Semi-supervised Fine-grained Approach for Arabic Dialect Detection

Nitin Nikamanth Appiah Balaji
SSN College of Engineering
Kalavakkam, India
nitinnikamanthab17099@cse.ssn.edu.in

B. Bharathi
SSN College of Engineering
Kalavakkam, India
bharathib@ssn.edu.in

Abstract

Arabic is a language with numerous dialects. It becomes extremely important to device a technique to distinguish each dialect efficiently. This paper focuses on the fine-grained country-level and province-level classification of Arabic dialects. The experiments described in this paper are submissions of team TRY_NLP for NADI SHARED TASK on Nuanced Arabic Dialect Identification. Various text feature extraction techniques such as TF-IDF, AraVec, multilingual BERT and FastText embedding models are studied. We thereby, propose an approach of text embedding based model with macro average F1 score of 22.32% for subtask 1 and 4.83% for subtask 2, with the help of semi-supervised learning approach.

1 Introduction

As Arabic has a diverse collection of dialects and as Dialect Arabic varies phonologically, lexically, and morphologically from Modern Standard Arabic (Bouamor et al., 2018), fine-grained dialect identification becomes an important task. The classification helps for studying various indigenous properties among different dialect speakers. Previously, work on fine-grained city-level classification has been done (Salameh et al., 2018), for MADAR tasks (Bouamor et al., 2018). NADI 2020 task consist of 21 class country-level dialect labels and 100 class province-level fine-grained dialect labels.

With recent development in text classification, from emerging techniques like embedding models, not only for English, but trained on multiple languages have opened opportunities for Arabic text classification. As models like Word2Vec, FastText and BERT perform excellent in text classification of English twitter data, these can be extended into the usage of Arabic languages. Features extracted from these models are then further fine tuned using neural network layers, to fit the required classification.

In this paper we propose a model giving extremely good scores, using a combination of labelled and unlabelled data, by semi-supervised learning technique. Unlabelled 10 million twitter data is collected with their ID provided for the challenge and the top confidence predictions are added to extend the main labelled data set to increase the training set. Repeated training with the added data in each step, increased the F1 score significantly.

The remainder of the paper is organized as follows. Section 2 discusses the description of the data set and preprocessing methods used. Section 3 outlines the features used for the experiments for both subtask 1 and subtask 2. Results are discussed in Section 4. Section 5 concludes the paper.

2 Data Description and Processing

The data provided in the NADI 2020 task consist of 21,000 labelled and 10 million unlabelled data set for training, 4,957 tweet dev set and 5,000 tweet test set. The data consist of 21 different country-level labels and 100 different province-level labels with unequal distribution of tweets in various classes. The details about the shared task is described in (Abdul-Mageed et al., 2020)

Proceedings of the Fifth Arabic Natural Language Processing Workshop, pages 257–261
Barcelona, Spain (Online), December 12, 2020

Additionally 10 million tweet IDs were given without labels. These unlabelled tweets were extracted by using the Twitter API. Out of the 10 million tweets approximately 92% of tweets are available and is considered for semi-supervised learning.

As the texts are direct tweet messages, it contains links, hashtags and numbers, which is against the context of our study. So tweets are cleaned by removing the unwanted links, English and special characters, considering just the Arabic text to maintain generality.

Data type	# tweets
train-labelled	21,000
train-unlabelled	10 million
dev	4,957
test	5,000

Table 1: Dataset distribution.

3 Experimental Setup

For both subtask1 and subtask2, TF-IDF, AraVec, BERT and FastText features are extracted and these feature vectors are fine-tuned to classify dialect classes by appending an extra layer. By this method of transfer learning we have established an effective and faster way of training the model.

Due to the large number of classes and great class imbalance, macro average F1 score is suitable and is used in the NADI task. Also accuracy, precision and recall are observed.

3.1 TF-IDF Vectorization

TF-IDF abbreviated as term frequency-inverse domain frequency, which is a count based text-extraction model with inverse domain frequency term normalising the representation. This representation reduces the impact of the large repetition of one unimportant word in the sentence. This approach is better than just count vectorization of the sentences and works well for small train set (Mishra and Mujadia, 2019). Previously, TF-IDF model fusion models, presented to MADAR SHARED task (Bouamor et al., 2019) have shown good performance (Abu Kwaik and Saad, 2019; Mishra and Mujadia, 2019).

3.2 AraVec

As Arabic is a language with complex morphology and having different ways to express the same meaning, it becomes important to capture the syntactic and semantic relations among the words in the sentence.

Word2Vec models are shallow, two-layer neural networks, trained to reconstruct linguistic contexts of words (Mikolov et al., 2013). So it becomes a basic approach for understanding the semantics as well as run fast. Similar to word2vec trained on English corpus, AraVec (Soliman et al., 2017) is trained on Arabic corpus and becomes quicker and requires lesser data to fine tune the model. AraVec model, a n-gram model trained on Arabic twitter data and Wikipedia Arabic articles is considered suitable for this task.

3.3 Multilingual BERT

BERT is a Bidirectional Encoder Representations from Transformers architecture (Devlin et al., 2018). BERT is the state-of-the-art model available for twitter text classification, with transformer based unsupervised learning model. As the BERT model is pre-trained on a large corpus of data, to learn the context similarities of the sentences, it is easier to fine-tune and get good results. The multilingual cased BERT model which is trained on 104 different languages, including Arabic is used to generate the word vectors.

3.4 Arabic FastText

We use the FastText model trained on common crawl and Wikipedia data of Arabic sentences (Grave et al., 2018). It has been proven to give good results in (Samih et al., 2019). This produces a fixed

embedding of length 300. The model is trained using CBOW (Continuous Bag of Words) with position-weights, with character n-grams of length 5, a window of size 5 and 10 negatives. The FastText model is fragment based, and gives good representation on even unseen words. The twitter dataset for the classification, mostly contains unknown words and so gives better representation with the implementation of FastText from magnitude (Patel et al., 2018).

Out of all the experiments conducted the Arabic FastText fine-tuned model produced great results on the development set with up-to 0.18% macro avg F1 score, out performing the multilingual BERT model. This is majorly because FastText Arabic specific pre-trained model is readily available, whereas BERT only has generalized multilingual variant and the limitation in the train data set size have made FastText shine.

3.5 Semi-supervised Learning

The limited labelled data constraints the model, to learn much and to overfit on the training corpus. To generate a generalized model, which could perform well on the test set, it is must to increase the data set to see any further improvements in the performance. So, by semi-supervised learning strategy (Zhang and Abdul-Mageed, 2019), we tried to get more labelled samples by pseudo labelling the unlabelled data with a previously trained model. The steps for semi-supervised learning used in the proposed approach is depicted in Figure 1.

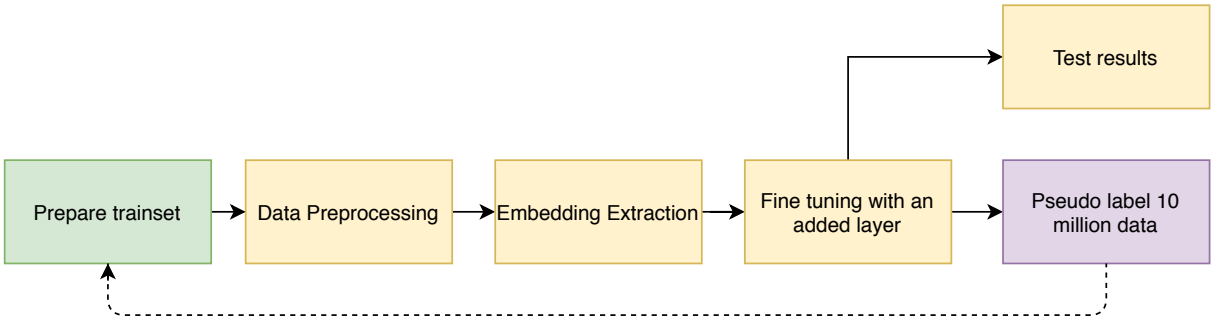

Figure 1: Semi-supervised learning pipeline

The FastText model, which has the best performance is considered and the unlabelled 10 million tweets are predicted along with their confidence scores. Then the top confidence score results are selected and added to the main training data. Then the training process is repeated again, and the addition of pseudo labelled data is repeated multiple times. The threshold of confidence score to be added to training set is experimented with different values as explained in (Zhang and Abdul-Mageed, 2019). This process of semi-supervised learning gave a significant improvement and reached 0.2232 macro avg F1 score.

4 Results

The pre-trained models for Arabic corpus is hard to find, making models like BERT perform slightly lesser than expected. Even though normalized count based model is basic, TF-IDF method showed relatively better results than the AraVec model, as TF-IDF transform is fitted on the training corpus, and not using different corpus.

Out of all models trained on only the labelled training data, the FastText model performs the best as its implementation with the magnitude library (Patel et al., 2018), as it assigns the same id for repeating words that doesn't exist in the pre-trained corpus. By this model we reach an excellent F1 score of 0.1890 for country-level classification and 0.0437 for province-level classification, on the dev set.

The results of each feature extraction technique are observed and tabulated in Table 2 and 3.

Then considering the unlabelled tweets, extracted using the tweeter API, the model is trained in a semi-supervised fashion. First the model already trained on the labelled data, is used to produce the labels for the 10 million tweets and out of which the top confidence results are selected and concatenated with the labelled data. To not induce excess error into the model, various thresholds of 1, 5, 10%

Model	macro avg F1	accuracy	precision	recall
TF-IDF	0.1522	0.3099	0.1697	0.1487
AraVec	0.1159	0.3123	0.1321	0.1237
multilingual BERT	0.1175	0.2417	0.1294	0.1146
FastText ar	0.1890	**0.3752**	0.1939	0.1922
Semi-supervised FastText	**0.2232**	0.3674	**0.2285**	**0.2404**

Table 2: Subtask1 - Country-level classification report on dev set

is experimented. This addition of data to the train data, helped to increase the models performance significantly, approximately 20% increase compared to the supervised learning score.

Model	macro avg F1	accuracy	precision	recall
TF-IDF	0.0442	0.0444	0.0476	0.0447
AraVec	0.0213	0.0268	0.0253	0.0271
multilingual BERT	0.0275	0.0313	0.0278	0.0331
FastText ar	0.0437	0.0516	0.0390	0.0560
Semi-supervised FastText	**0.0483**	**0.0551**	**0.0451**	**0.0608**

Table 3: Subtask2 - Province-level classification report on dev set

The FastText model was considered for submission in the NADI 2020 tasks and the results are as tabulated in Table 4 and 5.

Model	macro avg F1	accuracy	precision	recall
Semi-supervised FastText	0.2004	0.3366	0.2070	0.2107

Table 4: Subtask1 - Country-level classification report on test set

Model	macro avg F1	accuracy	precision	recall
Semi-supervised FastText	0.0403	0.0486	0.0374	0.0468

Table 5: Subtask2 - Province-level classification report on test set

5 Conclusion

coling.bst, line 1244 Fine-grained Arabic dialect classification is explored, with NADI challenge data set. The tasks of country-level and province-level dialect identification for 21 classes and 100 classes are studied. In this paper, we explore various models such as TF-IDF, AraVec, multi-BERT and ar-FastText. We could see that the FastText model performed relatively better than the other models for such small training set. And also we could see the improvement of performance, by semi-supervised learning technique using the unlabelled data. By this strategy we achieve 0.2232 and 0.0483 macro avg F1 for subtask 1 and subtask 2 respectively on dev data set. The model showed 0.2004 and 0.0403 macro avg F1 for subtask 1 and subtask 2 on test data set.

References

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 (WANLP 2020)*, Barcelona, Spain.

Kathrein Abu Kwaik and Motaz Saad. 2019. ArbDialectID at MADAR shared task 1: Language modelling and ensemble learning for fine grained Arabic dialect identification. In *Proceedings of the Fourth Arabic Natural Language Processing Workshop*, pages 254–258, Florence, Italy, August. Association for Computational Linguistics.

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, May. European Language Resources Association (ELRA).

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*, pages 199–207, Florence, Italy. Association for Computational Linguistics.

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*.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space.

Pruthwik Mishra and Vandan Mujadia. 2019. Arabic dialect identification for travel and twitter text. In *Proceedings of the Fourth Arabic Natural Language Processing Workshop*, pages 234–238.

Ajay Patel, Alexander Sands, Chris Callison-Burch, and Marianna Apidianaki. 2018. Magnitude: A fast, efficient universal vector embedding utility package. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 120–126.

Mohammad Salameh, Houda Bouamor, and Nizar Habash. 2018. Fine-grained arabic dialect identification. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1332–1344.

Younes Samih, Hamdy Mubarak, Ahmed Abdelali, Mohammed Attia, Mohamed Eldesouki, and Kareem Darwish. 2019. QC-GO submission for MADAR shared task: Arabic fine-grained dialect identification. In *Proceedings of the Fourth Arabic Natural Language Processing Workshop*, pages 290–294, Florence, Italy, August. Association for Computational Linguistics.

Abu Bakr Soliman, Kareem Eissa, and Samhaa R El-Beltagy. 2017. Aravec: A set of arabic word embedding models for use in arabic nlp. *Procedia Computer Science*, 117:256–265.

Chiyu Zhang and Muhammad Abdul-Mageed. 2019. No army, no navy: Bert semi-supervised learning of arabic dialects. In *Proceedings of the Fourth Arabic Natural Language Processing Workshop*, pages 279–284.