ZCU-NLP at MADAR 2019: Recognizing Arabic Dialects

Pavel Přibáň[1,2], and **Stephen Taylor**[1]

[1] Department of Computer Science and Engineering, Faculty of Applied Sciences,
[2] NTIS – New Technologies for the Information Society, Faculty of Applied Sciences,
University of West Bohemia, Univerzitní 8, 306 14 Plzeň, Czech Republic

E-mail: `pribanp@kiv.zcu.cz`, `stepheneugenetaylor@gmail.com` Web: `nlp.kiv.zcu.cz`

UNIVERSITY OF WEST BOHEMIA

NTIS

<DCSE> DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

MADAR SubTask 1

The goal of the Subtask–1 is to detect one of 25 specific Arabic city dialects or MSA[1] in a given sentence.

هذا الطريق من فضلك . خذ هذا المصعد .

⇒ MSA

[1] Modern Standard Arabic

MADAR SubTask 2

The goal of the Subtask–2 is to predict the country (out of 21 Arab countries) of origin of a Twitter user by using tweets posted by the user.

مد.. يدك وامنحهم الدفء الذى ينتظرونه تحت الصفر

⇒ Qatar

Subtask–1 Overview

Our Approach?

- **Tortuous Classifier**
  - Language model features + Classic machine learning method (SVM, Naive Bayes)
- **Neural Network Classifier**
  - Language model features, Character Embeddings + BiLSTM

Subtask–2 Overview

- Pre-trained 21 language models built on the development tweets
- Tweet assigned to the country with the largest language model score
- The user country is decided based on the counts of tweet assignments

Tortuous Classifier

Inputs:

- Pre-trained 26 dialect word/character language models, word unigrams and bigrams, character 3-gram, 4-gram, and 5-gram

Classifier[1]:

- Several Multinomial Naive Bayes and SVM classifiers
- Combined into voting classifiers
  - Experiments with soft/hard voting
- Similar features used by the baseline character 5-gram language models

[1] We call it *tortuous* because it twists around to apply multiple classifiers to the same features

Neural Network Classifier

Inputs:

- Pre-trained 26 dialect character language models
- Sequence of first 200 character n-grams of a given text

  ⇒ Character Embeddings

Architecture:

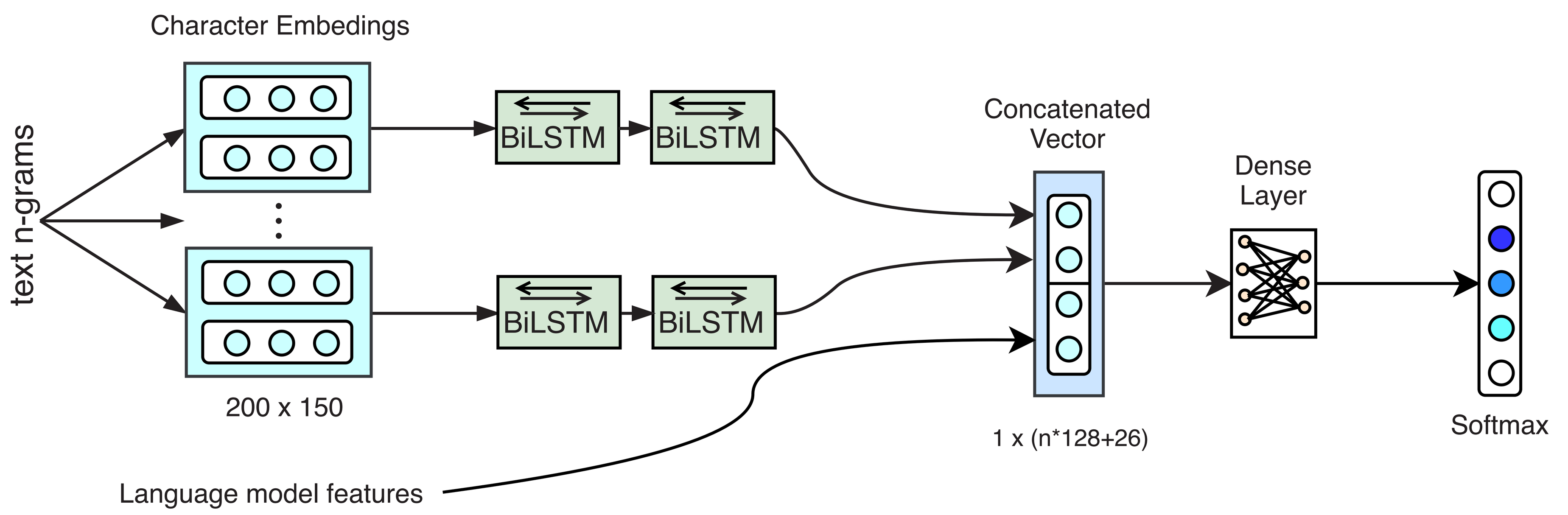
- Embedding layer is followed by two BiLSTMs with 64 units
- The Output vector of the BiLSTMs is concatenated with language model features

  ⇒ Character Embeddings
- The concatenated vector is passed to MLP layer (with 400) units which is followed by a softmax layer

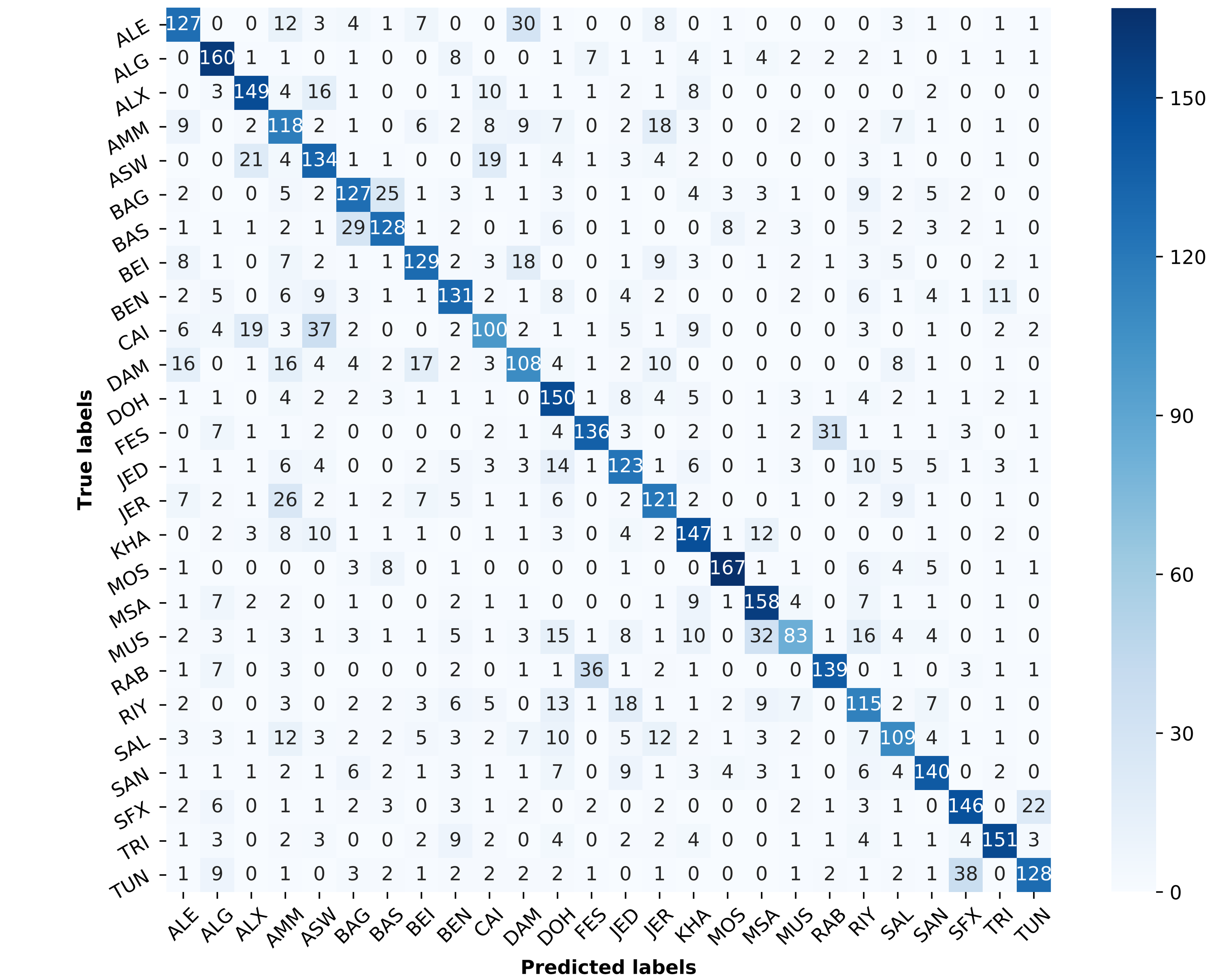
Model Training & Hyper-Parameters:

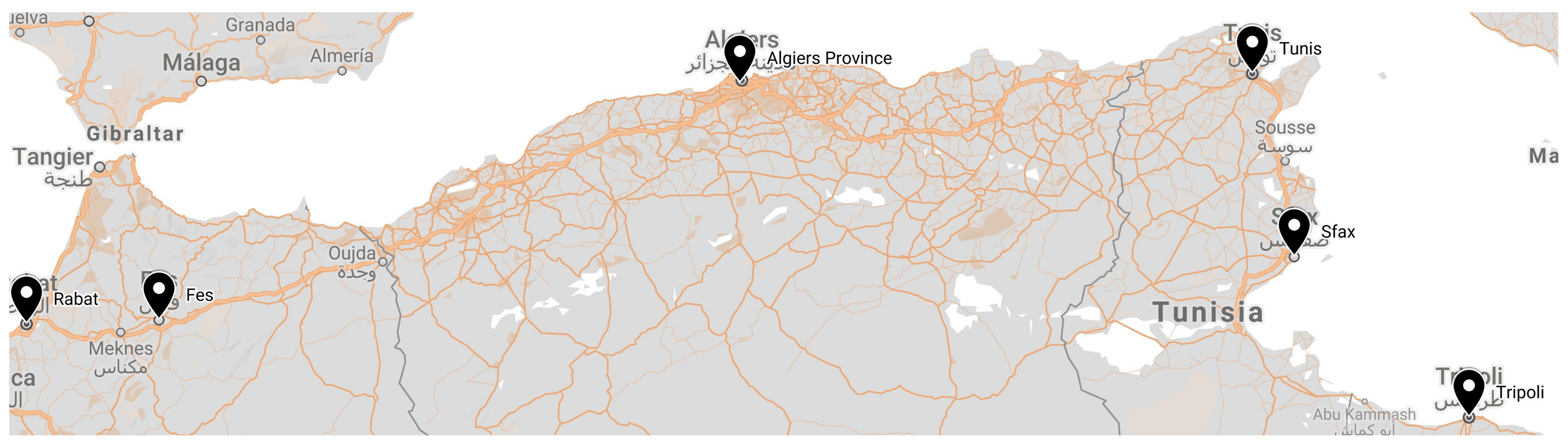
- Training for 800 epochs
- Adam optimizer with learning rate 0.01, no dropout

Neural Network Classifier Architecture

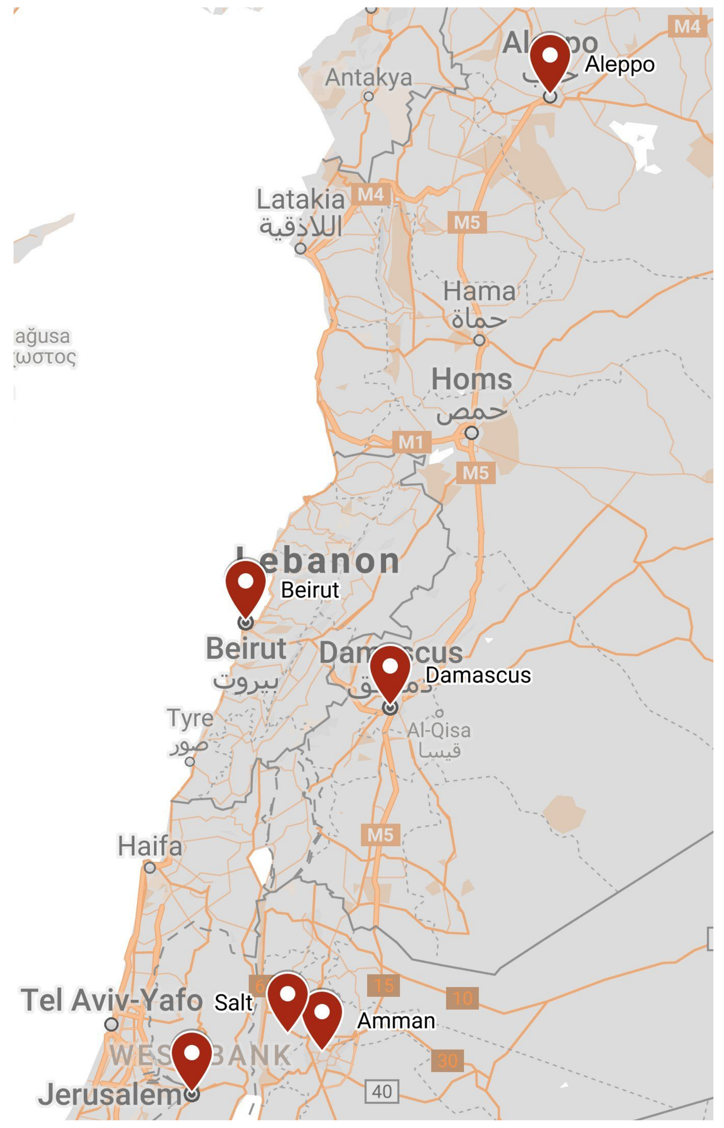


Tortuous Confusion Matrix & Cities Map





Cities Map



Source Code

`https://qrco.de/UWB-MADAR`

SubTask 1 Results

- **Tortuous Classifier**
  - Best 0.658 macro F_1-score on the test data
- **Neural Network Classifier**
  - 0.648 macro F_1-score on the test data [1]
  - 0.555 macro F_1-score[2], only with n-gram input (unigrams, bigrams and trigrams)
- Classic machine learning approach outperforms neural network
- Best results achieved only with a language model features
- Many geographically related errors

[1] Only with a language model features
[2] On the development data

SubTask 2 Results II

- 47.51 macro F_1-score on the test data
- This is below the baseline (50.31) which also used character 5-gram language model scores
- Apparently the baseline combined tweet results differently; perhaps it combined all tweets for a user before scoring.

Conclusion

This paper presents an automatic approach for Arabic dialect detection. Our proposed systems for the Subtask–1 use language model features. Our experiments showed that simpler machine learning algorithms outperform RNN using language model features.

Subtask–2 turned out to be more challenging because Tweets, which are real-world wild data, are more difficult to process than systematically prepared texts.