Arabic Regional Dialect Identification (ARDI) using Pair of Continuous Bag-of-Words and Data Augmentation

Ahmed H. AbuElAtta, Mahmoud Sobhy*, Ahmed A. El-Sawy, Hamada Nayel
Department of Computer Science-Faculty of Computers and Artificial Intelligence,
Benha University, Egypt

***Abstract*—Author profiling is the process of finding characteristics that make up an author's profile. This paper presents a machine learning-based author profiling model for Arabic users, considering the author's regional dialect as a crucial characteristic. Various classification algorithms have been implemented: decision tree, KNN, multilayer perceptron, random forest, and support vector machines. A pair of Continuous Bag-of-Word (CBOW) models has been used for word representation. A well-known data set has been used to evaluate the proposed model and a data augmentation process has been implemented to improve the quality of training data. Support vector machines achieved a 50.52% f1-score, outperforming other models.**

***Keywords*—Dialect identification; continuous Bag-of-Words; data augmentation; text classification.**

I. Introduction

The Arabic language presents a captivating and challenging duality. Its source stems from its historical significance, the strategic importance of its native speakers and their region, along with its abundant cultural and literary legacy. Simultaneously, its complex linguistic framework poses difficulties [1]. More than 330 million people speak Arabic as their native tongue, and as a Semitic language, it has several distinctive linguistic features such as right-to-left writing, and the presence of a dual number of nouns. One of the most prominent features observed in Semitic languages, including Arabic, is the utilization of both female and male genders alongside the root. This aspect stands out significantly and distinguishes these languages [1].

The various Arabic dialects that exist today are together referred to as the Arabic language. There is one "written" form that is used to write Modern Standard Arabic (MSA), while numerous "spoken" variants based on the regional dialect have been used. Due to its use in official settings and written communication, the sole variant undergoes standardization, regulation, and formal instruction in educational institutions. When compared to MSA, regional dialects, which are mostly employed for spoken communication and daily interactions, are still slightly absent from written communication. The same letters used in MSA and the same (mainly phonetic) spelling conventions of MSA can, however, be utilized to create Dialectal Arabic (DA) text [2].

The possible data source is of importance to two key sectors, the commercial sector where marketing intelligence places a larger value on client information including age, gender, nationality, and native language, and the security industry is responsible for guarding against crimes like plagiarism and identity theft, among others, on the internet. As a result, the research community encourages scientists to find and create efficient procedures and methodologies in related disciplines like plagiarism detection and author profiling [3].

The use of DA is prevalent on social media platforms. Computational linguists could generate vast datasets that could be employed in statistical learning environments by gathering information from such sources. It is a challenge to differentiate and separate the dialects from one another; as all Arabic dialects share the same character set and a large portion of their vocabulary. There are six main Arabic regional dialects in addition to MSA, which is typically not commonly spoken as a primary language. Fig. 1 shows the regional dialects of Arabic world.

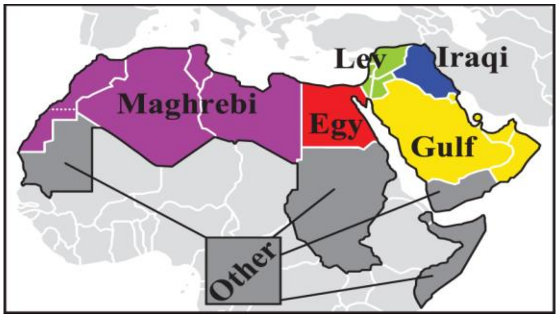

Fig. 1. Regional dialects of Arabic world [2].

- Egyptian: The dialect that is most generally known and understood, due to Egypt's strong film and television industries. As well as it's significant influence throughout a significant portion of the 20th century [4].
- Levantine: A group of dialects linked to Aramaic that sound somewhat different and have different intonations but are substantially comparable when written [5].
- Gulf: The regional dialect that is most like MSA as the current version of MSA is developed from an Arabic dialect that originated in the Gulf region. Compared to other variants, the Gulf dialect has retained a greater portion of MSA's verb conjugation, despite the variances [6].

- Iraqi: Although it has distinctive qualities of its own in terms of prepositions, verb conjugation, and sound, it is occasionally regarded as one of the Gulf dialects [6].

- Maghrebi: French and Berber had a big influence on this dialect. In spoken form, the western-most dialects may be incomprehensible to speakers from other Middle Eastern countries [7].

Due to the complexity of the Arabic language's morphology, the dearth of datasets, and most of the available datasets are imbalanced. Arabic research obtained little attention in its primary phases, especially regarding dialect identification. There are many challenges caused by the high similarity of dialects, particularly in short phrases, such as:

- The same word might have similar meanings in different dialects, for example, the word "**كتاب**" (pronounced "Ketab") means book.

- For the same dialect, there are different short phrases with the same meanings. For example, in Egyptian dialect, the words "**طيب**" (pronounced "Tayb"), "**حاضر**" (pronounced "Hader"), "**عنيا**" (pronounced "Enya"), "**انت تؤمر**" (pronounced "Enta To'mor"), "**إشطه**" (pronounced "Eshta") means "ok".

Effective dialect identification improves the performance of different applications and services, such as machine translation, Automatic Speech Recognition (ASR), remote access, e-commerce, e-learning, and exposing forensic evidence. Arabic dialect identification has been performed at the regional-level (e.g., Levant, Gulf) [8], country-level (e.g., Egypt, Saudia Arabia) [9], and province-level (e.g., Cairo, Al-Madinah) [10]. This work concentrates on the challenge of Arabic Regional Dialect Identification (ARDI) for social media users. We propose machine learning-based classification models to perform ARDI for Arabic tweets. A word embedding model has been used for word representations and a data augmentation process has been applied to improve the quality of data. In the classification step, Decision Tree (DT), K-Nearest Neighbor (KNN), Multi-Layer Perceptron (MLP), Random Forest (RF), and Support Vector Machines (SVM) algorithms were used.

The remaining portions of the paper are structured as follows: Related work is introduced in Section II. Section III describes the dataset that has been used for model development. Section IV introduces the general architecture of the proposed model. The results obtained by the proposed model presented in Section V. The detailed explanation of the results is represented in Section VI, while Section VII concludes the paper.

II. Related Work

There are major efforts that have been carried out for ARDI; some of these works will be described in this section. The First Nuanced Arabic Dialect Identification Shared Task (NADI 2020) has been presented in [11]. This shared task includes the identification of Arabic countries as subtask-1, and the identification of Arabic provinces as subtask-2. The dataset for NADI 2020 covers 21 Arab countries including 100 provinces obtained from Twitter. The baseline, Google's mBERT, model was fine-tuned with 50 tokens as a sequence's maximum length and 8 batches. Various approaches have been applied for NADI such as Machine Learning (ML) approaches and Deep Learning (DL) approaches, and the best performed model achieved a 26.78% f1-score for the first subtask and 6.39% for the second subtask. NADI 2021 was the second shared task that aimed at identifying the linguistic diversity of brief texts based on small geographical regions of origin in Arabic dialects [10]. In NADI 2021, an unlabeled corpus for 10M tweets has been added for optional use. The same baseline model for NADI 2020 was fine-tuned in addition to the ML and DL approaches. The best winner model reported 22.38%, 32.26%, 6.43%, and 8.60% f1-score for country-level-MSA, province-level-MSA, country-level-Dialectal Arabic, and province-level-Dialectal Arabic, respectively. Another NADI shared task in 2022 [9] aimed at the identification of Arabic country-level dialects. Three baselines were finetuned in NADI 2022, Baseline-mBERT, Baseline-XLMR, and Baseline-MARBERT in addition to some pre-trained models based on the BERT model. The top systems reported 36.48% and 18.95% f1-score for Test-A and Test-B sets respectively.

Antoun et al. [12], introduced the AraBERT model which trained on a large Arabic corpus and achieved state-of-the-art on various Arabic NLP tasks, including sentiment analysis, named entity recognition, and question answering. The BERT-base configuration was 12 encoder blocks, 768 hidden dimensions, 12 attention heads, 512 maximum sequence length, and a total of 110M parameters. The model outperforms the multilingual version of BERT and other previous approaches. Another transformer-based model designed for Arabic language understanding was introduced by Abdul-Mageed et al. [8]. They introduced ARBERT and MARBERT, bidirectional transformers for Arabic language processing, focusing on MSA and Arabic Dialects respectively. A random 1B Arabic tweets were selected to train MARBERT, and a dataset of about 6B tweets was formed. Tweets greater than two Arabic words were only included. MARBERT was trained for 36 epochs with 256 batch size and 128 sequence length and achieved high scores in various Arabic dialects datasets.

Talafha et al. [13] used BERT architecture for NADI and achieved a 26.78% f1-score. Also, Gaanoun and Benelallam [14] presented an Arabic-BERT model combined with ensemble methods and data augmentation for NADI. The Arabic-BERT model was trained on the provided training data; then data augmentation was performed by splitting the training data into three parts and mixing them for each country. The augmented data was used to train multiple models, including the "Mix" model, which showed good performance and obtained an f1-score of 23.26% and 5.75% for country-level and province-level respectively. A combination of BERT and N-GRAM characteristics was presented in [3]. The authors investigated the task for the identification of dialects at the national and provincial levels. They introduced an ensemble model that achieved promising results. M-NGRAM uses TF-IDF with character and word n-grams and a Stochastic Gradient Descent (SGD) classifier. The ensemble method achieved a f1-score of 25.99% and 6.39% for country-level identification and province-level identification, respectively.

The contributions of this paper include: (1) develop five classification models based on a pair of Continuous Bag-of-Words model (CBOW) for ARDI, (2) build a new corpus from various datasets to use for building our CBOW model, (3) apply the data augmentation process to enhance the quality of training data.

III. Data

The dataset that has been used in this research is ArSarcasm [8]. Which is a collection of Arabic sentiment analysis datasets called SemEval 2017 [15] and ASTD [16]. The dataset contains 10,547 tweets modified by adding dialect labels. The distribution of the train set, and test set over all classes is shown in Table I.

TABLE I. Distribution of Training Set and Test Set in each Class of ArSarcasm Dataset

Arabic Region	Number of documents in train set	Number of documents in test set
msa	5652	1410
egypt	1904	479
levant	439	112
gulf	414	105
maghreb	28	4
Total	8,437	2,110

The dataset contains the following fields:

- tweet: the text of the original tweets.
- sarcasm: a Boolean value indicating whether a tweet is sarcastic.
- sentiment: the new annotation's sentiment (good, neutral, or negative).
- source: the original tweet's SemEval or ASTD source.
- dialect: the Arabic regional dialect used in tweets, msa, egypt, levant, gulf, and maghreb, which were shown in Section I.

IV. Methodology

The proposed model uses classic ML-based classifiers integrated with a data augmentation approach for word representation. As shown in Fig. 2, the proposed system is divided into five primary phases, including data augmentation, text preprocessing, feature extraction, classification, and evaluation.

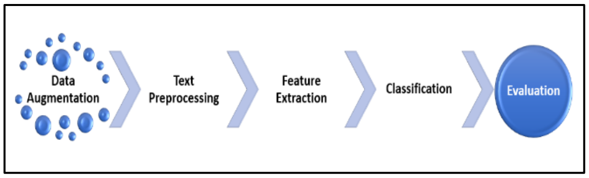

Fig. 2. General architecture of the proposed system.

In the data augmentation phase, the training data was augmented with extra Arabic text from itself and other datasets. Some text cleaning steps and adjustments were applied in the text preprocessing phase. In the feature extraction phase, all documents have been represented based on the embeddings of words in each document. In the classification phase, we classify each vector in the feature vectors to its correct class. Finally, the proposed model has been evaluated by measuring its performance in the evaluation phase. All steps are explained in the following sections.

A. Data Augmentation

- Data augmentation basically allows one to artificially increase the training set by making updated copies of training samples using existing data [17]. It involves modifying the dataset slightly or creating new data points using deep learning approaches. Data augmentation is used for several purposes; (1) to prevent overfitting; (2) when the initial training set is insufficient; (3) to increase the accuracy of models; and (4) to handle the unbalanced dataset. As shown in Table I, the training data is unbalanced as “msa” and “egypt” classes were much bigger than the other classes, as well as the portion of class “maghreb” is too small. In this work, several text augmentation approaches have been used, like:
- Rearranging words or sentences in a random manner.
- Substituting words with their synonyms.
- Rephrasing sentences using the same meanings.
- Insertion or deletion of words at random.

Furthermore, to increase the quality of the dataset, the original training data has been augmented with other datasets in the same domain. In this step, we selected some datasets which have the same Arabic dialect texts as our dataset. Especially, we selected records of data with the same regions in our dataset. The first external dataset is NADI 2022 [9] which focused on nuanced Arabic dialect identification at country-level for Arabic tweets and covers 18 dialects (a total of approximately 20K tweets). The second external dataset is NADI 2021 [10], which covers MSA and DA. The dataset contains a training set of 21,000 tweets, a development set of 5,000 tweets, and a test set of 5,000 tweets. Habibi is the third external dataset [18] which is the earliest Arabic song lyrics corpus. More than 30,000 Arabic song lyrics by vocalists from 18 different Arabic countries are included in the corpus, which includes songs in six Arabic dialects.

TABLE II. Distribution of Training Set for each Class before and after the Augmentation Process.

Class (Arabic region)	Number of documents before augmentation	Number of documents after augmentation
msa	5652	5652
egypt	1904	6187
levant	439	4227
gulf	414	4484
maghreb	28	3554
Total	8,437	24.104

More than 500,000 sentences (song verses) and more than 3.5 million words make up the lyrics [18]. The benefit of this corpus is that all words are written in DA not in MSA. After the data augmentation step was done, the dataset became balanced somewhat and the next step was data preprocessing. Table II shows the distribution of training data in each class before and after the augmentation process.

B. *Text Preprocessing*

To get the data ready for training, light preprocessing has been used which preserves a true representation of the text that naturally appears. Emojis, Latin letters, URLs, mentions, numerals, and non-Arabic characters were all excluded from the data because Arabic texts, particularly those found on social media, are unstructured and exceedingly loud. In addition, the following steps have been implemented.

- Convert the various forms of Arabic characters into their unique forms, such as "ة " (pronounced as Haa) and " ه " to be " ه".

- Deleting extraneous Arabic forms, such as," ال " (pronounced "al") and it operates as a determiner.

- Deleting punctuation marks such as {'?'; '.'; '!'; '$'} which make more unnecessary features that can expand the dimension of the feature space.

- Reducing the letter repetition since Arabic tweets tend to be less structured. Clearing the letters from the extraneous tokens helps in reducing feature space. In this work, we considered the letter, which is repeated more than twice as redundant. For example, the word "كاااااامل "(pronounced as "Kamel") which means "complete" will be decreased to "كاامل ", also the word "رهيييييب"(pronounced as "Rahib") which means "awesome") will be reduced to "رهييب ".

C. *Feature Extraction*

In this phase, all tweets are represented as feature vectors, each of which contains an embedding for each word in the tweet. A variety of approaches are employed to get the word embedding vector from the context in which the words are found. In this study, a pair of Word2vec models has been used. Google has suggested the Word2vec neural network [19] to analyze text input. The Word2vec model is a neural network with three layers: an input layer, an output layer, and a hidden layer without activation function. Additionally, the number of neurons in the hidden layer is the same size as the word embeddings' vector's dimensions. The Word2Vec model makes use of huge datasets during training to precisely capture the semantic and syntactic structure of the words, allowing for the efficient measurement of word similarity [19].

Continuous Bag of Words (CBOW) and Skip-gram are the two learning models included in Word2Vec: CBOW predicts the word given its context, while Skip-gram predicts the context given a word as shown in Fig. 3. The window size and vocabulary size are two hyperparameters that are shared by the two methods. The window size indicates the number of words in the context. Given the near future and historical words, the CBOW technique classifies the projected middle word using a log-linear classifier.

The number of words in the context is equal to the size of the sliding window; for example, if the sliding window is seven words, then there are six words in the context. Additionally, while predicting a word, the context of the preceding three words and the succeeding three words of the middle word must be considered.

The first Word2Vec model that has been used was built using UNLABELED-10M, a set of 10 million unlabeled Arabic tweets provided by NADI [9] in the form of tweet IDs. A ready implementation of Word2Vec model, gensim [20], using CBOW has been used to generate word vectors of size 300.

Nevertheless, the embedding vectors for words in the first model were not enough to cover all the words in the dataset. So, we created another Word2Vec model. We employed CBOW to generate the word vectors as it has higher computing speed, and it is more efficient with frequent words than Skip-gram [21]. The vocabulary has been built from the entire training data and some external aforementioned datasets, NADI 2021 shared task dataset [10], and Habibi corpus [18]. Previously, if the embedding vector did not exist in the corpus, the vector was set randomly. Now, it is obtained from the second corpus, and this increases the correctness of the vector. Following the training phase, a vector is used to represent each word.

Next, we construct the high dimension matrix. Rows in the matrix represent the training tweets and columns represent words. The classification phase, which is described in the following section, follows the creation of the feature vector matrix for all training instances.

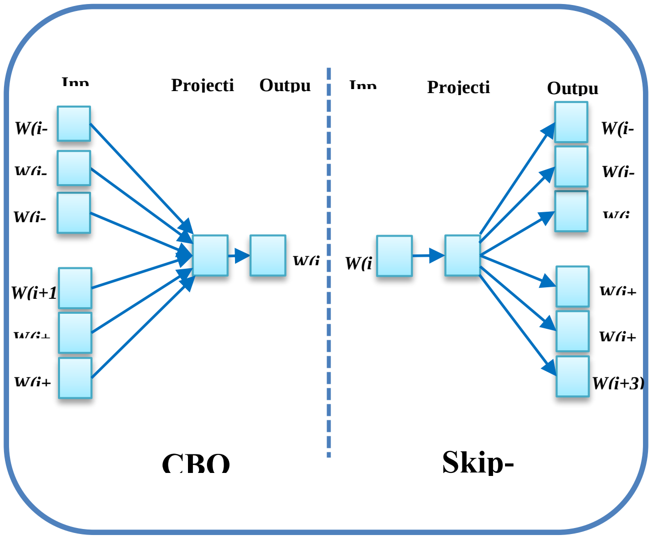

Fig. 3. The architecture of CBOW model.

D. *Classification*

Five classification algorithms, DT, KNN, MLP, RF, and SVM, were used in this study. To improve the performance of these classifiers, their hyperparameters were utilized.

The SVM is a linear classifier that uses training samples close to the borders of classes [22]. The SVM model uses kernel functions for classifying non-linear data such as linear, sigmoid, and Radial Basis Function (RBF) kernels, which were used in this work. The KNN algorithm assumes that the new sample and the available samples are similar, and it places the new sample in the category that resembles the available categories [23]. The DT classifier [24] utilizes the decision tree as a model for making pre-dictions based on observations about the items that are represented in the tree's branches to inferences about the target value of items that are represented in the decision tree's leaves.

The RF is an average-based meta-estimator that is used to increase predictive accuracy and reduce overfitting by applying several decision tree classifiers to various dataset sub-samples [25]. The MLP is a completely connected class of feedforward Artificial Neural Network (ANN) [26]. A typical MLP has an input layer, a hidden layer, and an output layer, which make up together less than or equal to three layers of nodes. Every node uses a nonlinear activation function, apart from the input nodes.

E. Evaluation

All algorithms mentioned in the paper that were evaluated on the ArSarcasm dataset. The evaluation metrics that have been used are Accuracy (Acc), Precision (P), Recall (R), and f1-score [27]. Accuracy measures the number of truly classified tweets divided by all tweets. Precision is another metric that calculates the number of correctly classified tweets divided by all classified tweets. Another metric is Recall, which calculates the number of correct correctly classified tweets divided by all correct tweets. The macro-averaged f1-score is the official metric for most NLP tasks as it is the most basic aggregation for the f1-score. The macro-averaged f1-score is the unweighted mean of the f1-scores determined for each class. The formula for macro f1-score [27] is:

$$f1\text{-}score = 2 * \frac{R*P}{R+P} \quad (1)$$

V. Experimental Results and Discussion

To evaluate the proposed model, several experiments have been carried out using different parameters as shown in Table III.

TABLE III. PARAMETERS OF SOME PROPOSED MODELS.

Algorithm	Parameters
SVM	Kernel functions = {linear, sigmoid, RBF}
KNN	Number of neighbors (*n*) = {30, 40, 50}
MLP	Number of hidden layers (*h*) = {5, 10, 20}

The proposed models have been tested using two variations of word embedding models. The results of the proposed models with the first embedding (UNLABELED-10M) without augmentation and with augmentation are shown in Table IV and Table V respectively. The results of the proposed models with the second embedding model (CBOW) without augmentation and with augmentation are shown in Table VI and Table VII respectively.

As shown in Table I above, the test data were unbalanced; the number of documents with the label (msa) is 1410, whereas it was only 4 with the label (maghreb). This was a big challenge to classify at least two documents with the label (maghreb) correctly. Fig. 4 shows the confusion matrix plot for the SVM (RBF) classifier, which has the best results.

It is clear that, the SVM (RBF-kernel) with the augmentation of data and the pair of Word2Vec models outperformed all other classifiers. Table IV shows the results of using the CBOW model, UNLABELED-10M, without data augmentation. The SVM classifier with RBF kernel function has achieved the highest accuracy, precision, and f1-score while MLP with five hidden layers has achieved the highest recall. Table V shows that the SVM classifier with RBF kernel function has achieved the highest accuracy, and F1-score, while MLP with five hidden layers has achieved the highest precision and the highest recall has achieved by MLP with 10 hidden layers.

TABLE IV. PERFORMANCE OF USING UNLABELED-10M MODEL (WITHOUT AUGMENTATION).

Algorithm	Parameter	P	R	f-score	Acc
SVM	linear kernel	46.683	39.737	41.902	77.014
	sigmoid kernel	39.296	35.618	36.684	70.900
	RBF kernel	**52.495**	40.650	43.478	**78.027**
KNN	n = 30	47.853	38.152	40.250	77.014
	n = 40	48.833	37.947	40.057	77.156
	n = 50	48.261	37.777	39.198	76.730
DT		32.024	32.815	32.365	64.739
RF		43.333	33.292	35.057	76.398
MLP	h = 5	44.125	45.456	42.512	74.041
	h = 10	47.568	**47.453**	44.542	73.652
	h = 20	48.225	47.342	**44.654**	72.850

TABLE V. PERFORMANCE OF USING UNLABELED-10M MODEL (WITH AUGMENTATION).

Algorithm	Parameter	P	R	f-score	Acc
SVM	linear kernel	48.195	45.783	43.214	75.877
	sigmoid kernel	39.896	42.723	40.115	69.384
	RBF kernel	**56.008**	46.530	**46.050**	**78.341**
KNN	n = 30	48.460	38.436	40.570	77.204
	n = 40	47.262	37.287	40.900	69.005
	n = 50	46.334	36.564	38.152	68.512
DT		35.701	36.096	35.849	67.915
RF		43.420	30.964	32.567	75.450
MLP	h = 5	49.105	46.466	43.758	75.071
	h = 10	48.940	**48.403**	45.655	74.739
	h = 20	47.334	47.042	44.259	73.791

In Table VI, the results of using two CBOW models, UNLABELED-10M, and our own CBOW model, without data augmentation. This table shows that the SVM classifier with RBF kernel function has achieved the highest accuracy and recall while MLP with five hidden layers has achieved the highest f1-score and precision. Table VII shows the results of using the same pair of CBOW models in Table V after the data augmentation process was done. This table shows that the SVM classifier with RBF kernel function has achieved the highest performance for all metrics.

TABLE VI. PERFORMANCE OF USING PAIR CBOW MODELS (WITHOUT AUGMENTATION).

Algorithm	Parameter	P	R	f-score	Acc
SVM	linear kernel	49.159	44.001	45.777	77.488
	sigmoid kernel	38.310	35.141	36.059	70.900
	RBF kernel	**55.018**	41.684	44.556	**78.863**
KNN	n = 30	48.460	38.436	40.570	77.204
	n = 40	49.090	38.015	40.338	76.967
	n = 50	47.450	37.263	39.274	76.872
DT		33.360	34.956	33.999	65.355
RF		46.045	33.991	36.091	76.730
MLP	h = 5	49.250	**44.225**	**46.016**	77.583
	h = 10	48.068	43.591	45.230	77.204
	h = 20	46.603	42.883	44.259	77.062

TABLE VII. PERFORMANCE OF USING PAIR CBOW MODELS (WITH AUGMENTATION).

Algorithm	Parameter	P	R	f-score	Acc
SVM	linear kernel	48.820	50.844	47.087	76.303
	sigmoid kernel	40.619	41.186	38.478	69.100
	RBF kernel	**50.294**	**55.893**	**50.534**	**77.251**
KNN	n = 30	47.557	47.748	41.723	68.578
	n = 40	48.314	47.672	41.506	69.194
	n = 50	48.049	48.080	41.614	69.431
DT		33.013	48.632	32.989	57.630
RF		46.795	46.722	42.798	74.787
MLP	h = 5	47.185	50.548	46.610	75.640
	h = 10	49.018	52.481	47.272	75.924
	h = 20	46.714	54.902	46.545	74.313

TABLE VIII. F1-SCORE OF OUR BEST MODEL AND PREVIOUS MODELS.

Algorithm	F1-score
mBERT [8]	43.81
XLM-R_L [8]	41.83
AraBERT [8]	47.54
SVM-RBF-With-Augmentation	**50.53**

Table VIII compares the performance of the proposed model and state-of-the-art in terms of f1-score. The results show that the proposed model outperformed the state-of-the-art models using low resources than those previous models which use huge resources to train language models using billions of words; as the proposed model depends on two embeddings and augmented data.

Fig. 4 shows the change in the values in the confusion matrix with the three attempts we made in the augmentation step. Let's focus on the true and predicted values of maghreb. In Fig. 4(a), without augmenting training data, the four test samples were misclassified. In Fig. 4(b), after the augmentation of the training samples of the maghreb class, 25% of the test samples were correctly classified. In Fig. 4(c), after the augmentation of all training samples, 50% of the test samples were correctly classified.

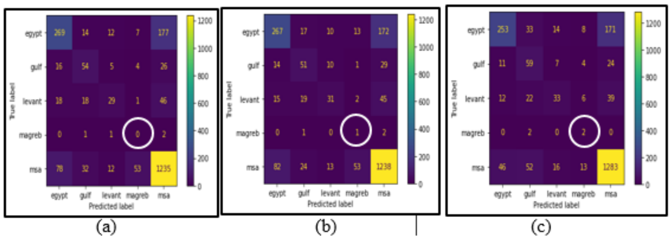

Fig. 4. Confusion matrix of classes after applying SVM with RBF kernel: (a) without augmentation; (b) with augmentation of magreb documents only; (c) with augmentation of documents of all classes.

I. CONCLUSION

Due to its complexity, ARDI becomes a challenge. This work proposed a machine learning-based model that implements DT, KNN, MLP, RF, and SVM for ARDI. A pair of CBOWs has been used for data representation and a data augmentation approach has been implemented to overcome the problem of imbalanced data. SVM reported 50.53% f1-score which was higher than previous work. The results proved that using a pair of CBOW models is better than using only one and proved that data augmentation was very useful for improving the quality of data. In future work, different representation models can be used to improve the performance of ARDI such as BERT models.

Authors' Contribution: Conceptualization, M.A. and H.N.; methodology, M.A. and H.N.; software, A.A. and M.A.; validation, H.N., A.E. and A.A.; formal analysis, M.A. and H.N.; resources, M.A. and H.N.; data curation, A.A. and A.E.; writing—original draft preparation, M.A. and H.N.; writing—review and editing, A.A. and A.E.; visualization, M.A. and H.N.; supervision, A.E.; All authors have read and agreed to the published version of the manuscript.

Data Availability Statement: the dataset that has been used in this article is open source and can be downloaded from https://github.com/iabufarha/ArSarcasm.

Conflicts of Interest: We confirm that neither the manuscript nor any parts of its content are currently under consideration or published in another journal. All authors have approved the manuscript and agree with its submission to the journal "Information".

REFERENCES

[1] I. Al-Huri, "Arabic Language: Historic and Sociolinguistic Characteristics English Literature and Language Review Arabic Language: Historic and Sociolinguistic Characteristics," vol. 1, no. 4, pp. 28–36, 2015, doi: 10.13140/RG.2.2.16163.66089/1.

[2] O. F. Zaidan and C. Callison-Burch, "Arabic Dialect Identification," 2014, doi: 10.1162/COLI.

[3] A. Abbassi, S. Mechti, L. Hadrich Belguith, and R. Faiz, "Author Profiling for Arabic Tweets based on n-grams." [Online]. Available: http://www.internetlivestats.com/

[4] N. Haeri, "Sacred language, ordinary people: Dilemmas of culture and politics in Egypt," Sacred Language, Ordinary People: Dilemmas of Culture and Politics in Egypt, pp. 1–184, Jan. 2003, doi: 10.1057/9780230107373/COVER.

[5] A. H. Aliwy, H. A. Taher, and Z. A. Abutiheen, "Arabic Dialects Identification for All Arabic countries," pp. 302–307, 2020.

[6] R. Bassiouney, A. Sociolinguistics, and M. Amara, "Reem Bassiouney: Arabic Sociolinguistics," Language Policy 2010 9:4, vol. 9, no. 4, pp. 379–381, May 2010, doi: 10.1007/S10993-010-9169-0.

[7] M. Tilmatine, "Substrat et convergences: le berbère et l¿arabe nord-africain," 1999.

[8] M. Abdul-Mageed, A. Elmadany, and E. M. B. Nagoudi, "ARBERT & MARBERT: Deep Bidirectional Transformers for Arabic," Dec. 2020, [Online]. Available: http://arxiv.org/abs/2101.01785

[9] M. Abdul-Mageed, C. Zhang, A. Elmadany, H. Bouamor, and N. Habash, "NADI 2022: The Third Nuanced Arabic Dialect Identification Shared Task," Oct. 2022. Available: http://arxiv.org/abs/2210.09582

[10] M. Abdul-Mageed, C. Zhang, A. Elmadany, H. Bouamor, and N. Habash, "NADI 2021: The Second Nuanced Arabic Dialect Identification Shared Task," Mar. 2021. Available: http://arxiv.org/abs/2103.08466

[11] M. Abdul-Mageed, C. Zhang, H. Bouamor, and N. Habash, "NADI 2020: The First Nuanced Arabic Dialect Identification Shared Task." pp. 97–110, 2020. Accessed: Aug. 08, 2023. Available: https://aclanthology.org/2020.wanlp-1.9

[12] W. Antoun, F. Baly, and H. Hajj, "AraBERT: Transformer-based Model for Arabic Language Understanding," Feb. 2020. Available: http://arxiv.org/abs/2003.00104

[13] B. Talafha et al., "Multi-Dialect Arabic BERT for Country-Level Dialect Identification," 2020. Available: https://github.com/mawdoo3/Multi-dialect-Arabic-BERT

[14] K. Gaanoun and I. Benelallam, "Arabic dialect identification: An Arabic-BERT model with data augmentation and ensembling strategy," 2020.

[15] S. Rosenthal, N. Farra, and P. Nakov, "SemEval-2017 Task 4: Sentiment Analysis in Twitter," pp. 502–518, Accessed: Aug. 06, 2023. Available: https://trends24.in/

[16] M. Nabil, M. Aly, and A. F. Atiya, "ASTD: Arabic Sentiment Tweets Dataset," pp. 17–21, 2015, Accessed: Aug. 06, 2023. Available: https://github.com/boto/boto.

[17] C. Shorten, T. M. Khoshgoftaar, and B. Furht, "Text Data Augmentation for Deep Learning," Journal of Big Data 2021 8:1, vol. 8, no. 1, pp. 1–34, Jul. 2021, doi: 10.1186/S40537-021-00492-0.

[18] M. El-Haj, "Habibi-a multi Dialect multi National Arabic Song Lyrics Corpus," pp. 11–16, 2020, Accessed: Aug. 06, 2023. [Online]. Available: www.figure-eight.com/

[19] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space", Accessed: Aug. 06, 2023. [Online]. Available: http://ronan.collobert.com/senna/

[20] R. Rehurek and P. Sojka, "Gensim -- Statistical Semantics in Python," 2011.

[21] D. Suleiman and A. Awajan, "Comparative Study of Word Embeddings Models and Their Usage in Arabic Language Applications," ACIT 2018 - 19th International Arab Conference on Information Technology, Mar. 2019, doi: 10.1109/ACIT.2018.8672674.

[22] K. P. Ukey and A. S. Alvi, "Text Classification using Support Vector Machine", Accessed: Aug. 09, 2023. [Online]. Available: www.ijert.org

[23] S. Jiang, G. Pang, M. Wu, and L. Kuang, "An improved K-nearest-neighbor algorithm for text categorization," Expert Syst Appl, vol. 39, no. 1, pp. 1503–1509, Jan. 2012, doi: 10.1016/J.ESWA.2011.08.040.

[24] F. Harrag, E. El-Qawasmeh, and P. Pichappan, "Improving Arabic text categorization using decision trees," 2009 1st International Conference on Networked Digital Technologies, NDT 2009, pp. 110–115, 2009, doi: 10.1109/NDT.2009.5272214.

[25] N. Jalal, A. Mehmood, G. S. Choi, and I. Ashraf, "A novel improved random forest for text classification using feature ranking and optimal number of trees," Journal of King Saud University - Computer and Information Sciences, vol. 34, no. 6, pp. 2733–2742, Jun. 2022, doi: 10.1016/J.JKSUCI.2022.03.012.

[26] H. Alla, L. Moumoun, Y. Balouki, and J. Gou, "A Multilayer Perceptron Neural Network with Selective-Data Training for Flight Arrival Delay Prediction," Sci. Program., vol. 2021, Jan. 2021, doi: 10.1155/2021/5558918.

[27] H. Dalianis, "Evaluation Metrics and Evaluation," in Clinical Text Mining: Secondary Use of Electronic Patient Records, Cham: Springer International Publishing, 2018, pp. 45–53. doi: 10.1007/978-3-319-78503-5_6.