[Arabic] Customer Review Classification

Abdalla Maged Gamal, Sherif Ehab Yousry, Abdelrahman Ahmed Badr, Abdulaziz Mustafa Amori

Ai, Nile University
Sheikh Zayed City, Giza
A.maged2102@nu.edu.eg
s.ehab2131@nu.edu.eg
A.ahmed2191@nu.edu.eg
a.amori@nu.edu.eg

***Abstract*— In this study, we use a preprocessing pipeline that includes tokenization, lemmatization, and the extraction of Arabic terms from the large dataset to investigate the nuances of customer review categorization applied to a corpus of 10,000 Arabic reviews. The course of our inquiry involves the use of two unique models: an 83.87% accurate BERT-based model and an 84% accurate TF-IDF model. We offer a thorough analysis of the results, going into detail on the experimental design, training procedures, and assessment criteria.**

***Keywords*— Natural Language Processing (NLP), Emotion Analysis, Neural Network, Arabic Customer Feedback, Sentiment Analysis, Amazon Egypt.**

I. Background

There are many strong and widespread fields in the field of artificial intelligence, but one of the most difficult things and the one in which the required progress has not been achieved yet is the field of automated natural language processing (NLP). There is some notable progress that has occurred in this field in the English language, but as for the Arabic language, the topic is very difficult due to the lack of Sufficient data in the Arabic language, through which some models can be trained so that we can use them in our own projects.

Therefore, the Arabic language does not receive enough attention in this field. It is difficult to obtain data in the Arabic language or it is almost impossible to do so, so you must work on your own to collect this data and make it available so that you can use it to build your own models, and this was the challenge in this project.

When we wanted to work on a model that would analyse the customer's review of the product and determine whether it was positive or negative, we found it difficult to find data specifically in the Arabic language so that we could train the model and obtain sufficient accuracy for this model. Therefore, training it is not the problem or the method that we will use in all libraries. It allows you to use advanced methods in the NLP part, but the obvious difficulty lies in the Arabic language and the insufficient data, and this is the gap in which we are working in this project, trying to reach an accuracy exceeding 80%, and trying to collect data approaching 10,000 customer reviews in the Arabic language.

II. Introduction

In this project, we will begin collecting data from the Amazon Egypt website by building a code in Python that will be able to extract customer reviews on products. Also, by the number of stars this review will get, we will be able to determine whether this review is positive or negative. For example, if it is more than three, it is considered positive, but vice versa, it will be considered negative. Through scrapping, we will be able to collect private data.

We will also use pre-processing on our data, and we will perform tokenization of the data and lemmatization until we get rid of stop words in the data and it becomes ready to enter the model appropriately and free of words that may hinder the model's understanding of the collected data.

Then we can work and start building our own model, which I think will be sufficient if we reach the goal, which is 10,000 data, that we can exceed 80% accuracy. Then we will use two methods in the model, one of which is pre-training the model from Hugging Face Company, and we will train it. On the data that we collected; this model is based on beer. On the other hand, we will use another model, which is the model known in machine learning as naive bayes, after encoding the Arabic data using TF-IDF. We will compare the results from the two models and determine which results are better. As we mentioned before, the goal of this experiment is to reach an accuracy exceeding 80%, given the lack of data. I think that 80% will be quite good.

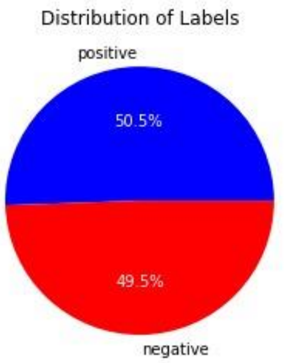
III. Methodology

1. Data Collecting (Scrapping):

First, I will talk about scrapping from Amazon Egypt. We have noticed that the basic structure of the HTML site is fixed for all products. They place the comments that come on the products from buyers in the same Tag, which contains a Class named (review:data-hook), and if another comment occurs, it is placed in the TAG exactly like it. We worked on extracting this comment or review from buyers and storing it in a CSV file so that we can deal with it easily in the next steps. There was another problem, which was whether this comment was positive or negative. We benefited from having the number of stars that each review received, and we were able to determine this number using SPAN which contains the Class 'a-icon-alt', which has the number of stars that this review received. Review by the client. We set a condition that if it is greater than three, it is considered positive, but if it is less, it is considered negative. This way we put a lot of links to different products and were able to get comments. We also used regular expression to filter and extract Arabic comments only, avoiding English ones, so as not to distract our model. Thus, the data was well assembled and ready to be worked on as preprocessing on 10,000 data set.

2. Preprocessing:

We pre-processed the data using the NLTK library, which created a WORD_TOKANIZATION for each sentence or comment contained in the previously collected data. Then remove stop_words from each sentence, then perform Lemmatization, then group the words again to get the sentence after processing it in the correct way. Thus, it will be ready to go directly to the next step, which is using it in the models that we will build. But before this, we made a DROP of the old column that contained the complete data (CONTENT) and left the new column that contains the pure data (PREPROCESSING), then we removed the Nulls from the data and divided the data into training and testing with a ratio of 30% data for testing and 70% data for training. Thus, you will be ready to enter directly into the models, train on them, and then test the accuracy that we will obtain after that.



3. MARBERT Model:

The first model we used is Marbert. It is considered one of the types of pre-trained models and belongs to Hugging Face Company, which is one of the largest companies in the field of NLP. This model was previously trained on data which is close to half a billion of data, and this is a very large number, and therefore this model is great for use on our data. We applied this model to the training data and then tested it on the test data. With the application of the AdamW optimizer and a learning rate set at 2e-5, our model embarks on a rigorous three-epoch training loop, dynamically adjusting its parameters with a linear scheduler. We noticed that the model worked efficiently, and the accuracy reached 83.87% after training on only three epochs. This was the result we expected from this model, which requires more data and therefore will give a better result.

4. Naive Bayes Model:

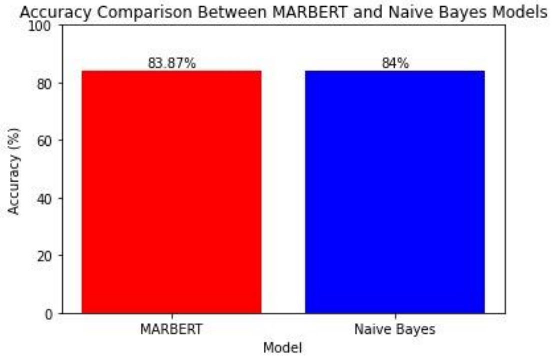
As for the second mode, we initially used Embedding of words using the TF-IDF library on the training and test data. Then we applied a model from the sklearn library, which is Naive Bayes. The result was better than the previous model, as the accuracy reached 84%. Also, this result was good compared to the small data that was used.

IV. PRACTICAL APPLICATIONS

The benefit that will be obtained by anyone who benefits from this model is that if he is able to collect more data, it will be an achievement, and therefore using the model for this project can be relied upon on private purchasing sites in a very wide and reliable manner. He can distinguish between comments or reviews of products, whether they are negative or positive. Therefore, he can delete the product if it gets a lot of negative reviews, and its results are widely adopted by these companies or sites specialized in buying or selling products online. This will save a lot of money and time for these large companies that receive a very large number of products and purchases daily, and using this model will greatly save and improve the performance of the products available there.

V. RESULTS

When comparing the two models with each other, we will notice that the results are very close and that the problem is not in the model, but in the small data, and because they obtained 84% and the other obtained 83.87%, these percentages are very close to each other, and therefore the problem is not in choosing the model, even if we were able to obtain a larger data. On a larger and different scale other than Amazon Egypt, for example Noon and OLX and many of these different sites and different products on them, the model can achieve much higher success and much greater accuracy, which may reach 90% or even 95%, and it will become an excellent model if it obtains sufficient data.



VI. Discussions

There are not many papers and research that talk about this project in the Arabic language, but most of the existing projects face the same problem, which is the lack of sufficient data to implement a model that works with very high efficiency (95% - 99%) and can be adopted in large places naturally and obtain guaranteed results from it in a way.

Future work:

As for future work, we hope that Arabic data, like English data, will be available by companies, as they will benefit greatly from this model and will use this model in developing their company. Therefore, the benefit will accrue to everyone if they are able to provide this data in Arabic, and engineers in the field of NLP work on This data and they developed a model that can reach very high accuracy. This will greatly benefit everyone and will make the Arabic language very advanced in this field and we will be able to develop and evolve in the future.

VII. Conclusion

In the end, data processing on NLP is weak in the Arabic language, as we have highlighted the lack of many data that can be used and applied to the huge models that exist now. Therefore, in this paper, only 10,000 pieces of data are used to classify customer comments whether they are positive or negative. This is a small number, and even the results from the two models (BERT 83.87% - Naive 84%) were quite reasonable Compared to the amount of data. At the beginning of this research, we set a goal of reaching 80% accuracy, and we have already exceeded this accuracy during this project, in which we started by scraping data, then preprocessing the data, using NLTK, then using MARBERT and Naive Bayes models. The final performance was better than the percentage we set at the beginning, and it exceeded 4% of this previously set percentage. Then we discussed in these practical applications that can be used on this model, which are specific to online purchasing sites, and if they are able to provide the data they have in the Arabic language, the accuracy of the model can reach 90 to 95%, and this will be enough so that they can use it on their sites. Their companies will increase significantly, and this will save a lot of time and money and will benefit everyone.

Finally, we talked about the scarcity of existing programming research in the Arabic language and the extent of the difficulty and challenge that exists in starting this project. We conclude the research paper with future works that hope companies will provide their data in the Arabic language so that we can reach a much better result and advance NLP and enhance development soon.

References

[1] https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html

[2] https://medium.com/@cmukesh8688/tf-idf-vectorizer-scikit-learn-dbc0244a911a

[3] https://www.analyticsvidhya.com/blog/2021/11/how-sklearns-tfidfvectorizer-calculates-tf-idf-values/

[4] https://kavita-ganesan.com/tfidftransformer-tfidfvectorizer-usage-differences/

[5] https://huggingface.co/UBC-NLP/MARBERT

[6] https://huggingface.co/UBC-NLP/MARBERTv2

[7] https://metatext.io/models/ubc-nlp-marbert