The SMarT Classifier for Arabic Fine-Grained Dialect Identification
Karima Meftouh, Karima Abidi, Salima Harrat, Kamel Smaïli

To cite this version:

Karima Meftouh, Karima Abidi, Salima Harrat, Kamel Smaïli. The SMarT Classifier for Arabic Fine-Grained Dialect Identification. The Fourth Arabic Natural Language Processing Workshop co-located with ACL, Aug 2019, Florence, Italy. hal-02166384

HAL Id: hal-02166384
https://hal.archives-ouvertes.fr/hal-02166384
Submitted on 26 Jun 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Abstract

This paper describes the approach adopted by the SMarT research group to build a dialect identification system in the framework of the Madar shared task on Arabic fine-grained dialect identification. We experimented several approaches, but we finally decided to use a Multinomial Naïve Bayes classifier based on word and character ngrams in addition to the language model probabilities. We achieved a score of 67.73% in terms of Macro accuracy and a macro-averaged F1-score of 67.31%.

1 Introduction

Arabic is a complex language which presents significant challenges for natural language processing and its applications. Arabic is characterized by its plurality. It consists of a wide variety of languages, which includes the Modern Standard Arabic (MSA), and a set of various dialects differing according to regions and countries.

Language identification is the task of identifying the language of a given text. It is an important preprocessing step for many Natural Language Processing (NLP) tasks such as machine translation (Meftouh et al., 2018; Harrat et al., 2017), sentiment analysis (Rana et al., 2016; Abdul-Mageed et al., 2014; Saad et al., 2013), etc. In general, language identification is not a high challenging issue since this research has been supported for a long time and several machine learning techniques have been tested in this area that yielded to more or less good results. Nonetheless, in cases such as identifying languages from very little data, from mixed input or when the languages are extremely close to each other, the task becomes very challenging (Goutte et al., 2014).

This paper describes the submission of Loria (SMarT research group) to the Madar shared task on Arabic fine-grained dialect identification covering 25 specific cities from across the Arab World, in addition to Modern Standard Arabic (Bouamor et al., 2019). This shared task is the first to target a large set of dialect labels at the city and country levels. It has two subtasks.

Subtask 1: MADAR Travel Domain Dialect Identification.

Subtask 2: MADAR Twitter User Dialect Identification.

Our submission to this campaign is dealing with the first subtask. The remainder of this paper is organized as follows: in the next section, we discuss related work pertaining to Arabic dialect identification. Section 3 reviews the modeling choices we made for the shared task, and Section 4 describes results in detail.

2 Related Work

Several research works addressed the problem of Arabic dialect identification. The authors of Habash et al. (2008) presented standard annotation guidelines to identify the switching between the MSA and at least one dialect. These guidelines can be used to annotate large collections of data used for training and testing NLP tools. In Zaidan and Callison-Burch (2012), a large annotated dataset, created by harvesting an important number of reader commentaries on online newspapers content, is used to train and evaluate automatic classifiers for dialect detection and identification. The authors crowdsourced an annotation task to obtain sentence-level labels indicating what proportion of the sentence is dialectal, and which dialect the sentence is written in. The approach used in dialect identification relies on training language models for the different varieties of Arabic. Another work presents a supervised approach for performing sentence level dialect identification...
between Modern Standard Arabic and Egyptian Dialectal Arabic (Elfardy and Diab, 2013). The authors use token level labels to derive sentence-level features. These features are then used with other core and meta features to train a generative classifier that predicts the correct label for each sentence in the given input text. In addition to a multi-dialect, multi-genre, human annotated corpus, the authors in Cotterell and Callison-Bursh (2014) present the results of a language identification task extended to include 5 dialects. They considered Naïve Bayes and Support Vector Machines. The approach used by Darwish et al. (2014) for the identification of the Egyptian dialect was based on lexical, morphological and phonological information. They show that accounting for such information can improve dialect detection accuracy by nearly 10%. Using a set of surface features based on characters and words Malmasi et al. (2015) conduct three experiments with a linear SVM classifier and a meta-classifier using stacked generalization on the Multidialectal Parallel Corpus of Arabic (MPCA) compiled by Bouamor et al. (2014). They first conduct a 6-way multi-dialect classification task then investigate pairwise binary dialect classification and finally conduct cross-corpus evaluation on the Arabic Online Commentary (AOC) dataset. In Al-Badrashiny et al. (2015), the authors present a hybrid approach for performing token and sentence levels Arabic Dialect Identification. The token level component relies on a Conditional Random Field (CRF) classifier that take decisions based on several underlying components such as language models, a named entity recognizer and a morphological analyzer to label each word in the sentence. The sentence level component uses an ensemble of classifiers that models different aspects of the language. In another work, Al-Badrashiny and Diab (2016) present a system that detects points of code-switching in sentences between the MSA and dialectal Arabic. In Sadat et al. (2014), the authors present a bi-gram character-level model to identify the dialect of sentences, in the social media context, among dialects of 18 Arab countries. Bougrine et al. (2015) addressed the problem of spoken Algerian dialect identification by using prosodic speech information (intonation and rhythm). They performed an experiment on six dialects from different Algerian regions. In Salameh et al. (2018), the authors present the first system dealing with fine-grained dialect classification task and covering 25 specific cities from across the Arab World, in addition to Standard Arabic. For this purpose, they build several classification systems using a Multinomial Naïve Bayes classifier and exploring a large space of features.

3 The Modeling Choices

3.1 Data

For the experiments reported in this paper, we only use the training and the development data available in the subtask 1 of the shared task. The dataset of this subtask is the same as the one reported on Bouamor et al. (2018) and Salameh et al. (2018). It is composed of two corpora. The first (Corpus-26) is a collection of parallel sentences, built to cover the dialects of 25 cities from the Arab World, in addition to MSA. The training part consists of 1600 labeled instances per class, while the development part has 200 labeled instances per class. The second (Corpus-6) contains 10,000 additional sentences translated to the dialects of only five cities: Beirut, Cairo, Doha, Tunis and Rabat, in addition to MSA. They are splitted on two categories: 9,000 instances per language for the training and 1,000 instances per language for the development.

3.2 Method

In order to develop a language identification system that can distinguish between several Arabic dialects, we tested three methods namely simple neural networks (LSTM) (Sak et al., 2015), a method based on word embedding (Word2vec) (Mikolov et al., 2013) and Naïve Bayes classifiers. Given the limited size of the provided corpora, the first two methods have proven ineffective. We give in Table 1 the results we obtain using Corpus 26 in terms of Macro averaged F1-score, precision and recall.

Table 1: Macro averaged F1-score, Precision and Recall for Word2vec and LSTM method.

<table>
<thead>
<tr>
<th>Method</th>
<th>Corpus 26</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Precision</td>
</tr>
<tr>
<td>Word2vec</td>
<td>50.11</td>
</tr>
<tr>
<td>LSTM</td>
<td>58.04</td>
</tr>
</tbody>
</table>

We used a Naïve Bayes method because in the past, we did a comparative study of methods for Topic identification. This method for French leads
In this work we consider a Multinomial Na"ive Bayes classifier, in fact a study proposed in McCallum and Nigam (1998) showed that the multinomial model is found to be almost better than the multivariate Bernoulli model and the experimental results yielded to better results. So, we consider a Multinomial Na"ive Bayes classifier for this task. In this case, the term Multinomial Na"ive Bayes lets us know that each $p(f_i|c)$ (where $f_i$ is a feature and $c$ the category or the class) is a multinomial distribution, rather than some other distribution such as a Bernoulli distribution.

To develop our system, we used Python, relying on Scikit-Learn module (Pedregosa et al., 2011).

### 3.3 Features

A Na"ive Bayes model classifier identifies a category by calculating the distributions of the features within a category. It also assumes that each of the features it uses are conditionally independent of one another given a category. Identifying features is a critical step when applying Na"ive Bayes classifiers. That is why we did several experiments to select some adequate features. After several experiments, we selected for each sentence, the following 38 features as follows:

- A unigram of words,
- A bigram of words
- Character n-grams: from 1 to 5
- Character n-grams: from 1 to 5, by taking into account the spaces between words; in other words ngrams at the edges of words are padded with space. All the symbols of punctuation have been removed from the training, development and test data.
- 26 likelihoods estimated by the 26 unigram language models

For all the features, we use a special character to mark the start of the sentences. We utilize Term Frequency-Inverse Document Frequency (Tf-Idf) scores (Spärck Jones, 1972) as it has been shown to outperform count weights in several NLP applications.

### 4 Results and Discussion

For the purpose of this campaign, we built several systems using the model described in section 3. We did several experiments to determine the smoothing adding value, necessary for the Naive Bayes method, and we set it to 0.093 for all the systems. In Table 2, we report the results of all the experiments concerning the Multinomial Naive Bayes method. For the evaluation purpose, we use the Macro averaged F1-score which is retained as the official metric by the organizers of Madar shared task.

<table>
<thead>
<tr>
<th>Ngrams features</th>
<th>F1-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Word Char_wo Char_wi Dev Test</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>-</td>
</tr>
<tr>
<td>1-2</td>
<td>-</td>
</tr>
<tr>
<td>1-3</td>
<td>-</td>
</tr>
<tr>
<td>- 1-3</td>
<td>-</td>
</tr>
<tr>
<td>- 1-4</td>
<td>-</td>
</tr>
<tr>
<td>- 1-5</td>
<td>-</td>
</tr>
<tr>
<td>- - 1-3</td>
<td>59.66</td>
</tr>
<tr>
<td>- - 1-4</td>
<td>64.45</td>
</tr>
<tr>
<td>- - 1-5</td>
<td>66.50</td>
</tr>
<tr>
<td>- 1-5 1-5</td>
<td>66.92</td>
</tr>
<tr>
<td>1-2 1-5 1-5</td>
<td>69.06</td>
</tr>
<tr>
<td>1-2 1-5 1-5</td>
<td>69.09</td>
</tr>
</tbody>
</table>

First, we train the multinomial NB on word ngrams. The best results are achieved with the use of unigrams and bigrams. For higher order of n-grams, the performance of the model degrades due to the data sparsity. Then, we tested the effect of character ngrams features with (wi) and without (wo) taking into account the space at the end of the words. We experimented using the features of each option alone and combined. In Table 2 the symbol $x-y$ means that all the n-grams features from $x$ to $y$ of the corresponding column are taken into account in the classification.

In all the experiments, the best model is obtained for n ranging from 1 to 5. We remark that a classifier based on character ngrams features (1-5) outperforms the classifier based on word ngrams features by at least 3 points. Finally, the best classifier is the one using word unigrams and bigrams, and character ngrams ranging from 1 to 5 with and without space. The introduction of the language model features improved the result on the development corpus and reduced it on the test corpus.
We decided finally to participate to the campaign with the classifier including the language model parameters.

5 Conclusion

In this paper, we described the experiments we conducted as part of the MADAR shared task on Arabic fine-grained dialect identification. This task is the first covering the dialects of 25 specific cities from across the Arab World, in addition to MSA. Thus, we tested several systems exploring a large set of features. A blind run on the test set was then performed and submitted as part of the shared task. The Macro accuracy is 67.73% (macro-averaged F1-score 67.31%), placing our classifier first among 19 participants. This result shows that our approach despite its simplicity performs very well and even if it is ranked first, we need to make more efforts to make it powerful so that it can become an effective tool for the community.

References


