An Ensemble Framework for Imbalanced Arabic Text-Based Emotion Analysis

Fath Ahmed Qayed Aljradi [1,*]
Mohammed Albared [2]
Abdullah Saeed Ghareb [3]
Abdulaziz Thawaba [4]

Received: 14/11/2024
Revised: 16/11/2024
Accepted: 7/12/2024

[1] University of Saba Region, Yemen.
[2] University of Saba Region, Yemen. Email: dr.albared@gmail.com
[3] University of Saba Region, Yemen. Email: aghurieb@usr.ac
[4] University of Saba Region, Yemen. Email: azizth@usr.ac
* Corresponding Author's Email: Fateh736481212@gmail.com

An Ensemble Framework for Imbalanced Arabic Text-Based Emotion Analysis

Fath Ahmed Qayed Aljradi
University of Saba Region, Yemen
Fateh736481212@gmail.com

Mohammed Albared
University of Saba Region, Yemen
dr.albared@gmail.com

Abdullah Saeed Ghareb
University of Saba Region, Yemen
aghurieb@usr.ac

Abdulaziz Thawaba
University of Saba Region, Yemen
azizth@usr.ac

Abstract— Abstract. Text analysis involves extracting knowledge from textual data for various applications. Emotion analysis can be conducted through multiple methodologies and serves a diverse array of purposes. In contemporary society, the sharing of experiences on social media platforms has become increasingly prevalent. For instance, Twitter serves as a valuable data source for organizations seeking to assess public opinions, sentiments, and emotional responses. Both organizations and individuals are keen to leverage social media for understanding public sentiment, extracting emotions, and gauging perspectives on specific issues; however, the field of emotion detection has received relatively limited focus. Previous studies have primarily explored emotional classifications within the text, particularly in Arabic content. The imbalance in datasets containing Arabic texts adversely impacts the classification process's effectiveness. Consequently, this research introduces an ensemble learning framework aimed at addressing this challenge, employing the Synthetic Minority Oversampling Technique (SMOTE) to achieve data balance, alongside Support Vector Machine (SVM), Naive Bayes (NB), and K-Nearest Neighbors (KNN) algorithms for emotion analysis. The SemEval-2018 dataset was utilized to evaluate the performance of the proposed methodology. Experimental findings validate the efficacy of the proposed model, which enhances the existing standards in classifying Arabic tweets, achieving an accuracy of 87.51% based on F-measures. The results indicate that the proposed analytical approach significantly advances text-based emotion detection and analysis, proving effective for Arabic text emotion analysis.

Keywords — : Emotion Analysis, Machine Learning, Arabic Language.

I. INTRODUCTION

The extensive volume of opinions and reviews posted online by social media users regarding policies, services, and products highlights the importance of understanding the vital information contained in social media content. This understanding is critical for a variety of stakeholder groups, including customers, business owners, and investors. Individuals use social media for many reasons, one of which is to express their opinions about products and political issues. This activity encourages various parties, such as consumers, businesses, and government entities, to participate in analyzing these opinions. Indeed, paying attention to customer feedback and reviews is a key tool in influencing decision-making processes. For organizations and individuals to improve their products and services, it is essential to uncover the range of sentiments conveyed and then use this information to formulate recommendations that are tailored to the unique needs of customers [1].

Emotion analysis (EA) is a subtask of natural language processing (NLP) that aims to analyze big data to discover people's opinions and emotions. Emotion analysis, or the detection of more complex feelings, is a relatively new field that presents new challenges in addition to those faced by sentiment analysis, where emotion analysis is a better classification [2-4]. Existing multi-label text-based emotion analysis Arabic datasets suffer from a high level of class imbalance. Where the number of cases in a certain class is very high, while in other classes the number of cases is low. In real life, the distribution of examples (training tweets) is biased since comments belonging to certain emotion classes rarely appear. This presents a difficulty for learning algorithms because they are biased towards the majority of classes. However, most text-based Arabic emotion analysis work assumes balanced sample sizes for each emotion class, which is not in accordance with reality[4-6]. The application of supervised learning in Arabic sentiment analysis is hampered by the insufficient datasets containing multi-label sentiment annotations, which are limited in size and imbalanced. Consequently, supervised learning techniques designed for balanced classification struggle to deliver desired results when faced with imbalanced data, negatively impacting the overall performance of sentiment analysis. Moreover, there has been a paucity of research addressing the challenges posed by imbalanced class distribution in the field of sentiment analysis [7-10]. There is also a lack of in-depth study on the impact of imbalanced classes in Arabic sentiment analysis. This work addresses the problem of class imbalance, which is one of the most difficult problems in multi-sentence analysis describing text-based Arabic. Moreover, sentiment analysis of Arabic is still in its infancy; in fact, researchers do not cover many dialects and there are few sentiment resources, which discourages field research from balancing the work done in other languages such as English[11]. The objective of this paper is to develop an improved model for Arabic text-based sentiment analysis with imbalanced datasets. In addition, to design an ensemble framework for heterogeneous learning models for Arabic text-based sentiment analysis. The rest of this paper is organized as follows. Section 2 presents related work while Section 3 presents the proposed methodology. The experimental setup is described in Section 4. Section 5 talks about the outcomes of the experiment. In Section 6, we finally wrap up our findings and explore potential avenues for further research.

II. Related Work

Emotion analysis can be conducted through various methodologies and has numerous applications. The primary techniques include lexicon-based, machine-learning-based, and deep learning-based approaches for emotion analysis. A comprehensive survey detailing research efforts in emotion analysis, the techniques employed, and the resources available are presented in [12]. Additionally, a systematic review focusing on the applications of natural language processing and the future challenges, particularly in text-based emotion detection, is discussed in [13]. The study in [14], examines several machine learning (ML) algorithms, including naive Bayes, support vector machines, and decision trees (DT), applied to sentiment analysis of airline review datasets. Furthermore, [15] offers a systematic review of machine-learning-based text classification methods. The research in [16] employs decision trees (DT), support vector machines (SVM), artificial neural networks (ANN), K-Nearest Neighbors (KNN), and Naïve Bayes (NB), along with ensemble models such as random forest (RF) and gradient boosting (GB), which utilize bagging and boosting techniques, as well as three sampling strategies based on ensemble hybrid sampling for addressing imbalanced data.

A study referenced in [17] proposed a method for classifying emotions in Arabic tweets utilizing a deep Convolutional Neural Network (CNN). This deep learning architecture functions as an end-to-end network, incorporating steps for word, sentence, and document vectorization. In [6], they used optimization BiLSTM network for multi-label Arabic emotion analysis and employed a CBOW word embedding model for word representation. A full survey about emotion detection in Arabic text in social media is introduced in [18]. The research presented in [19] uses bidirectional encoder representation by transformer models (BERT) for sentiment analysis and emotion recognition of Twitter data. In [20], they present an automatic text annotation methodology to label Arabic text data as multi-labels based on sets of extracted key phrases. They used to reduce the size of the features with the vector representation of the Bi-gram alphabet to build the document vectors. In [5], they present a model based on three state-of-the-art deep learning models. Two models are special types of recurrent neural networks RNN (Bi-LSTM and Bi-GRU), and the third model is a pre-formed linguistic model (PLM) based on BERT.

The methodology presented in [21] comprises three primary components: an embedding layer for word representation, a Bi-LSTM framework for capturing both forward and backward contextual information, and a sigmoid layer for classification aimed at emotion recognition, focusing on five core emotions: joy, sadness, fear, shame, and guilt. In [22], the authors introduced Enhanced Long Short-Term Memory (ELSTM) to identify emotions within Twitter data. The study in [23] employed LSTM, SVM, and nested LSTM techniques to classify multiple emotion labels successfully. In [24], they classified emotions into seven they are: (fear, anger, love, joy, surprise, thankfulness and sadness) using LSTM AND nested LSTM. In [25], they used method naïve Bayes, support vector machines, artificial neural network (ANN), and recurrent neural network (RNN). The study referenced in [26] implemented a multi-head attention mechanism combined with bidirectional long short-term memory and convolutional neural networks (MHA-BCNN). In [27], the researchers applied multiple methods for analyzing emotions in Arabic text, including bidirectional GRU_CNN (BiGRU_CNN), conventional neural networks (CNN), and an XGBoost regressor (XGB). They gathered a dataset of tweets by utilizing the Twitter API and conducting searches using emotion-related keywords. The findings displayed a Pearson coefficient of 69.2%.

In [28], they combine an attention-based LSTM-BiLSTM deep model with the transformer-based pre-trained Arabic Bidirectional Encoder Representations from the Transformers model (AraBERT) for Arabic language understanding to address the issue of Arabic affect detection (multi-label emotion categorization). The label-emotion of tweets is determined by the attention-based LSTM-BiLSTM, whereas AraBERT creates the contextualized embedding. Their suggested strategy performs better than the eight baseline techniques. It obtains a noteworthy accuracy rate of 53.82% on the SemEval2018-Task1.

III. Methodology

In this study, we employed a comprehensive methodology that encompasses all essential steps for the detection and analysis of emotions. The proposed approach consists of multiple phases, including preprocessing, management of unbalanced classes, feature selection, and emotion analysis, as illustrated in Figure 1. The primary objective of this paper is to develop an ensemble framework for analyzing emotions in imbalanced Arabic text.

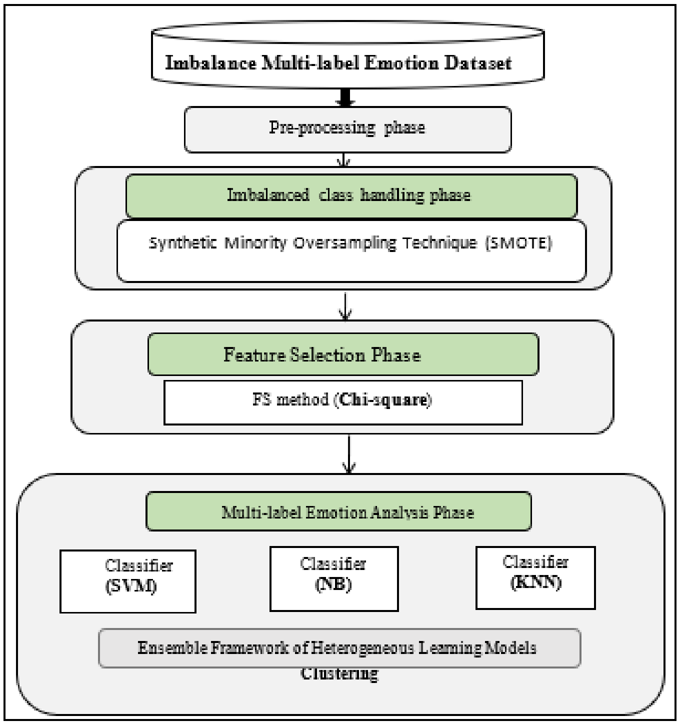
Imbalance Multi-label Emotion Dataset
Pre-processing phase
Imbalanced class handling phase
Synthetic Minority Oversampling Technique (SMOTE)
Feature Selection Phase
FS method (Chi-square)
Multi-label Emotion Analysis Phase
Classifier (SVM)
Classifier (NB)
Classifier (KNN)
Ensemble Framework of Heterogeneous Learning Models
Clustering

Figure. 1. The primary objective of this paper

The reviews and information collected from social media platforms and websites are inherently unstructured, similar to other forms of user-generated content, which complicates the analysis of sentiments. These datasets often contain errors such as misspellings, abbreviations, repeated characters, special symbols, and HTML tags. Therefore, it is essential to preprocess this data before any further analysis can take place. This preprocessing phase can also be regarded as a

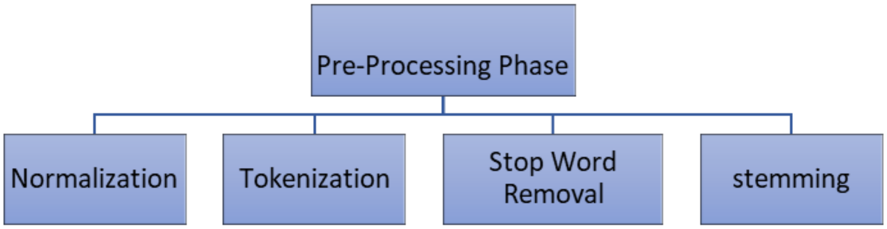

Figure. 1. The Pre-Processing Techniques.

dimensionality reduction step, as it standardizes different word forms and eliminates irrelevant stop words, including prepositions, conjunctions, and articles, which do not influence sentiment and are commonly present in reviews and opinion pieces.

Figure 2. displays the preprocessing methods utilized in this research encompassed tokenization, normalization, stop word removal, and stemming.

Normalization

Datasets on emotions that are gathered from social media sites are invariably unstructured and noisy. In social media, user-generated material is inherently casual, includes emoticons and emojis, and is frequently misspelled. Sometimes English words and special characters appear in Arabic evaluations. All HTML, links, and programming language code are deleted from the text along with certain English words and special characters. The second stage, normalization, transforms various Arabic word forms into a regular form. The many Arabic letters, such as " ا أ إ ء ئ ؤ " have led to the various forms of Arabic words. For instance, the word " احمد" can be written in a variety of ways, including "" أحمد "," إحمد . These variations may cause it to be mistakenly thought of as three distinct terms. They must all be changed into one of these representations as a result with uniformity of shape.

Tokenization

Tokenization stage is a vital step in any text mining process. Texts are divided into sentences, which are subsequently divided into lists of words or n-grams. A review is sent into the tokenization process, which transforms it into a representation in the form of a bag of words or bag of n-grams. The n-gram representation bigrams, trigrams, and unigrams is used in this work to express the meaning of valued emotions in phrases. To indicate the borders of words and sentences (or major tokens), punctuation marks and white spaces are utilized [29].

Stop Word Removal

In every language, stop words are the most prevalent and frequently nonsemantic expressions. Reviews contain stop words like determiners, prepositions, and pronouns, just as other types of writing. While certain stop words mostly negations are helpful in analyzing emotions, others are not. There is often a list of stop words in every language, such terms were eliminated since they are regularly seen in literature from all classes and do not contribute to class discrimination.

Stemming

This step is frequently called stemmers. A computational process known as stemming lowers all words that have the same root (or stem, if prefixes are omitted) to a common form. We used Root-based approach to do this. Typically, this is accomplished by depriving each word of it suffixes that are derivational and inflectional. By using a stemming method, the words are boiled down to their root. Here, our goal is to distill a word's various incarnations to its essential root or stem. This is useful in the field of information retrieval (IR) since it makes managing words with similar basic meanings more convenient. In information retrieval, matching documents to a query becomes more successful when terms with the same root (or stem) are grouped together. For the purposes of IR, a basic stemming of the English language that entails the removal of suffixes is enough. However, removing suffixes by alone would not be adequate for Arabic. Antefixes, prefixes, suffixes, and postfixes are the four types of affixes that can be added to words in Arabic[30] ,[31].

A. Handling Imbalanced Class

The process of balancing a dataset using SMOTE, which stands for Synthetic Minority Oversampling Technique, involves the generation of synthetic data points for the minority class to achieve equilibrium within the dataset. This is accomplished by augmenting the minority class data through the creation of new synthetic instances derived from the existing data. The methodology employs a KNN algorithm to facilitate the generation of these synthetic data points [32],[33]. In this study, the over-sampling strategy is implemented to ensure a balanced dataset. SMOTE effectively addresses the challenges posed by unbalanced datasets. This technique involves over-sampling the minority classes by generating synthetic samples that are based on the similarities in feature space among the existing minority instances. A vector is formed between the current data point and one of its KNNs, which is then multiplied by a randomly generated integer between 0 and 1 to produce a new synthetic data point. The steps involved in the SMOTE algorithm are outlined succinctly.

1- Identify the minority class vector.
2- Decide the number of nearest numbers (k), to consider.
3- Compute a line between the minority data points and any of its neighbors and place a synthetic point.
4- Repeat step 3 for all minority data points and their k neighbors, till the data is balanced.

B. Feature Selection Phase

The curse of dimensionality, wherein the number of created features is disproportionately large, is one of the most important issues with text mining jobs. Feature selection, also known as dimensionality reduction, is one of the most crucial stages in emotion analysis, during this process only discriminating characteristics are chosen. Many characteristics still lack discriminative power after the preprocessing and data representation stages. Only a small portion of the very large characteristics that were gathered include information that is useful for emotion analysis. An appropriate feature selection strategy that minimizes feature size is therefore required. The act of choosing the lowest subset of features to use in an analysis to minimize dimensionality while maintaining acceptable analysis performance is known as feature selection.

Filtering-based methods assign weights to features based on their effectiveness in differentiating between classes, as indicated by the data representation matrix. These weights help assess the degree of association between the features and the target class. A feature with a high weight indicates its potential utility in classification tasks. The features are maintained in a ranked list, organized in descending order of importance. Only the top n rated characteristics are chosen. The chi-squared statistic (x^2) is one of the most popular FS, it was utilized in this study and it is effective for emotion analysis. The x^2 statistic is one commonly used feature selection. Chi-square estimates whether the class label is independent of a feature. The chi-square score with class c and feature/word w is defined as:

$$\chi^2(c,w) = \frac{N \times (AD\text{-}BC)}{(A+C)(B+C)(A+B)(C+D)} \qquad (1)$$

where ***A*** is the number of times that ***w*** and c co-occur, ***B*** is the number of times that ***w*** occurs without ***c***, ***C*** is the number of times that c occurs without ***w***, ***D*** is the number of times that neither ***c*** nor ***w*** occurs, and ***N*** is the total number of reviews[34],[35].

C. Emotion Analysis/Classification

This section outlines the proposed ensemble learning model, which incorporates the classification algorithms SVM, NV, and KNN. The subsequent subsection provides a detailed description of these algorithms. The comprehensive ensemble framework is illustrated in Figure 2.

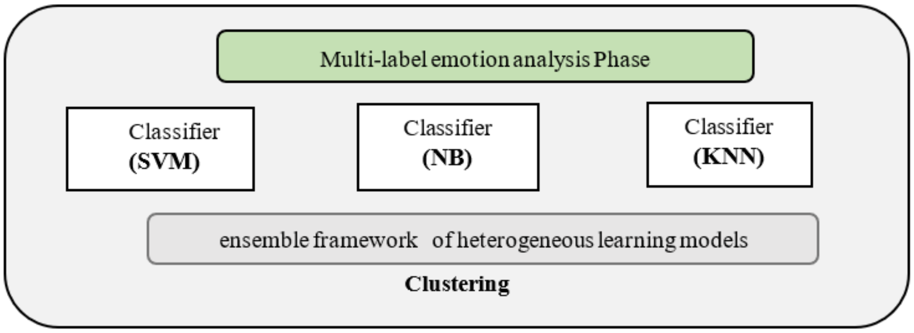
Multi-label emotion analysis Phase
Classifier (SVM)
Classifier (NB)
Classifier (KNN)
ensemble framework of heterogeneous learning models
Clustering

Figure. 2 . The General Architecture of the Proposed Ensemble.

Naive Bayes (NB)

The algorithm determines the posterior probability given a representation matrix and gives the review of the class with the largest posterior probability. The main benefit of NB algorithms is that, in many cases, they perform better and are simple to implement. The NB binary classifiers solve the emotion analysis problem given a review which is represented as a set of feature terms and is a class in the class set [36],[37]. Naive Bayes (NB) can be defined as the conditional probability of a given constructed as in Equation (2):

$$p(c|d) = p(c|t_1.\dots\dots.t_i) = p(c)\prod_i p(t_i|c) \qquad (1)$$

Equation Explanation (3):

P(c|d): This part represents the probability that the text (document) d belongs to the emotional category c. This is what we are trying to calculate.

The right part of the equation:

P(c): This is the prior probability of the emotional category c.

P(t_i|c): This is the conditional probability of a given word t_i (term) to appear in a text that belongs to the emotional category c.

∏(Pi): This symbol represents the repeated multiplication of all the word probabilities in the text.

Parameters explained in detail

c: represents the emotional category that we want to classify the text into (e.g. positive, negative, neutral).

d: Represents the text that we want to classify.

t_i: Represents the i-th word in the text.

P(c|d): As mentioned, this is the posterior probability that we want to calculate.

P(c): is the prior probability of the class, and can be calculated by dividing the number of texts in this class by the total number of texts in the data set. P(t_i|c): is the probability of the word t_i appearing in a text from class c, and can be calculated by dividing the number of times the word t_i appears in texts from class c by the total number of words in texts from class c.

Thus, the maximum posterior classifier is given in the following equation (3):

$$c^* = \underset{c\in C}{argmax}\, p(c)\prod_i p(t_i|c) \qquad (1)$$

Equation Explanation (3):

c*: is the class we want to define for the text, i.e. the class with the highest probability.

argmax: means the argument that gives the maximum value. That is, we are looking for the value of c that makes the expression on its right reach the largest value.

C: is the set of all possible classes.

P(c): is the prior probability of class c. That is, the probability that any random text belongs to this class.

t_i: is the i-th word in the text.

P(t_i|c): is the conditional probability of the word t_i appearing in a text of class c. That is, the probability that we find the word t_i in a text known to belong to class c.

Π: is a symbol for multiplicative sum, i.e. multiplying all values.

K-Nearest Neighbor (KNN)

One common example-based classifier is the K-nearest neighbor (KNN). Based on the similarity score, the search finds the K-nearest neighbors across all training reviews given a test review d [38]. The following equation (4) can be used to express the weighted sum in KNN categorization:

$$score(d, c_i) = \sum_{d_j = KNN(d)} sim(d, d_j)\, \delta(d_j, c_i) \qquad (2)$$

The details of the elements in the equation (4):

score(d, t_i): This function represents the degree or confidence of classifying text d into class t_i. In other words, it is a numerical value that expresses the extent to which we believe that text d belongs to class t_i.

d: represents the text we want to classify.

t_i: represents the class i that we want to classify the text into.

∑: the addition symbol, indicating that we are summing the function values for all the neighbors close to text d.

KNN(d): represents the set of neighbors closest to text d. K in this case is the number of closest neighbors we are considering.

sim(d, d_j): represents the degree of similarity between text d and its neighbor d_j. This degree can be calculated using various measures, such as similarity cosine or minority distance.

δ(d_j, c_i): is a delta function, taking the value 1 if neighbor d_j belongs to class c_i, and taking the value 0 otherwise. In other words, this function checks if the neighbor belongs to the same class that we want to classify the text into.

Support Vector Machine (SVM)

SVM is a powerful technique for solving problems in non-linear classification, function estimation and density estimation, which has led to many recent developments in kernel-based learning methods. Transforming a multi-label classification problem into a set of independent binary classification problems via the one-vs-all scheme is a conceptually simple and computationally efficient solution for multi-label classification. In this work, we conduct multi-label learning under such a mechanism by using standard support vector machines (SVMs) for the binary classification problems associated with each class. Given a labelled multi-label training set $D=\{(x_i,y_i)\}^N_{i=1}$ where xi is the input feature vector for the i-th instance, and its label vector y_i is a{+1,-1} valued vector with length K such as as $K = |Y|$. If $Y_{ik} = 1$, it indicates that the instance x_i is assigned to the *k*-th class; otherwise, the instance does not belong to the *k*-th class. For the *k*-th class (k = 1, · · ·, K) [36],[4].

IV. Experimental Setting

This section delineates the methodological approach employed to evaluate the effectiveness of text-based Arabic emotion detection and analysis models. Numerous experiments were conducted to assess both baseline and enhanced models. All experiments were carried out utilizing the SemEval-2018 dataset, which serves as the cornerstone of this research. The Arabic SemEval-2018 dataset is a corpus specifically curated for the SemEval (Semantic Evaluation) competition held in 2018. This dataset is sourced from Arabic text snippets extracted from Twitter, making it inherently rich in informal and colloquial language commonly found on social media platforms. The corpus is meticulously annotated with labels corresponding to eleven distinct emotions: anger, anticipation, disgust, fear, happiness, love, optimism, pessimism, sadness, surprise, and trust. Each text snippet within the dataset is categorized into one or more of these emotion categories based on the emotion expressed within the tweet. The dataset is partitioned in our experiments into two subsets: training, and testing, enabling researchers and practitioners to train, validate, and evaluate their models effectively. Table 1 provides an overview of the SemEval-2018 dataset.

Table 1. Description of SemEval-2018 Dataset.

Emotion	Training	Development	Testing
Anger	899	215	609
Anticipation	206	57	158
Disgust	433	106	316
fear	391	94	295
Happiness	605	179	393
Love	562	175	367
Optimism	561	169	344
pessimism	499	125	377
sadness	842	217	579
Supervise	47	13	38
Trust	120	36	77

To evaluate the proposed model, standard classification measurement precision, recall and F-measure are used Precision (Pi), Recall (Ri) and F-Measure (Fi) are mathematically defined shown in equations (5), (6) and (7).

$$P_i = \frac{TP_i}{TP_i + FP_i} \qquad (1)$$

$$R_i = \frac{TP_i}{TP_i + FN_i} \qquad (1)$$

$$F_i = \frac{2(P_i * R_i)}{P_i + R_i} \qquad (1)$$

TPi (True Positives): the number of cases that were correctly classified as belonging to class i.

FPi (False Positives): the number of cases that were incorrectly classified as belonging to class i, when they actually belong to another class.

FNi (False Negatives): the number of cases that were incorrectly classified as not belonging to class i, when they actually belong to it.

V. Result and Discussion

A. Individual Supervised Model Experiments

Multiple experiments were conducted to gauge the performance of individual supervised machine learning models, coupled with the Synthetic Minority Oversampling Technique (SMOTE) to address data imbalance. Initially, a series of tests are conducted to assess three basic text-based Arabic emotion analysis models: Support Vector Machine (SVM), K-nearest neighbors (KNN) classifier, and Naive Bayes (NB). Figure 3 shows the performance (F-measure) of the top outcomes from the fundamental models of text-based Arabic emotion analysis. The SemEval-2018 dataset was used for all of the tests. The main objective of this research is to examine how well standard machine learning performs when it comes to emotion identification and analysis on both balanced and unbalanced data using SMOTE. One can see that the K Nearest Neighbor (KNN) classification approach yields less accurate results than the Support Vector Machine (SVM) methods. The results also show that traditional machine learning working on a balanced dataset by SMOTE outperforms traditional machine learning working on an unbalanced dataset. It can be observed that the meta-ensemble model outperforms other basic classifiers. The meta-classifier combines the strength of its individuals (basic classifiers). He is waiting for her when many individual classifiers agree on the majority classification cases and do not agree only for small cases (when one of them is wrong), the combination of these classifiers always gets higher scores. In addition, the combination of the decisions of several unique classifiers that take a high score is better than the individual classifier (base classifier).

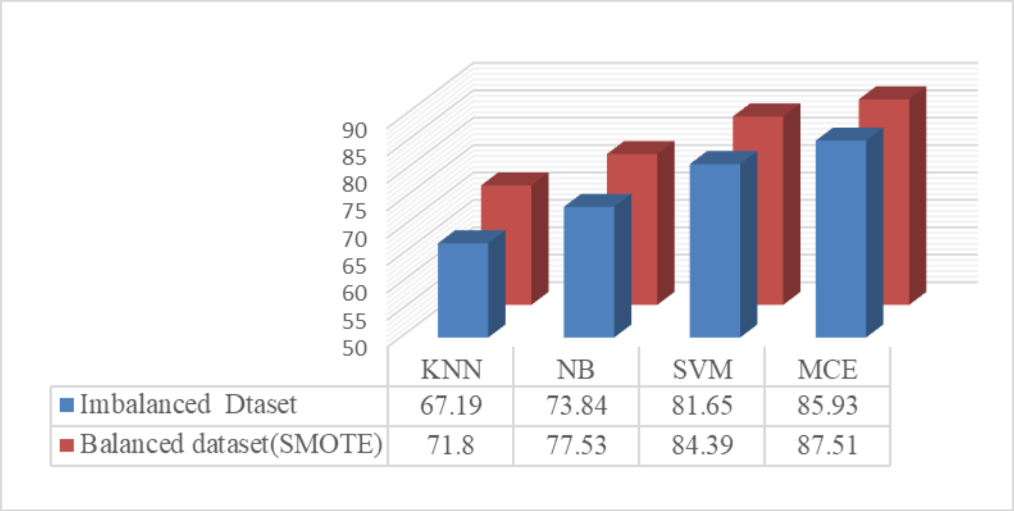

Figure. 3. Performance of Baseline Classification Models with Proposed Enhanced Methods for Text-Based Arabic Emotion Analysis on SemEval-2018 Corpus.

B. Ensemble Model Experiments

This study conducted various experiments to assess the effectiveness of ensembles of supervised learning models, also utilizing the SMOTE technique to tackle data imbalance. several experiments are conducted to evaluate the proposed meta-classifier ensemble learning model which combines a set of supervised learning models for Arabic text-based emotion analysis. This meta-classifier ensemble learning combines NB, KNN and SVM. Table 3, and Figure 4 show the results of the proposed ensemble method. Comparing these results with the performances of other classifiers in an isolated method. including Support Vector Machine (SVM), K-nearest neighbor (KNN), and Naive Bayes (NB), it becomes evident that the meta-classifier ensemble consistently achieves competitive or superior performance, particularly when SMOTE is employed. Across different feature sizes, MCE with SMOTE consistently demonstrates higher Precision, Recall, and F-measure values compared to other classifiers with SMOTE. These findings highlight the effectiveness of MCE as a robust classification approach for imbalanced datasets, especially when combined with SMOTE to address class imbalance.

Table 2. Performance of meta-classifier ensemble (MCE) with and without SMOTE on SemEval-2018 datasets.

	MCE without SMOTE			MCE with SMOTE		
Feature Size	Precision	Recall	F-measure	Precision	Recall	F-measure
250	78.46	77.18	77.81	80.84	79.84	80.34
500	82.28	81.69	81.98	84.22	83.04	83.63
750	78.42	76.25	77.32	79.97	77.51	78.72
1000	82.19	83.22	82.7	83.58	84.7	84.14
1250	82.93	82.99	82.96	84.7	84.89	84.79
1500	77.57	78.23	77.9	78.88	80.08	79.48
1750	84.6	84.94	84.77	85.96	86.46	86.21
2000	85.17	86.71	85.93	86.82	88.22	87.51
Avg	81.45	81.4	81.42	83.12	83.09	83.1

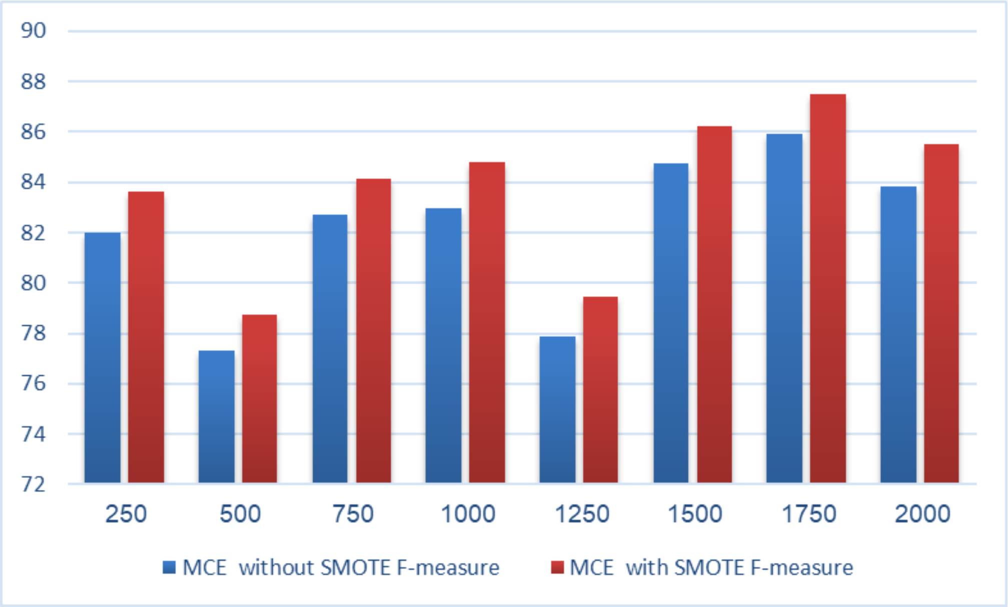

Figure. 4. Performance of Meta-Classifier Ensemble (MCE) with and Without the SMOTE on SemEval-2018 Dataset.

Overall, the best-performing classifier, both single and ensemble, was the meta-classifier ensemble (MCE), achieving the highest accuracy of 87.51% on the balanced dataset. This highlights the effectiveness of ensemble learning techniques in improving emotion classification accuracy for Arabic text-based datasets. Moreover, the results emphasize the importance of addressing class imbalance in emotion classification tasks, with SMOTE proving to be a valuable technique for enhancing the performance of classification models on imbalanced datasets. These findings contribute to the advancement of emotion analysis in Arabic text.

VI. Conclusion

This study conducts an empirical assessment of three foundational machine learning techniques: Support Vector Machine (SVM), K-Nearest Neighbors (KNN) classifier, and Naive Bayes (NB). Furthermore, it presents an Ensemble Framework specifically designed for Imbalanced Arabic Text-Based Emotion Analysis. The proposed approach demonstrates a commendable F-measure (F-score) of 87.51%, surpassing the performance of the basic methods. The results indicate that the proposed analytical method significantly enhances the detection and analysis of emotions in Arabic text. These outcomes suggest that the ensemble learning technique introduced is effective for the task of Text-based Arabic emotion detection and analysis. Future investigations should aim to develop emotion databases that encompass various Arabic dialects and slang, evaluate the proposed method across a range of datasets to confirm the model's efficacy and integrate advanced deep learning techniques to enhance ensemble learning strategies.

VII. References

[1] F. Greco and A. Polli, "Emotional Text Mining: Customer profiling in brand management," *International Journal of Information Management*, vol. 51, p. 101934, 2020.

[2] M. Arcan and P. Buitelaar, "MixedEmotions: Social Semantic Emotion Analysis for Innovative Multilingual Big Data Analytics Markets," in *Proceedings of the 18th Annual Conference of the European Association for Machine Translation*, 2015.

[3] A. Dvoynikova, O. Verkholyak, and A. Karpov, "Emotion Recognition and Sentiment Analysis of Extemporaneous Speech Transcriptions in Russian," Cham: Springer International Publishing, 2020.

[4] N. Alswaidan and M. E. B. Menai, "Hybrid Feature Model for Emotion Recognition in Arabic Text," *IEEE Access*, vol. 8, pp. 37843–37854, 2020.

[5] A. Mansy, S. Rady, and T. Gharib, "An Ensemble Deep Learning Approach for Emotion Detection in Arabic Tweets," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 4, 2022.

[6] E. A. H. Khalil, E. M. El Houby, and H. K. Mohamed, "Deep learning for emotion analysis in Arabic tweets," *Journal of Big Data*, vol. 8, no. 1, p. 1–15, 2021.

[7] S. V. Vora, R. G. Mehta, and S. K. Patel, "Impact of Balancing Techniques for Imbalanced Class Distribution on Twitter Data for Emotion Analysis: A Case Study," in *Data Preprocessing, Active Learning, and Cost Perceptive Approaches for Resolving Data Imbalance*. IGI Global, 2021, pp. 211–231.

[8] N. Jamal *et al.*, "A Deep Learning–based Approach for Emotions Classification in Big Corpus of Imbalanced Tweets," *Transactions on Asian and Low-Resource Language Information Processing*, vol. 20, no. 3, p. 1–16, 2021.

[9] J. L. S. Yan and H. R. Turtle, "Fine-grained Emotion Classification: Class Imbalance Effects on Classifier Performance," in *2021 International Conference on Computer & Information Sciences (ICCOINS)*, IEEE, 2021.

[10] L. Farsiah, Y.-S. Chen, and A. Misbullah, "Multi-Classes Emotion Detection for Unbalanced Indonesian Tweets," in *2020 International Conference on Electrical Engineering and Informatics (ICELTICs)*, IEEE, 2020.

[11] F. A. Q. Aljradi, M. Albared, and A. S. Ghareb, "Review On Using Machine Learning and Deep Learning Algorithms for Emotion Analysis," *المجلة العلمية-جامعة إقليم سبأ*, vol. 7, no. 1, 2024.

[12] H. A. Uymaz and S. K. Metin, "Vector based sentiment and emotion analysis from text: A survey," *Engineering Applications of Artificial Intelligence*, vol. 113, p. 104922, 2022.

[13] S. Kusal *et al.*, "A systematic review of applications of natural language processing and future challenges with special emphasis in text-based emotion detection," *Artificial Intelligence Review*, pp. 1–87, 2023.

[14] A. Patel, P. Oza, and S. Agrawal, "Sentiment Analysis of Customer Feedback and Reviews for Airline Services using Language Representation Model," *Procedia Computer Science*, vol. 218, pp. 2459–2467, 2023.

[15] A. Palanivinayagam, C. Z. El-Bayeh, and R. Damaševičius, "Twenty Years of Machine-Learning-Based Text Classification: A Systematic Review," *Algorithms*, vol. 16, no. 5, p. 236, 2023.

[16] N. H. A. Malek *et al.*, "Comparison of ensemble hybrid sampling with bagging and boosting machine learning approach for imbalanced data," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 29, pp. 598–608, 2023.

[17] M. Baali and N. Ghneim, "Emotion analysis of Arabic tweets using deep learning approach," *Journal of Big Data*, vol. 6, no. 1, pp. 1–12, 2019.

[18] Z. H. Ali and H. J. Aleqabie, "Emotion Detection in Arabic Text in Social Media: A Brief Survey," *Al-Furat Journal of Innovations in Electronics and Computer Engineering*, 2024, pp. 412–421.

[19] A. Chiorrini *et al.*, "Emotion and sentiment analysis of tweets using BERT," in *EDBT/ICDT Workshops*, 2021.

[20] F. Elghannam, "Multi-Label Annotation and Classification of Arabic Texts Based on Extracted Seed Keyphrases and Bi-Gram Alphabet Feed Forward Neural Networks Model," *ACM Transactions on Interactive Intelligent Systems*, 2022.

[21] M. Z. Asghar *et al.*, "A Deep Neural Network Model for the Detection and Classification of Emotions from Textual Content," *Complexity*, vol. 2022, 2022.

[22] S. S. BABOO and M. AMIRTHAPRIYA, "EMOTIONAL ANALYSIS OF TWITTER SOCIAL MEDIA DATA WITH AN EFFICIENT DEEP LEARNING MODEL," 2022.

[23] M. Karna, D. S. Juliet, and R. C. Joy, "Deep learning based text emotion recognition for chatbot applications," in *2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)*, IEEE, 2020.

[24] D. Haryadi and G. P. Kusuma, "Emotion detection in text using nested long short-term memory," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 10, no. 6, 2019.

[25] P. Mukherjee *et al.*, "Effect of negation in sentences on sentiment analysis and polarity detection," *Procedia Computer Science*, vol. 185, pp. 370–379, 2021.

[26] K. Dheeraj and T. Ramakrishnudu, "Negative emotions detection on online mental-health related patients texts using the deep learning with MHA-BCNN model," *Expert Systems with Applications*, vol. 182, p. 115265, 2021.

[27] O. AlZoubi, S. K. Tawalbeh, and A.-S. Mohammad, "Affect detection from Arabic tweets using ensemble and deep learning techniques," *Journal of King Saud University-Computer and Information Sciences*, 2020.

[28] H. Elfaik, "Combining Context-Aware Embeddings and an Attentional Deep Learning Model for Arabic Affect Analysis on Twitter," *IEEE Access*, vol. 9, pp. 111214–111230, 2021.

[29] R. M. Saeed, S. Rady, and T. F. Gharib, "Optimizing sentiment classification for Arabic opinion texts," *Cognitive Computation*, vol. 13, no. 1, pp. 164–178, 2021.

[30] M. Naili, A. H. Chaibi, and H. H. B. Ghezala, "Comparative study of Arabic stemming algorithms for topic identification," *Procedia Computer Science*, vol. 159, pp. 794–802, 2019.

[31] M. Mustafa *et al.*, "A comparative survey on Arabic stemming: approaches and challenges," *Intelligent Information Management*, vol. 9, no. 2, pp. 39–51, 2017.

[32] N. V. Chawla *et al.*, "SMOTE: synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.

[33] J. Brandt and E. Lanzén, "A comparative review of SMOTE and ADASYN in imbalanced data classification," 2021.

[34] S. L. Marie-Sainte and N. Alalyani, "Firefly algorithm based feature selection for Arabic text classification," *Journal of King Saud University-Computer and Information Sciences*, vol. 32, no. 3, pp. 320–328, 2020.

[35] F. A. Abdulghani and N. A. Abdullah, "A survey on Arabic text classification using deep and machine learning algorithms," *Iraqi Journal of Science*, 2022, pp. 409–419.

[36] M. Abdullah *et al.*, "Emotions extraction from Arabic tweets," *International Journal of Computers and Applications*, vol. 42, no. 7, pp. 661–675, 2020.

[37] H. Elfaik, "Social Arabic Emotion Analysis: A Comparative Study of Multiclass Classification Techniques," in *2021 Fifth International Conference On Intelligent Computing in Data Sciences (ICDS)*, IEEE, 2021.

[38] A. A. Sayed *et al.*, "Sentiment analysis for Arabic reviews using machine learning classification algorithms," in *2020 International Conference on Innovative Trends in Communication and Computer Engineering (ITCE)*, IEEE, 2020.