# Multilingual Sentiment Analysis to Support Business Decision-making via Machine learning models

Suher ElBasha[1], Amna Elhawil [2*] , Nabil Drawil[3]
[1] s.elbasha @uot.edu.ly, [2] a.elhawil @uot.edu.ly, [3] n.drawil @uot.edu.ly
[1] Department of Computer Engineering, Faculty of Engineering, University of Tripoli, Tripoli, Libya
*Corresponding author email: a.elhawil @uot.edu.ly

Sentiment analysis has become nowadays an important field in the business world because it retrieves the customers' opinions, emotions, and evaluations of a product. The written comments, collected from the social media, are a significant influencer on the marketing decision. The most significant challenges for sentiment analysis are that it is a language-dependent. Furthermore, optimizing classifiers for each language is very time- consuming and labor-intensive, especially for Neural Network (NN) models. It has been shown that the accuracy of the English sentiment analysis has higher accuracy than other languages, even for small datasets. Arabic sentiment analysis faces some challenges because Arabic language is unique in its structure. The same root word can have more than one form, depending on grammatical rules. However, large Arabic datasets are required in addition to intensive preprocessing techniques to get a respectable performance. This paper presents a new Arabic sentiment analysis model that is trained on an English sentiment analysis model. The idea is to develop a framework that benefits from languages with reach data-set in building a comprehensive model. A word embedding technique in addition to machine learning model approach are both implemented to translate, eliminate feature extraction and resource requirements for sentiment analysis. The developed model is trained such that it can be used by organizations to realize customers' attitudes about their products. The results are very motivating and the proposed model gives good performance.

**Keywords:** Arabic language, Multilingual Sentiment Analysis, Neural network, HuggingFace transformer

## Introduction

Sentiment analysis has been involved in different applications such as marketing finance, journalism, political science, etc. It recently became a big challenge for non-English languages. Researchers, therefore, opt to machine learning techniques in handling sentiment analysis, but such technique mandates a processing stage. In all sentiment analysis techniques text, which might be tweets or reviews, is separated into tokens. The tokens are either words or sentences. This is called tokenization. The tokens are then converted into vectors. These vectors are processed differently depending on the selected technique [1]. However, the tokenization is language processing. Even in the English language, there are some words that have ambiguous meanings. The exact meaning cannot be derived from a single word. In this case a word combination or multi-word tokenization is required. The most used techniques for tokenization include, but not limited to, splitting the text either by a specified separator or using rules that are specific for each language. The second method is to use text modelling similarity or matching the text to stored regular expressions [2]. In all cases, the performance of the tokenizer depends on the nature of the language. Furthermore, the commonly used programming languages such as Python offers libraries for different tokenizers. The most common library is Natural Language Toolkit (NLKT) [3]. The NLKT tokenizer supports many languages such as Dutch, French, Spanish, Turkish, Greek, Italian, Norwegian and Swedish. There is a scarcity in resources dealing with the Arabic language. Despite the fact that this language is spoken by roughly 422 million people [4]. The technique in [5] offers one solution to this problem. It employs a sentiment lexicon that is built by assigning a sentiment value or score to each word in a tweet. It functioned fine; however, it couldn't handle diverse dialects of Arabic. Another simpler solution is to use text translation. It basically takes the advantage of what is called resource-rich language such as English language tokenizers to support resource-poor or non-English text languages. That is done by translating the original text to English. Then process it as an English text. This appears to be a reasonable approach, but there is another issue: translation accuracy.

Recent research suggests an alternative method called multilingual sentiment analysis. It is applied on different non-English languages such as Chines, Indian, Swiss, and Russian languages [6-9]. Nonetheless, there is a lack of research in applying multilingual sentiment analysis for Arabic language. The purpose of this research is to find an answer to the following research question:

Is it possible to reuse a sentiment analysis classifier that has been trained on one language for sentiment analysis in other languages, even if the available data is limited؟ Other languages with limited resources could benefit from such an approach.

The remainder of this paper is organized as follows: Section 2 illustrates the sentiment analysis approach and lists its main steps. Section 3 gives an idea of the employed machine learning classifiers. In Section 4 we describe the obtained optimal setting parameters of the classifiers. Section 5 discusses the results and Section 6 presents the main conclusions and proposes the future work.

## 1. Sentiment analysis

Sentiment analysis is basically process of customer reviewers to determine opinion mining. Its goal is to determine the overall attitude (positive or negative). Positive means the customer likes the product or service while negative means the customer is not satisfied. The general procedure of the sentiment analysis has the following main steps:

1. The first step starts with collecting the appropriate datasets. All datasets must be binary classified (positive or negative).

2. In the pre-processing step, all symbols, numbers and emoticons are removed from the text. The emoticons removed because their Unicode codes effect both the tokenization and the translation processes.

3. Each tweet is split into individual words called tokens. Theses extracted words or tokens are stemmed where each word is replaced by its root.

4. The next step is to extract the features of each word and convert it to vectors.

5. The vectors are applied to the machine learning classifiers for classification.

6. Finally, the performance of the classifiers is evaluated.

In this paper, the proposed method for sentiment analysis is to train an English sentiment analysis model then apply a translated non-English, Arabic in our case, text to the model. The main purpose is to develop a general model that can analyse any non-English text and classify it with good accuracy.

The previous steps are applied to English text. For non-English text, after the pre-processing step, the text is translated to English before applying the tokenization and vectorization steps. The next subsections illustrate all these steps in details.

### 1.1 Data collection

Two corpora composed of collection of reviews and tweets, are used in this study, both are publicly available. The first corpus consists of English reviews, and the second corpus contains Arabic reviews. We focused on polarity detection in reviews/ tweets; therefore, all data sets in this study have two class values (positive and negative). These data sets are part of the SemEval-2016 Challenge Task 5 [10].

### 1.2 Pre-processing

To prepare the raw data for pretraining, we perform light pre-processing. This helps retain a faithful representation of the naturally occurring text. We only remove diacritics, user mentions, black spaces, URLs and hashtags that may exist.

### 1.3 Translation

In Natural Language Processing (NLP), there are many developed translation tools that translate from one language to another. The most common tool is Google Translate API tool. It is a fast and dynamic translation service developed by Google that supports 109 languages for text, media and speech [11]. This service can translate

multiple forms of text such as websites, documents, mobile applications and even images. Also Hugging Face Transformer is also one of the most powerful translation tools. It was trained using the MarianNMT framework and the Open Parallel Corpus (OPUS) dataset. This transformer comprises about 1000 language pairs, and 169 language family translations to English [12]. In this paper, Google Translate API tool and Hugging Face Transformer are both applied and their performance is compared.

### 1.4 Tokenization

After translating the text, it is split into words using a tokenizer. One of the top word tokenizers is NLTK Tokenizer. In addition to the Hugging Face transformer. Both are trained on a large collection of data and designed to be flexible and easy to use.

### 1.5 Feature extraction

The feature extraction step aims to assign the appropriate vector to a word. The vector can be considered as a weight assigned to the word. Higher weight means the word is more positive. The Word2Vec transforms words in each document into 300-dimensional dense vectors. Then computing an elementwise sum of the vectors divided by the number of terms in the text (simple averaging of vectors). As a result, we get 300-dimensional aggregated vector per text. It has been shown that Word2Vec gives high accuracy with Arabic, Chines, Spanish, French and Dutch languages [13]. For this mission we have used Google word2Vec model.

### 1.6 Classification

In this paper, four different machine learning classifiers are used. They are: Bidirectional Encoder Representations from Transformers (BERT), Random Forests (RF), Support Vector Machine (SVM) and Feedforward Neural Network (FFANN). The performance of these classifiers is evaluated and compared in order to determine the best model for such type of problem.

### 1.7 Evaluation

The respective model's evaluation is made using the de facto standard metric. The considered evaluation metric is the accuracy which measures the fraction of correctly classified patterns.

## 2. Machine learning Classifiers

In this section, the machine learning classifiers are briefly described.

### 2.1 Bidirectional Encoder Representations from Transformers (BERT)

(BERT) is an open-source machine learning framework for natural language processing (NLP). It is created and published in 2018 by Google. In this classifier, each output node is connected to each input element, and the weightings between them are dynamically calculated based on their connection [14].

### 2.2 Random Forests (RF)

Random forests are an ensemble learning method for both classification and regression tasks. It operates by constructing a multitude of decision trees at training time. For classification tasks, the output of the random forest is the class selected by most trees. For regression tasks, the mean or average prediction of the individual trees is returned.

### 2.3 Support Vector Machine (SVM)

SVM are supervised learning classifiers with associated learning algorithms that analyze data for classification and regression analysis. It is one of the most used classifiers for sentiment analysis [14-15].

### 2.4 Feedforward Neural Network (FFANN)

Multilayer perceptron (MLP) is implemented as a FFANN classifier for sentiment classification. In our implementation, the MLP which consists of multiple layers of neurons uses backward error propagation with Adaptive Moment estimation algorithm (Adam) for the training process. The *Word2Vec* features are fed to the input layer of

MLP. Hidden layers connect the input layer and the output layer by linear transformations and linear activation functions. The output layer is a fully connected layer which uses *softmax* function to predict the results of training examples. The *softmax* function is defined as:

$$S(y_i) = \frac{e^{y_i}}{\sum_{j=1}^{n} e^{y_j}} \quad \text{for } i = 1,2,\dots,n \tag{1}$$

where $y_i$ is values of the input vectors and *n* is the number of classes. Also, all hidden layers use the Rectified Linear Unit (ReLU) as an activation function defined as $f(x) = \max(0,x)$. The ReLU activation function is selected herein because it has been reported that it is capable of handling sparsity and reduced likelihood of vanishing gradient.

## 3. Methodology

The aim of this paper is to train the neural network model on English dataset. Once the good performance is achieved, a translated Arabic dataset is applied to the same model. The accuracy metric is used to evaluated the performance. The developed model is not limited to Arabic datasets, but it should work with any other language. Fig. 1 shows the block diagram of the model. As it can be seen, the training data is English whereas the test data could be in any language.

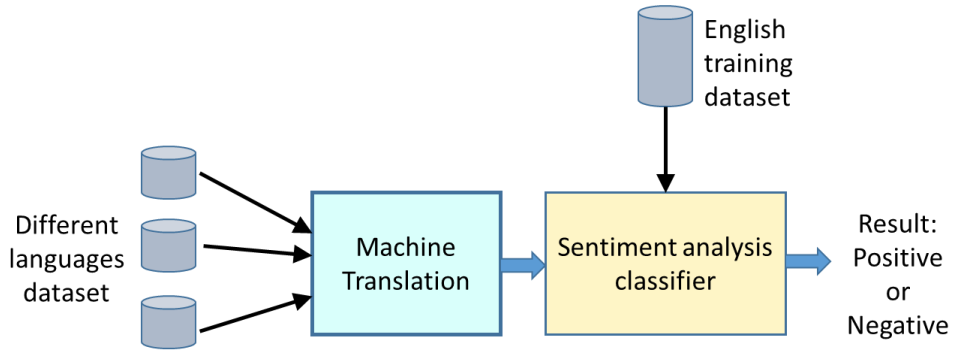


Figure 1. Block diagram of the multilingual model

The performance of machine learning classifiers highly depends on the configuration settings. Each classifier runs multiple times in order to determine the best values for hyperparameters. Here is the list the optimal setting of each classifier:

– **BERT:** by default, BERT has input 12 layers and 256 hidden layers. In this work, only two of the 256 hidden layers are used because they are enough to provide the required results with the efficient cost.

– **RF:** the number of trees in the forest is set to 100. In addition, the function to measure the quality of a split is *gini*. Supported criteria are "*gini*" for the Gini impurity and "*entropy*" for the information gain.

– **SVM***:* Poly kernel is used, which represents the similarity of vectors (training samples) in a feature space over polynomials of the original variables, allowing learning of non-linear classifiers. Also, the scale is set to Gamma.

– **FFANN:** The type of the neural network is Multi-layer perceptron. The size of input vector is set to 300 with 1 output vector. In addition to 3 hidden layers with 128 neurons in them. Moreover, the Activation function of the hidden layers is *ReLu* and for the output layer is *softmax* functions respectively.

## 4. Results and discussion

As we mentioned in the previous sections, the performance key of this paper is the translation of non-English texts. The more accurate translation is the better the sentiment analysis. Nevertheless, Arabic language is unique in its structure. The same root word in official language can have more than one meaning depending on grammatical rules. In addition, there is a great challenge in translating informal or dialect language used in verbal dialog and social media. Samples of Arabic reviews are listed in Table 1. These reviews are simple and closed to the official language. It is obvious that the translation of Google Translate API and HuggingFace transformer are both closed to each other.

Table 1**:** *Samples of the output of the translators*

| # | Arabic reviews | Google Translate API translation | HuggingFace transformer translation |
|---|---|---|---|
| 1 | شكرا جزيلا على التعامل الرائع والمصداقيه بالبضاعه المطلوبه' | 'Thank you very much for the great deal and credibility of the ordered goods' | 'Thank you very much for the wonderful handling and the credibility of the goods required' |
| 2 | البضاعه بتجنن وحسن التعامل وسرعة التوصيل الله يوفقكم' | 'The goods are crazy, good handling and fast delivery, may God help you' | 'The goods are insane, they're well handled, and the speed of delivery, God bless you.' |
| 3 | كل الشكر لكم استلمت الطلب وكتير ممتاز | 'Thank you very much. I received the order and it is very good' | 'All thanks to you, I've received the request and a great deal' |

Moreover, Table 2 lists some Arabic reviews or tweets that contain ambiguous words such as 'شعر' which could mean either 'poetry', 'hair' or 'feeling' depending on the sentence. Also, the word 'نخلص' which in Libyan dialect means 'I will pay for you' as in sentence 3 of Table 2. This is classified as positive. The same word 'نخلص' in the Egyptian dialect means 'I will kill you', as in sentence 4, which has negative classification. As shown in Table 2, HuggingFace transformer works better than Google translate API for these reviews. Many other examples have been performed and HuggingFace transformer shows a good performance. For this reason, it is considered in this paper as a translator and tokenizer.

Table 2: *Samples of the output of the used translators*

| # | Arabic tweets | Correct translation | Google Translate API translation | HuggingFace transformer |
|---|---|---|---|---|
| 1 | كتابة الشعر تحتاج الى موهبة | 'Writing the poetry requires talent' | 'Writing poetry takes talent' | 'Poetry writing needs talent.' |
| 2 | يتأثر نمو الشعر بعامل الوراثة | 'Hair growth is influenced by genetics' | 'Hair growth is influenced by genetics' | 'Hair growth is influenced by the genetic factor.' |
| 3 | نخلص عليك القهوة | 'We will pay for the coffee.' | 'We deliver you coffee' | 'We'll get you some coffee.' |
| 4 | نقدر نخلص عليك في اي لحظه | 'We can kill you at any moment' | 'We can solve you at any moment' | 'We can get rid of you any minute.' |

The datasets are divided into training and testing sets with 80% and 20% splits respectively. The size of the English dataset is 15,000 reviews, and the Arabic dataset is also about 15,000 reviews. Two main types of experiments are done. The first type is monolingual sentiment analysis. It consists of two implemented models:

1. English model: the neural network classifiers are trained and tested on English datasets. The results are shown in Fig. 2. The achieved average accuracy of this model is 87%.

2. Arabic model: in this model the neural network classifiers are trained and tested by Arabic datasets. The Arabic dataset is treated without translation. The achieved average accuracy of all classifiers is about 63%.

The second type experiment is multilingual sentiment analysis model. In this model, the neural network classifiers are first trained on English dataset, but the test is performed on Arabic dataset after translating it to English. The corresponding English text is processed and fed to the classifiers. The average performance has been improved to about 87%.

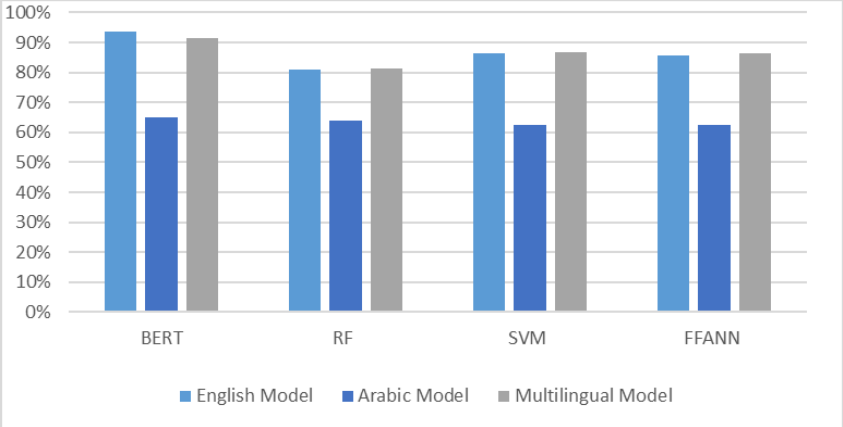


Figure 2. The achieved accuracy of the classifiers for monolingual and multilingual sentiment analysis

Table 3: *Accuracy results of the classifiers*

| | Classifiers | | | |
|---|---|---|---|---|
| | **BERT** | **RF** | **SVM** | **FFANN** |
| **English Model** | 94% | 81% | 86% | 86% |
| **Arabic Model** | 65% | 64% | 62% | 63% |
| **Multilingual Model** | 92% | 81% | 87% | 86% |

Fig. 2 and Table 3 are interesting in two ways. First, the accuracy of the multilingual model is too close to that of the English model. This superior result is mainly related

to translation step. It shows the success of choosing the appropriate translator which is the HuggingFace transformer. The accuracy of the Arabic sentiment analysis using the developed multilingual model is as same as that of the English sentiment analysis. Second, Fig. 2 shows that BERT classifier gives the highest accuracy which is about 94% for the English model and 92% for the multilingual model. SVM classifier comes next with an accuracy of 86% and 87% for the English model and the multilingual model, respectively.

## 5. Conclusions

Machine learning field is involved in the Business decisions. It supports buyer and consumer decision making process. One example is the topic discussed in this paper. The reviewers about the products or services can be identified and tracked using the proposed model. Not matter what language is used. The achieved accuracy of the Arabic sentiment analysis is 92% which is very good achievement.

## References

[1] D. S. Rajput, R. S. Thakur and, S. M. Basha, “Sentiment Analysis and Knowledge Discovery in Contemporary Business”, Advanced in Business Information Systems and Analytics (ABISA) Book series, IGI Global, pp. 18-20, 2018.

[2] D. Sarkar, “Text Analytics with Python: A Practical Real-World Approach to Gaining Actionable Insights from your Data”, Apress, pp. 218, 2016.

[3] Natural Language Toolkit, NLTK Project, 2021, https://www.nltk.org/

[4] Wikipedia, “List of countries where Arabic is an official language”, https://en.wikipedia.org/wiki/List_of_countries_where_Arabic_is_an_official_language, September, 2021

[5] M. Al-Ayyoub, S. B. Essa, “Lexicon-based sentiment analysis of Arabic tweets”, . Int J Soc Netw Min. 2015, vol. 2, pp. 101–14, 2015.

[6] Y. Xia, T. Zhao, J. Yao and P. Jin, “Measuring Chinese-English cross-lingual word similarity with HowNet and parallel corpus. In: Computational linguistics and intelligent text processing, 12th international conference, CICLing 2011, vol. 2. pp. 221–33, 2011.

[7] S. Ledalla and T. S. Mahalakshmi, "Multilingual Sentiment Analysis of Hinglish Tweets", Indian Journal of Public Health Research and Development, vol. 9, no. 12, pp. 1627, December, 2018.

[8] E. Pustulka-Hunt, T. Hanne, E. Blumer and M. Frieder, "Multilingual Sentiment Analysis for a Swiss Gig", 6th International Symposium on Computation AL and Business Intelligence (ISCBI). pp. 94–98, 2018.

[9] D. Bogoradnikova, O. Makhnytkina, A. Matveev, A.Zakharova and A. Akulov, "Multilingual Sentiment Analysis and Toxicity Detection for Text Messages in Russian", In Proceedings of the 2021 29th Conference of Open Innovations Association (FRUCT)، Tampere, Finland, pp. 12–14, May, 2021.

[10] M. Pontiki , D. Galanis, H. Papageorgiou, I. Androutsopoulos, S. Manandhar, M. AL-Smadi, M. Al-Ayyoub, Y. Zhao, B. Qin, O. Clercq, V. Hoste, M. Apidianaki, X. Tannier, N. Loukachevitch, E. Kotelnikov, N. Bel, S. M. Jiménez-Zafra, G. Eryiğit, "SemEval-2016 Task 5: Aspect Based Sentiment Analysis", Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016), 2016, DOI: 10.18653/v1/S16-1002.

[11] B. Turovsky. "Ten years of Google Translate". Google Translate Blog, April, 2016, Retrieved December, 2019.

[12] A. Ai, "Neural Machine Translation with Hugging Face's Transformers Library", January, 2021, https://anno-ai.medium.com/neural-machine-translation-with-hugging-faces-transformers-library-eb3bcce93298.

[13] S. Sagnika, A. Pattanaik, B. Shankar, P. Mishra and S. K. Meher , "A Review on Multi-Lingual Sentiment Analysis by Machine Learning Methods", Journal of Engineering Science And Technology Review, vol. 13, no. 2, pp. 154 – 166, April, 2020.

[14] S. Ravichandiran "Getting Started with Google BERT: Build and Train State-of-the-art Natural Language Processing Models Using BERT", Packt Publishing, 2021, pp. 78-90.

[15] K. Dashtipour, S. Poria, A. Hussain, E. Cambria, A.Y. A. Hawalah, A. Gelbukh and Q. Zhou," Multilingual Sentiment Analysis: State of the Art and Independent Comparison of Techniques", Cogn Comput (2016) 8:757–771, 2016.