QC-GO Submission for MADAR Shared Task: Arabic Fine-Grained Dialect Identification

Younes Samih[1] **Hamdy Mubarak**[1] **Ahmed Abdelali**[1] **Mohammed Attia**[2]
Mohamed Eldesouki[1] **Kareem Darwish**[1]
[1]{ysamih,hmubarak,aabdelali,mohamohamed,kdarwish}@hbku.edu.qa
[2]{attia}@google.com
[1]Qatar Computing Research Institute, HBKU Research Complex, Doha, Qatar
[2]Google LLC, New York City, USA

Abstract

This paper describes the QC-GO team submission to the MADAR Shared Task Subtask 1 (travel domain dialect identification) and Subtask 2 (Twitter user location identification). In our participation in both subtasks, we explored a number of approaches and system combinations to obtain the best performance for both tasks. These include deep neural nets and heuristics. Since individual approaches suffer from various shortcomings, the combination of different approaches was able to fill some of these gaps. Our system achieves F1-Scores of 66.1% and 67.0% on the development sets for Subtasks 1 and 2 respectively.

1 Introduction

Arabic, similar to other languages have a number of dialectal varieties. With the emergence of social media, many of these varieties of Arabic started having wide representation in the written form. Twitter, Facebook, and YouTube are among the leading sources of such data (Zaidan and Callison-Burch, 2011; Mubarak and Darwish, 2014; Samih et al., 2017; Samih and Maier, 2016). The wide spread of dialectal use has increased the richness and diversity of the language, requiring greater complexity in dealing with it. Non-standard orthography, increased borrowing and coinage of new terms, and code switching are just a few among a long list of new challenges researchers have to deal with.

Studying language varieties in particular is associated with important applications such as Dialect Identification (DID), Machine Translation (MT), and other text mining tasks. Performing DID can be achieved using a variety of features, such as character n-grams (Darwish, 2014; Zaidan and Callison-Burch, 2014; Malmasi et al., 2015), and a myriad of techniques, such as string kernels (Ionescu and Popescu, 2016) and DNN (Elaraby and Abdul-Mageed, 2018).

In this paper, two resources created under the Multi-Arabic Dialect Applications and Resources (MADAR) project were used as the main resources for the task of Fine-Grained Dialect Identification (Salameh et al., 2018; Bouamor et al., 2018). The MADAR Shared Task (Bouamor et al., 2019) aims to identify dialects at the city/country level for two datasets . Subtask 1 uses a travel domain collection of 110k sentences that contain both Modern Standard Arabic (MSA) sentences and their translations into 25 dialects representing major cities in the Arab world. Subtask 2 aims to classify tweeps (Twitter users) per their location using 100 of their tweets or less. In this paper, we describe the approaches that we utilized for dialect identification, which include the use of deep neural networks and heuristics.

2 System descriptions

For both SubTask 1 and SubTask 2, we employed a hybrid system that incorporates different classifiers and components such DNNs and heuristics to perform sentence level dialectal Arabic identification. The classification strategy is built as a cascaded voting system that tags each sequence based on the decisions from two other underlying classifiers.

DNNs: This model uses both Bidirectional Long Short Term Memory (Bi-LSTM) and Convolutional Neural Network (CNN) architectures to jointly learn both word-level and character-level representations, and project them to a softmax output layer for dialectal Arabic identification. At the word level, we use pre-trained word embeddings for Dialectal Arabic to initialize our look-up table. Words with no pre-trained embeddings are randomly initialized with uniformly sampled em-

Proceedings of the Fourth Arabic Natural Language Processing Workshop, pages 290–294
Florence, Italy, August 1, 2019.

beddings. To use these embeddings in our model, we simply replace one hot encoding word representations with corresponding 300-dimensional vectors. Note that in this settings, we trained our embeddings on the provided training set. We used two approaches for preparing the embeddings, namely gensim word2vec (Řehůřek and Sojka, 2010) and fastText (Joulin et al., 2016), which will be referred later as DNN-wv and DNN-ft respectively.

At the character level, to capture word morphology and reduce out-of-vocabulary, we used convolutions to learn local n-gram features. This approach has also been especially useful for handling languages with rich morphology and large character sets (Kim et al., 2016). The first layer projects each character into its corresponding character embeddings, as with a look-up table, and stacks them to form a matrix C^k. Convolution operations with the same padding are applied between C^k and multiple filter matrices. A max-over-time pooling operation is then executed to infer a fixed-dimensional representation of the words. This representation is then concatenated with word embeddings and fed to a highway network (Srivastava et al., 2015). The highway network's output is applied to a multi-layer Bi-LSTM. At the output layer, a softmax is applied over the hidden representation of the two LSTMs to obtain the probability distribution over all labels. Training is performed using stochastic gradient descent with momentum, optimizing the cross-entropy objective function.

FastText: FastText is a deep learning based library for efficient learning of word representations and text classification. It represents words as the sum of their character *n*-grams vectors. It has been shown to be effective for text classification for different tasks (Joulin et al., 2017).

Arabic is a rich Semitic language with complex morphology where a large number of prefixes and suffixes can be attached to words. Additionally, in Arabic dialects, words can be written in many different ways, because there is no conventional orthography. The aforementioned reasons suggest that using words alone as features for classification is less optimal. We opted to compliment that with variable length character *n*-grams to capture sub-word information and local contextual information. For Subtask 1 and Subtask 2, we tuned different settings on the development set, and the best results were obtained when using character *n*-grams varying from 3 to 6 characters, dimensions of vectors of 100, a learning rate of 0.1, and 50 training epochs.

System	Dev. F-1 Score	Test F-1 Score
DNN-ft	59.78%	57.25%
DNN-wv	58.11%	58.72%
FastText	63.09%	60.42%
QC-GO1	64.53%	58.72%
QC-GO2	63.49%	58.45%
QC-GO3	63.29%	57.26%

Table 1: SubTask 1 Results for the submissions for Development and Test sets.

Heuristics: For sub-task 2, we constructed a list of all Arabic speaking countries (e.g. مصر (Egypt)) along with major cities in these countries (e.g. القاهرة (Cairo)). Given our list, we counted the number of times a tweep mentions the names of countries or any of the cities therein in his/her tweets. Then, we labeled a tweep with the county that is mentioned most in the tweets.

Ensemble model: For both sub-tasks, our final system combines the output from the different systems using a simple majority vote to perform dialectal Arabic predictions. The ensemble model can either assign varying weights to different systems depending on their overall performance on the dev set or it takes the average by setting equal weights for all systems.

3 Results

In this section we present the results of our system output for Subtask 1 and Subtask 2 on both the validation and the test sets.

3.1 SubTask 1

The results, shown in Table 1, contain a combination of the systems described above with variable weighting. Since the results of individual systems varied greatly, their combination proved to be more effective. Combining DNN-ft with DNN-wv with a weight of 0.66, 0.33 respectively improved the predictions from 57.25% to 58.45%. Adding fastText to the mix achieved 60.85%: a boost of more than 6.2%.

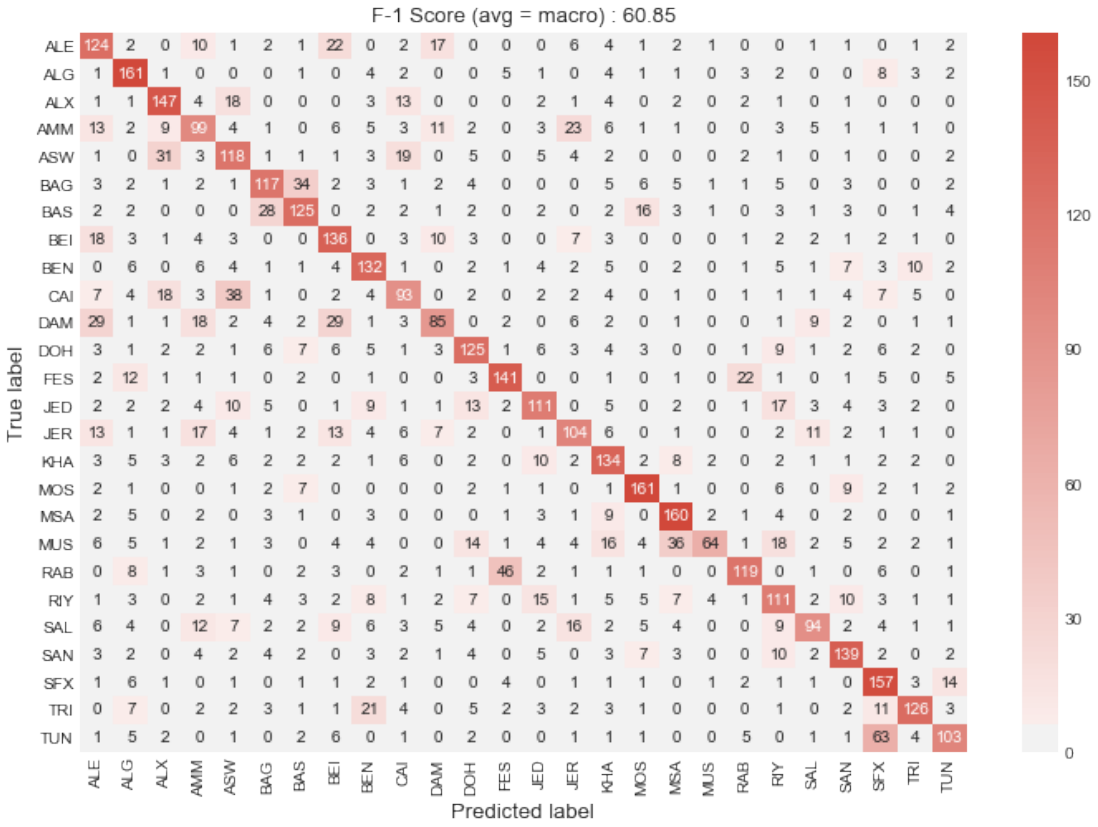

Figure 1: Confusion matrix for results of SubTask 1 system combination

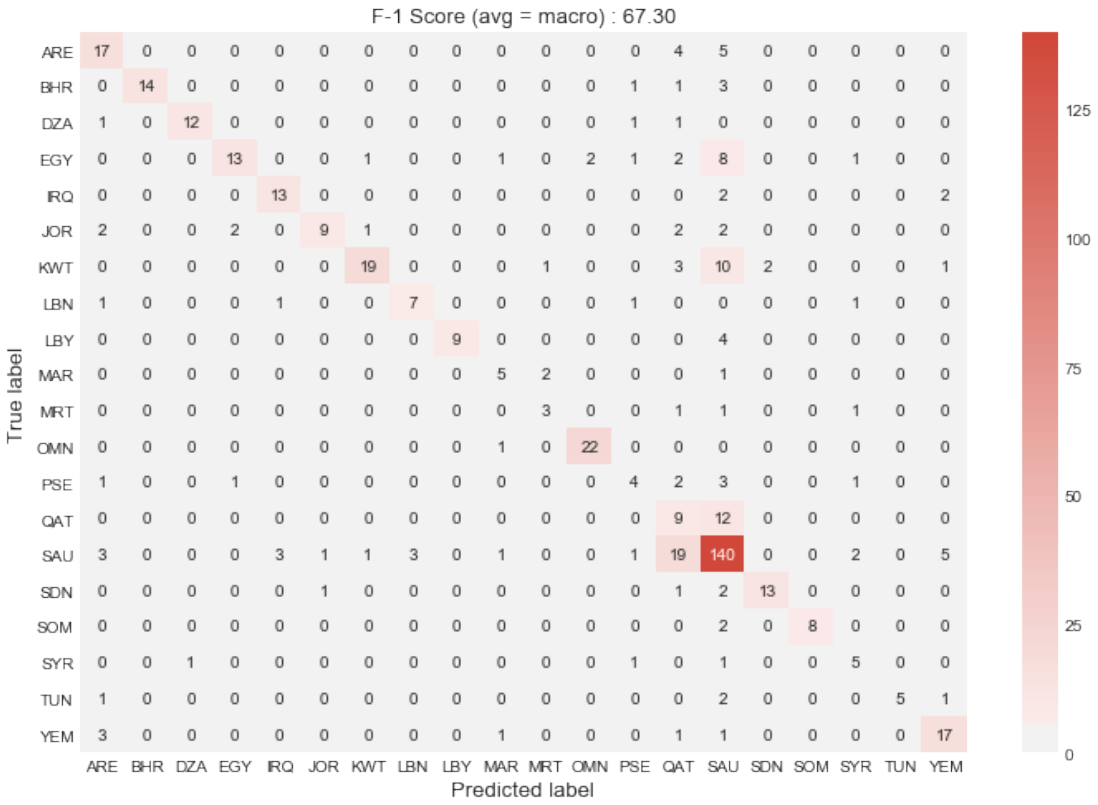

Figure 2: Confusion matrix for results of SubTask 2 FastText

Submission	Dev. F-1 Score	Test F-1 Score
DNN-ft	44.54%	54.50%
DNN-wv	47.04%	43.23%
FastText	57.41%	57.23%
Hueristics	65.22%	67.30%
QC-GO1	63.77%	66.68%
QC-GO3	63.77%	66.34%
QC-GO2	66.60%	63.92%

Table 2: SubTask 2 results for the submissions for Development and Test sets.

3.2 SubTask 2

As for SubTask 2, the combination of DNN-ft with DNN-wv was not as effective as either alone. A decrease of 1.2% was observed. On the other hand using fastText by itself achieved an F-1 score of 57.23%, which is higher than both DNN-ft and DNN-wv. Using the heuristics approach yielded the best performance with 64.09%. Adding a back-off to use a majority vote per user, when a tweep did not mention any country or any city therein, to get the most frequent predicted country improved result to 67.30%. This system ranked third among all submitted systems for SubTask 2.

4 Discussion and conclusions

Our analysis of the system output on the validation set for Subtask 1 shows that the highest accuracy obtained at the dialect level was for MSA, SFX, ALX, and MOS, (Figure 1) while the lowest accuracy was for MUS, DAM, and AMM. Local dialects within the same country caused the vast majority of confusion. For example, the most confusion for SFX came from TUN, for BAS came from BAG, and for JED came from RIY. We also observed a heightened confusion between cities from neighboring countries, such as ALG and FES, BEI and ALE, and JER and AMM. This observation emphasizes the perception that there is a level of homogeneity between dialects with physical proximity whether at the national and regional levels. As for the Subtask 2, the challenging ambiguity between gulf dialects is still a major issue that caused the accuracy drop; See (Figure 2). Expanding the data for these subdialects would enhance their respective accuracy.

References

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 Language Resources and Evaluation Conference (LREC)*, 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.

Kareem Darwish. 2014. Arabizi detection and conversion to arabic. *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*, pages 217–224.

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)*.

Radu Tudor Ionescu and Marius Popescu. 2016. Unibuckernel: An approach for arabic dialect identification based on multiple string kernels. In *Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3)*, pages 135–144.

Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hérve Jégou, and Tomas Mikolov. 2016. Fasttext.zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431. Association for Computational Linguistics.

Yoon Kim, Yacine Jernite, David Sontag, and Alexander M Rush. 2016. Character-aware neural language models. In *Thirtieth AAAI Conference on Artificial Intelligence*.

Shervin Malmasi, Eshrag Refaee, and Mark Dras. 2015. Arabic dialect identification using a parallel multidialectal corpus. In *Conference of the Pacific Association for Computational Linguistics*, pages 35–53. Springer.

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.

Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In

Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks, pages 45–50, Valletta, Malta. ELRA.

Mohammad Salameh, Houda Bouamor, and Nizar Habash. 2018. Fine-grained Arabic dialect identification. In *Proceedings of the International Conference on Computational Linguistics (COLING)*, pages 1332–1344, Santa Fe, New Mexico, USA.

Younes Samih, Mohamed Eldesouki, Mohammed Attia, Kareem Darwish, Ahmed Abdelali, Hamdy Mubarak, and Laura Kallmeyer. 2017. Learning from relatives: Unified dialectal Arabic segmentation. In *(CoNLL 2017)*, pages 432–441, Vancouver, Canada. Association for Computational Linguistics.

Younes Samih and Wolfgang Maier. 2016. An Arabic-Moroccan Darija code-switched corpus. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 4170–4175, Portorož, Slovenia. European Language Resources Association (ELRA).

Rupesh K. Srivastava, Klaus Greff, and Jürgen Schmidhuber. 2015. Training very deep networks. In *Advances in neural information processing systems*, pages 2377–2385.

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 ACL-HLT: short papers-Volume 2*, pages 37–41.

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