Sentiment Analysis and Classification

**Nuwayer Algoliqah**

Hood College

Frederick, MD

[na16@hood.edu](mailto:na16@hood.edu)

ABSTRACT

Sentiment analysis or opinion detection is the computation study of what people express in writing or saying, attitudes, and emotions in a context. It is one of the most active research areas in natural language processing and text mining in recent years as increasing the numbers of applications we used. There are mainly two main reasons. First, the wide range of applications we used because the people reviews are central to almost all human activities. Whenever we need to make a decision, we want to read their review or feedback. Second, it presents many challenging research problems

We applied text mining methods and algorithms and we compared the achieved accuracy for each after tuning or applying optimization methods aim to increase the validation accuracy we faced some of difficulties to achieve best accuracy on the validation curve but still there some obstacles and we know the core of text processing especially sentiment analysis is sentence transformation ore sentence vectorization  we need to transform our documents into vector representations such that we can apply numeric machine learning. This process is called feature extraction or more simply, vectorization The simplest encoding of semantic space is the bag-of-words model BOW just creates a set of vectors containing the count of word occurrences in the document (reviews), while the TF-IDF model contains information on the more important words and the less important ones as well.so in this work we used the last in sentence transformation before applying ML models and finally we applied deep learning LSTM model we found the best result obtained from this model

**KEYWORDS:**

Test Mining Techniques, Classification, SVM, Classiﬁcation and, Decision Trees, Sentiment Analysis, Test Analysis, NLP, Text Processing, Word Embedding

# **INTRODUCTION**

Sentiment Analysis (SA) is an ongoing field of research in text mining field. SA is the computational treatment of opinions, sentiments and subjectivity of text.

There are also many title and slightly different tasks, e.g., sentiment analysis, opinion mining, opinion extraction, sentiment mining, opinion detection, subjectivity analysis, affect analysis, emotion analysis, review mining, etc. the both sentiment analysis and opinion detection are frequently employed. They basically represent the same field of study. The term sentiment analysis perhaps first appeared in (Nasukawa and Yi,2003), and the term opinion mining first appeared in (Dave, Lawrence and Pennock, 2003). [1] The two expressions SA or OM are interchangeable. They express a mutual meaning. However, some researchers

stated that OM and SA have slightly different notions

Opinion Mining extracts and analyzes people’s opinion about an entity while Sentiment Analysis identifies the sentiment expressed in a text then analyzes it. Therefore, the target of SA is to find opinions, identify the sentiments they express, and then classify their polarity as shown in Fig. 1 [2]



Fig.1 Sentiment analysis process on product reviews.

# **LITRETURE REVIEW AND DATA SET**

Studies on sentiment analysis mainly focus on framework and lexicon construction, feature extraction, and polarity determination. This paper presents a survey on the latest development in sentiment analysis and apply LSTM starting from text cleaning and preprocessing under the Natural Language Processing (NLP) methods to build deep learning model and prediction, some of the existing challenges are due to the slang words, new accents, grammatical and spelling mistakes The area of sentiment analysis has become so large that any individual researcher would face several issues when keeping track of all the activities in the area and the information overload. An academic literature review can only focus on one particular area of sentiment analysis as it typically includes between 10 to 100 studies, e.g., a recent systematic review of the prediction of financial markets with sentiment analysis reviewed 24 papers [3]. Our dataset used is Sentiment Labelled Sentences released on [Data Sets - UCI Machine Learning Repository. It contains two classes of sentences with positive or negative sentiment labeled by1 and 0 respectively. The sentences come from three different websites/fields: imdb.com, amazon.com and yelp.com. For each website, there exist 500 positive and 500 negative sentences and in total 1386 sentences are positive and 1362 are negative. Those were selected randomly for larger datasets of reviews. We attempted to select sentences that have a positive or negative review or opinion. [4]](https://archive.ics.uci.edu/ml/datasets.php)

# **METHODOLOGY**

In this work, we will go through the steps cleaning and preprocessing for text and convert each sentence into vector we applied here TF-IDF algorithm to covert sentence in dense vector. TF-IDF stands for (term frequency-inverse document frequency) its working as in detecting the relevance of key-words to documents in the context. TF-IDF is a numerical statistic that shows the relevance of keywords to some specific documents or it can be said that, it provides those keywords, using which some specific documents can be identified or categorized. For example, a blogger is running a blog with hundreds of contributors and he just hired an internee whose main task is to add new blog posts on daily basis. It has been observed that most of the times internees does not take care of tags due to which many blog posts are not categorized. This is one of the ideal condition for applying TF-IDF algorithm, which can identify the tags automatically for the bloggers. [5]

As we mentioned before we will go through the steps we followed in text sentence preprocessing and introduce the similar method can be used in this area the first step in text mining is Tokenization is the process of breaking a stream of text into words, phrases, symbols, or other meaningful elements called tokens. The aim of the tokenization is the exploration of the words in a sentence. For further processing such as parsing or text mining the list of tokens becomes input. Stop words are very common words like ‘the’, ’he’, she’, ‘and’, ‘is’, ‘this’ etc. They are not useful in classification of documents. So they must be removed.

One another common text preprocessing technique is to remove the punctuations from the text data. This is again a text standardization process that will help to treat 'hurray' and 'hurray!' in the same way like that.

Stemming is the process of conflating the variant forms of a word into a common representation

For example, the words: “presentation”, “presented”, “presenting” could all be reduced to a common representation “present” [6]. In terms of word embedding are widely applied in information retrieval [7] recommendation systems [8], image description [9], One of the interesting trends in natural language processing is the use of word embedding. The aim of this latter is to build a low dimensional vector representation of word from a corpus of text. The main advantage of word embedding is that it allows to offer a more expressive and efficient representation by maintaining the contextual similarity of words and by building a low dimensional vectors. Recently, the two well-known methods for producing word embedding models are Word2Vec and Global Vectors GloVe. These two methods have been drawing great attention and it has been reported to be the most efficient ones for learning vector representations of words1, 2. For this reason, Word2Vec and Glove have been used in different natural language processing tasks such as Word Similarity. However, it is difficult to choose one of these two methods. In realty, Pennington et al. proved that GloVe is more efficient than Word2Vec. [10] Furthermore, they proved that classical methods can be more useful than Word2Vec in particular the Latent Semantic Analysis (LSA). This technique is considered as one of the most influential early models for word embedding. [11]

# **Data Mining Model**

Our problem here is a classification problem and after applying clean and preprocessing methods in addition word embedding method and here we applied TF-IDF on our text and then we tried to train more machine learning estimator to compare its performance after applying hyper parameter tuning for each estimator (SVM, Logistic Regression, Decision Tree, KNN, Bayesian Classifier):

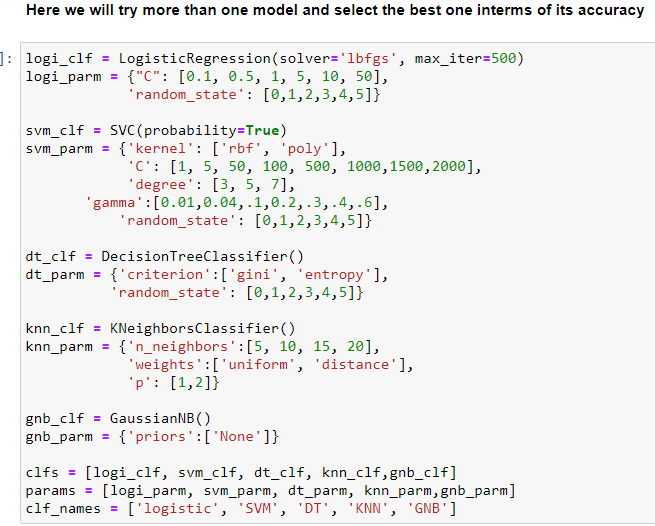
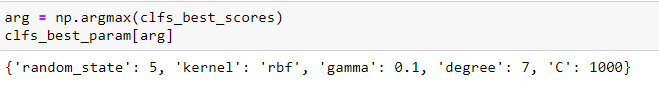


Fig2: machine learning classifier with their parameter range

And we found the best accuracy obtained on test data was from SVM Classifier by 61 percentage on parameter:-

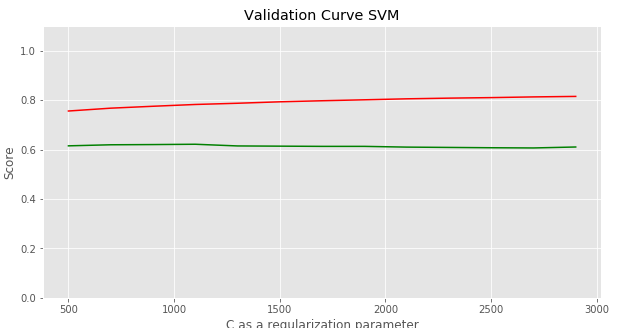
And the validation curve for a list of range for C as regularization parameter we here labeled the trading curve in red color and test in green color the accuracy achieved as 81 and 61 percentages for training and test data respectively. 

Fig3: validation curve between training& testing data using SVM

After as we note the accuracy is low and the large distance between the trained and test on validation curve so we have to find another method to apply to solve such problem so we applied deep learning algorithms aim to achieve or increase the current accuracy we applied Long Short-Term Memory (LSTM) A special variation of RNN, Long Short Term Memory (LSTM) networks is discussed. LSTM showed a striking accuracy in language modeling and speech recognition. We will be varying different forms of LSTM for our text classification purpose. A LSTM network contains LSTM units along with the input and output network layer units. [12] In addition we applied LSTM Network with simple architecture with only 2 layer we achieved 98 percentage accuracy in training #####after 20 epoch

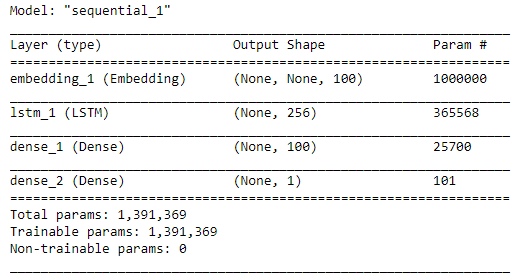
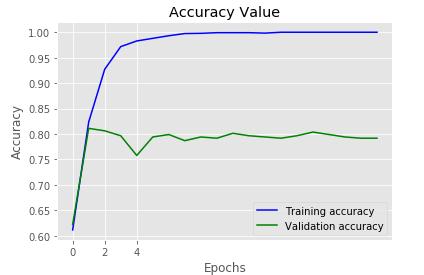


Fig4: Our LSTM Model Architecture

Keras offers an [Embedding](https://keras.io/layers/embeddings/#embedding) layer that can be used for neural networks on text data. Normally the embedding matrix is ​​learned during the same model learning, to adapt the best vectors for each object. [13] and we achieved 79 percentage accuracy on test data.

 fig5: validation curve in terms of accuracy between training &validating data

# **Data and work flow Challenges:**

Here we worked in text processing and text usually winds up having very bad accuracy/F1 measure the main challenges we faced here to detect a more in depth sentiment/emotion meaning is how to extract emotions like how much hate there is inside the opinion, how much happiness, how much sadness.in addition how to detect the object that the opinion is positive for and the object that the opinion is negative in same time For example, if you say "She won him!", this means a positive sentiment for her and a negative sentiment for him, at the same time. Also Sentiment Towards Aspects of an Entity A review of a product or service can express sentiment towards various aspects. For example,

a restaurant review can speak positively about the service, but express a negative attitude towards the food. There is now a growing amount of work in detecting aspects of products in text and also in determining sentiment towards these aspects. And Sentiment in Figurative Expressions their meaning cannot fully be derived from the meaning of their components in isolation. Finally challenges in applying sentiment analysis they can accurately capture significant changes in the proportion of instances that are positive (or negative). It is also worth noting that such sentiment tracking systems are more effective when incorporating carefully chosen baselines. For example, knowing the percentage of tweets that are negative towards Russian President, Vladimir Putin, is less useful than, for instance, knowing: the percentage of tweets that are negative towards Putin before vs. after the invasion of Crimea; or, the percentage of tweets that are negative towards Putin in Russia vs. the rest of the world; or, the percentage of tweets negative towards Putin vs. Barack Obama (US president). [14]

# **CONCLUSION**

Here we demonstrate the motivation of studying sentiment analysis and its application in the marketing specially we also pursue through the text cleaning and processing methods towards applying machine learning algorithms and we compare the result obtained by applying deep learning LSTM model and we note there are significate improvement happened in terms of the accuracy on validation data. On this work for machine learning estimators we only applied one method in the area of word embedding TF-IDF by achieving 61 percentage accuracy with best estimators and parameters founded in support vector machine SVM this value may be increased if we applying another method in terms of text transformation or vectorizations techniques. And then applied LSTM model as a deep learning estimator by achieving 80 percentage accuracy on validation data.