25%CS	0.91	**80.78**	**79.25**	75.80	69.17

Table 5: User-level results. TEST results come from the official leaderboard of the shared task. ***B-25%CS=*** `BERT-25%_Class_SEMI`.

5 Conclusion

In this paper, we described our submission to the MADAR shared task 2, focused on user-level Arabic dialect identification. We show how we acquire effective models using various supervised and semi-supervised methods, porting tweet-level labels to the user level. Our semi-supervised model with BERT achieves best results in the official task evaluation. In the future, we will investigate more extensive semi-supervised methods to improve performance.

6 Acknowledgement

We acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC), the Social Sciences Research Council of Canada (SSHRC), and Compute Canada (`www.computecanada.ca`).

References

Muhammad Abdul-Mageed. 2017a. Modeling arabic subjectivity and sentiment in lexical space. *Information Processing & Management*.

Muhammad Abdul-Mageed. 2017b. Not all segments are created equal: Syntactically motivated sentiment analysis in lexical space. In *Proceedings of the third Arabic natural language processing workshop*, pages 147–156.

Muhammad Abdul-Mageed, Hassan Alhuzali, and Mohamed Elaraby. 2018. You tweet what you speak: A city-level dataset of arabic dialects. In *LREC*, pages 3653–3659.

Muhammad Abdul-Mageed, Mona Diab, and Sandra Kübler. 2014. Samar: Subjectivity and sentiment analysis for arabic social media. *Computer Speech & Language*, 28(1):20–37.

Muhammad Abdul-Mageed and Mona T Diab. 2012. Awatif: A multi-genre corpus for modern standard arabic subjectivity and sentiment analysis. In *LREC*, pages 3907–3914.

Rania Al-Sabbagh and Roxana Girju. 2012. Yadac: Yet another dialectal arabic corpus. In *LREC*, pages 2882–2889.

Houda Bouamor, Nizar Habash, Mohammad Salameh, Wajdi Zaghouani, Owen Rambow, Dana Abdulrahim, Ossama Obeid, Salam Khalifa, Fadhl Eryani, Alexander Erdmann, et al. 2018. The madar arabic dialect corpus and lexicon. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*.

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.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.

Ryan Cotterell and Chris Callison-Burch. 2014. A multi-dialect, multi-genre corpus of informal written arabic. In *LREC*, pages 241–245.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Mona Diab, Nizar Habash, Owen Rambow, Mohamed Altantawy, and Yassine Benajiba. 2010. Colaba: Arabic dialect annotation and processing. In *Lrec workshop on semitic language processing*, pages 66–74.

Mohamed Elaraby and Muhammad Abdul-Mageed. 2018. Deep models for Arabic dialect identification on benchmarked data. In *Proceedings of the Fifth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2018)*, pages 263–274, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Heba Elfardy and Mona T Diab. 2012. Simplified guidelines for the creation of large scale dialectal arabic annotations. In *LREC*, pages 371–378.

Heba Elfardy and Mona T Diab. 2013. Sentence level dialect identification in arabic. In *ACL (2)*, pages 456–461.

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

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Shervin Malmasi and Marcos Zampieri. 2016. Arabic dialect identification in speech transcripts. *VarDial 3*, page 106.

Hamdy Mubarak and Kareem Darwish. 2014. Using twitter to collect a multi-dialectal corpus of arabic. In *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*, pages 1–7.

Miha Pavlinek and Vili Podgorelec. 2017. Text classification method based on self-training and lda topic models. *Expert Systems with Applications*, 80:83–93.

Chatrine Qwaider, Motaz Saad, Stergios Chatzikyriakidis, and Simon Dobnik. 2018. Shami: A corpus of levantine arabic dialects. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*.

Fatiha Sadat, Farnazeh Kazemi, and Atefeh Farzindar. 2014. Automatic identification of arabic language varieties and dialects in social media. *Proceedings of SocialNLP*, page 22.

Wael Salloum, Heba Elfardy, Linda Alamir-Salloum, Nizar Habash, and Mona Diab. 2014. Sentence level dialect identification for machine translation system selection. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 772–778.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958.

Isaac Triguero, Salvador García, and Francisco Herrera. 2015. Self-labeled techniques for semi-supervised learning: taxonomy, software and empirical study. *Knowledge and Information Systems*, 42(2):245–284.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Wajdi Zaghouani and Anis Charfi. 2018. Arap-tweet: A large multi-dialect twitter corpus for gender, age and language variety identification. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*.

Omar F Zaidan and Chris Callison-Burch. 2011. The arabic online commentary dataset: an annotated dataset of informal arabic with high dialectal content. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 37–41. Association for Computational Linguistics.

Omar F Zaidan and Chris Callison-Burch. 2014. Arabic dialect identification. *Computational Linguistics*, 40(1):171–202.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Ahmed Ali, Suwon Shon, James Glass, Yves Scherrer, Tanja Samardžić, Nikola Ljubešić, Jörg Tiedemann, et al. 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*. Association for Computational Linguistics.