Machine Learning-Based Approach for Arabic Dialect Identification

Mahmoud S. Ali, Ahmed H. Ali, Ahmed A. El-Sawy, Hamada A. Nayel
Department of Computer Science
Faculty of Computers and Artificial Intelligence
Benha University
{mahmoud.hassan,ahmed.ali,ahmed.el_sawy,hamada.ali}@fci.bu.edu.eg

Abstract

This paper describes our systems submitted to the Second Nuanced Arabic Dialect Identification Shared Task (NADI 2021). Dialect identification is the task of automatically detecting the source variety of a given text or speech segment. There are four subtasks, two subtasks for country-level identification and the other two subtasks for province-level identification. The data in this task covers a total of 100 provinces from all 21 Arab countries and come from the Twitter domain. The proposed systems depend on five machine-learning approaches namely Complement Naïve Bayes, Support Vector Machine, Decision Tree, Logistic Regression and Random Forest Classifiers. F_1 macro-averaged score of Naïve Bayes classifier outperformed all other classifiers for development and test data.

1 Introduction

Today, huge amounts of data in text, picture, and video format are posted to social network sites, the general web, and mobile devices. Social networks like Twitter are based on interactions with fast temporal dynamics which generate a large variety of contents with their own characteristics which are difficult to compute with classical tools used on traditional texts like essays and articles (Sun et al., 2019).

This article focuses on dialect identification, which is a technology critical in tasks such as author profiling and other NLP downstream tasks such as sentiment analysis, POS tagging, text summarization, among others. In dialect identification, we face some questions: how to find differences in writing style on social networks between men and women, age groups, or location. The answers to these questions are important for the new problem we face in the era of social networks such as fake news, plagiarism, and identity theft (Mansour et al., 2020).

Recently, research community concerning Arabic natural language processing focussed on dialect identification for Arabic(Bouamor et al., 2019; Salameh et al., 2018; Abdul-Mageed et al., 2021, 2020). A shared task for Nuanced Arabic Dialect Identification (NADI 2020) has been organized to identify the dialect in Arabic Tweets.

We propose five models to perform Dialect Identification for Arabic tweets. The Term Frequency Inverse Document Frequency (TF/IDF) algorithm is implemented for feature extraction after preprocessing process. Complement Naïve Bayes (CNB), Decision Tree (DT), Logistic Regression (LR), Random Forest (RF), and Support Vector Machine (SVM) classifiers are used in the classification step. The results of development set showing that CNB classifier outperforms all other classifiers.

2 Data

The NADI 2021 datasets were the main corpus used for training and testing the system (Abdul-Mageed et al., 2021). The NADI 2021 datasets are in both Modern Standard Arabic (MSA) and dialectal Arabic (DA). For each of these varieties, the dataset is partitioned into three parts, the training part of 21,000 tweets, development of 5,000 tweets, and test set of 5,000 tweets The data sets are labeled in two levels: the first level (country level) of 21 countries and the second level (provinces level) of 100 provinces. The test set was published unlabelled, and the system output was evaluated by the NADI shared task team. The training set was used to train our models while the development set was used to optimize models' parameters. The distribution of data over the country and province classes is unbalanced.

Proceedings of the Sixth Arabic Natural Language Processing Workshop, pages 287–290
Kyiv, Ukraine (Virtual), April 19, 2021.

3 System Architecture

As shown in Figure 1, our system composed of four main stages. These are: Text Preprocessing, Feature Extraction, Classification, Evaluation and Testing. Text preprocessing, where we make some

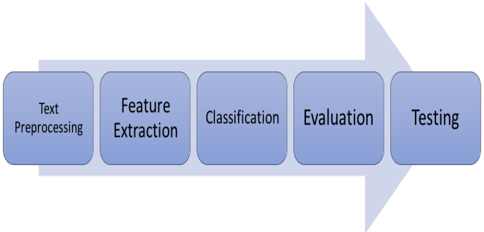

Figure 1: Stages of our system

cleaning and corrections on input as we will explain next. In the second stage important features from all documents have been selected. The third stage is classification, in which we classify each vector in the feature vectors to its correct class (correct Arabic dialect). The model has been tested in evaluation stage. At last, a blind test set has been used to predict the correct class (Arabic dialect).

3.1 Text Preprocessing

Arabic tweets are very noisy, so we removed URLs, emojis, Latin-characters, numbers, mentions, and any non-Arabic characters. Arabic hashtags were kept as they are, because they might contain important information (Touileb, 2020; Nayel, 2019). Preprocessing is a step used in almost all NLP applications (Ananthakrishnan et al., 2019; Nayel et al., 2019). In the proposed models, this stage consists of some steps as the following:

- Non-Arabic letter removal, which is achieved by deleting English letters, special symbols, numbers, Twitter markup, Emoticons, repetition letters etc.

- Text refinement, which is carried out to normalize different forms of some Arabic characters to unique form like," ي " (an Arabic letter pronounced Yaa) and " ى " to be " ى", removing redundant Arabic forms like, " ال " (pronounced al and it is used as determiner), and excessive character repetitions, Kashida "tatweel" and diacritics.

- Removing Punctuation, in which we removed punctuation marks such as $\{'+', '\#', '-', '\$', ...\}$ which are increasing the dimension of feature space with redundant features.

- Reduction of letter repetition, because all tweets in Twitter do not follow the standard rules of the language especially Arabic language. A common manner of users is to repeat a specific letter in a word. Cleaning the tokens from these redundant letters helps in feature space reduction. In our experiments, the letter is assumed to be redundant if it is repeated more than two times. For example the words " ههههههه " ("hahahah" i.e. giggles) and " عاااااالم " (i.e. "global") containing redundant letter and will be reduced to " هه " and " عالم " respectively.

The class distribution is highly imbalanced, which could make a model biased towards certain classes. Therefore, random up-sampling for each data class was applied to match the size of the majority class, the Egyptian class (Aliwy et al., 2020).

3.2 Feature Extraction

In this study, we use TF/IDF with unigram features (words) to represent each tweet as a feature vector, and the value of each cell in the vector is the weight of each feature within each tweet vector and is determined by the following formula:

$$w_{ij} = tf_{ij} * \log\left(\frac{N+1}{df_i + 1}\right)$$

where, w_{ij} is the weight of word i in vector j, tf_{ij} is the count of word i in document j, N is the total number of tweets, and df_i is the count of word i in all tweets.

In TF/IDF, we used unigram model, which deal with a single word as a token. For example, the sentence "ريال مدريد كان الأفضل", which means Real Madrid was the best, has the following set of features { ريال ، مدريد ، كان ، الأفضل }.

3.3 Classification

The classification step was achieved by voting among five well-known and very different five classifiers (CNB, DT, LR, RF, and SVM). The dataset has two levels of classification, country-level with 21 class and province-level with 100 province. Each province related to one of 21 countries. The

Complement Naive Bayes (CNB) classifier was designed to correct the "severe assumptions" made by the standard Multinomial Naive Bayes classifier. It is particularly suited for imbalanced data sets and this is actually proved in the results.

Decision Tree (DT) classifier uses a decision tree as a predictive model to go from observations about an item represented in the branches to conclusions about the item's target value that is represented in the leaves. Logistic regression (LR) is a statistical model that in its basic form uses a logistic function to model a binary dependent variable. Random Forest (RF) is a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting.

Support Vector Machine (SVM) is a linear classifier which uses training samples or vectors close to the boundaries of classes as support vectors. SVM implemented for different NLP tasks effectively. SVM can be used for classifying non-linear data using kernel functions such as, Gaussian, RBF, or Linear (Nayel, 2020).

Algorithm	Country Level		Province Level	
	MSA	**DA**	**MSA**	**DA**
CNB	14.06	21.34	4.16	4.39
DT	11.23	12.19	3.33	3.23
LR	9.15	12.49	3.56	4.51
RF	13.17	14.30	4.09	3.88
SVM	11.37	15.97	3.82	4.73

Table 1: F_1-score of our models on DEV data.

Subtask	**1.1**	**1.2**	**2.1**	**2.2**
Rank	5	8	4	3
Precision	15.09	21.61	4.09	4.71
Recall	12.46	18.12	3.46	4.55
F_1-score	12.99	18.72	3.51	4.55
Accuracy	23.24	37.16	3.38	4.80

Table 2: Performance of our models on TEST data.

4 Results

We propose five machine learning algorithms for the task: CNB, DT, LR, RF and SVM classifiers. All algorithms were implemented on NADI shared task data set. There are four subtask: 1.1 for MSA-country-level, 1.2 for DA-country-level, 2.1 for MSA-province-level, and 2.2 for DA-province-level. The F_1score is the evaluation metric for all subtasks. The Complement Naïve Bayes (CNB) classifier outperformed all other classifiers in all subtasks and achieved the highest F_1-score ratio for the development set data. Table 1 shows the results for all runs of the development set classification.

The results of test set is given in Table 2. Concerning the evaluation of our submissions for test data, our model achieved 5th rank for subtask 1.1, 8th rank for subtask 1.2, 4th rank for subtask 2.1, and 3rd rank for subtask 2.2 among all submissions as shown in Table 2.

5 Discussion

We provided a description of our different models we developed for NADI 2021. The dataset of 21 country-level classes and 100 province-level classes were used to evaluate the systems. There were four subtasks, two subtasks for identifying MSA and two subtasks for identifying DA. The F_1-score results of classification were relatively low because:

1. The MSA data used in different Arabic countries may have a significant degree of similarity,

2. Some dialects are very close to some other dialects,

3. Tweet ambiguity, resulting from use of MSA or dialect sequences where the same sentence can be spoken in different countries but is written the same way,

4. The dialects of some provinces may be closer to those of a different neighbouring country than it is to the country to which the province belongs,

5. Use of location as a proxy for dialect may not be straightforward as organizers indicate in NADI 2020.

6 Conclusion

We can conclude that Arabic dialect identification is one of the most challenging tasks for many reasons mentioned. This paper proposed five classifier models for identifying Arabic dialect in Twitter. The results of training using Complement Naïve Bayes classifier achieved the best F_1 macro-averaged score as it is very good to deal with NLP depending on Naïve Bayes rule. In future work, different weighting scores can be used to improve the performance of classification, such as word embeddings.

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)*, pages 97–110, Barcelona, Spain (Online). Association for Computational Linguistics.

Muhammad Abdul-Mageed, Chiyu Zhang, Abdel-Rahim Elmadany, Houda Bouamor, and Nizar Habash. 2021. NADI 2021: The Second Nuanced Arabic Dialect Identification Shared Task. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop (WANLP 2021)*.

Ahmed Aliwy, Hawraa Taher, and Zena AboAltaheen. 2020. Arabic dialects identification for all Arabic countries. In *Proceedings of the Fifth Arabic Natural Language Processing Workshop*, pages 302–307, Barcelona, Spain (Online). Association for Computational Linguistics.

Haritha Ananthakrishnan, Akshaya Ranganathan, D. Thenmozhi, and Chandrabose Aravindan. 2019. Arabic author profiling and deception detection using traditional learning methodologies with word embedding. In *Working Notes of FIRE 2019 - Forum for Information Retrieval Evaluation, Kolkata, India, December 12-15, 2019*, volume 2517 of *CEUR Workshop Proceedings*, pages 100–104. CEUR-WS.org.

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.

Moataz Mansour, Moustafa Tohamy, Zeyad Ezzat, and Marwan Torki. 2020. Arabic dialect identification using BERT fine-tuning. In *Proceedings of the Fifth Arabic Natural Language Processing Workshop*, pages 308–312, Barcelona, Spain (Online). Association for Computational Linguistics.

Hamada Nayel. 2020. NAYEL at SemEval-2020 task 12: TF/IDF-based approach for automatic offensive language detection in Arabic tweets. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 2086–2089, Barcelona (online). International Committee for Computational Linguistics.

Hamada A. Nayel. 2019. NAYEL@APDA: Machine Learning Approach for Author Profiling and Deception Detection in Arabic Texts. In *Working Notes of FIRE 2019 - Forum for Information Retrieval Evaluation, Kolkata, India, December 12-15, 2019*, volume 2517 of *CEUR Workshop Proceedings*, pages 92–99. CEUR-WS.org.

Hamada A. Nayel, Walaa Medhat, and Metwally Rashad. 2019. BENHA@IDAT: Improving Irony Detection in Arabic Tweets using Ensemble Approach. In *Working Notes of FIRE 2019 - Forum for Information Retrieval Evaluation, Kolkata, India, December 12-15, 2019*, volume 2517 of *CEUR Workshop Proceedings*, pages 401–408. CEUR-WS.org.

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.

Yutong Sun, Hui Ning, Kaisheng Chen, Leilei Kong, Yunpeng Yang, Jiexi Wang, and Haoliang Qi. 2019. Author profiling in arabic tweets: An approach based on multi-classification with word and character features. In *Working Notes of FIRE 2019 - Forum for Information Retrieval Evaluation, Kolkata, India, December 12-15, 2019*, volume 2517 of *CEUR Workshop Proceedings*, pages 105–109. CEUR-WS.org.

Samia Touileb. 2020. LTG-ST at NADI shared task 1: Arabic dialect identification using a stacking classifier. In *Proceedings of the Fifth Arabic Natural Language Processing Workshop*, pages 313–319, Barcelona, Spain (Online). Association for Computational Linguistics.