Report for Sentiment Analysis

Submitted as part of the requirements for:

CE807 Text Analytics

NAME: ZhengKun Yao

STUDENTMENT NUMBER:1807659

Professor: Ansgar Scherp

DATE: 22 Apr 2019

CLASS NUMBER: CE807

WORD:3130

**Abstract**

Sentiment Analysis is a branch of Natural Language Processing(NLP), which is used to analyses the subjective text with emotional colour(derogatory meaning or positive or negative direction) to determine the viewpoint, preference, and emotional tendency of the text. In this report, I will introduce what I did for this assessment. My project includes four part: Literature Review, Write the code for the training model, Evaluate the performance and write the report. About the code, I split it into three parts: Load the data file, Pre-processing the data, training the model and predicting & save result.

Keywords: Keras, Python, Scikit-Learn, LSTM, Machine Learning, Deep Learning

Content

[1.Introduction 3](#_Toc6888155)

[2.Literature Review 4](#_Toc6888156)

[2.1.Sentiment Dictionary 4](#_Toc6888157)

[2.2. Machine Learning 4](#_Toc6888158)

[2.3 Naïve Bayes Classifier 4](#_Toc6888159)

[2.4 K-NN Classifier 5](#_Toc6888160)

[2.5 Support Vector Classifier 5](#_Toc6888161)

[2.6 LSTM Neural Network 5](#_Toc6888162)

[2.7 Rotten Tomatoes Dataset 6](#_Toc6888163)

[3.Decide the Algorithm to use 6](#_Toc6888164)

[3.1 Naïve Bayes 6](#_Toc6888165)

[3.2 K-NN Classifier 6](#_Toc6888166)

[3.3 Support Vector Classifier 6](#_Toc6888167)

[3.4 LSTM Neural Network 7](#_Toc6888168)

[4.Implement the Object 7](#_Toc6888169)

[4.1Data loading 7](#_Toc6888170)

[4.2Feature Extraction 10](#_Toc6888171)

[4.3Training Model and Evaluation 11](#_Toc6888172)

[5.Comparison with the state of the art 13](#_Toc6888173)

[6.Lessons Learned 13](#_Toc6888174)

[7.Conclusion 14](#_Toc6888175)

# 1.Introduction

Sentiment Analysis is a branch of Natural Language Processing(NLP),which is a process of analysing, processing, summarizing and reasoning subjective texts with emotional colour. Sentiment Analysis is always used on Movie Review, Product Comment, Public Sentiment, Politics and Prediction.

Now, there are two mainstream methods of sentiment analysis based on sentiment dictionary and sentiment analysis based on machine learning. Based on sentiment dictionary, it according to the constructed sentiment dictionary to calculate the emotional tendency of the text by processing the text of extracting emotional words. The final classification effect depends on the perfection of the sentiment dictionary.

Based on machine learning, it means extract the emotional words as features, matrix the text and use different machine algorithms to classify the text such as Logistic Regression, Naïve Bayes and SVC. Furthermore, deep learning is also suitable on sentiment analysis. For example, we can use LSTM network, CNN network, AE and so on to classify the text. The final classification effect depends on the choice of training text, the processed data and the correct emotional labelling.

In this report, I will introduce what I did. Such as the first one is Literature review which I split it into three parts: “Sentiment Dictionary, “Machine Learning Algorithm ” , “LSTM Neural Network” and Rotten Tomatoes datasets. The second one is Decide the Algorithm to use. The third one is Data Preprocessing and I will introduce how I processed the datasets. The forth part is to implement the object, about this part, I split it into three parts: “Algorithm implement and Network build” and “Training Model and Evaluating”. The final one is the conclusion.

# 2.Literature Review

In this section, I will introduce two mainstream methods of sentiment analysis which are sentiment dictionary and machine learning. After machine learning part, I will introduce more specific about the algorithm of machine learning and how deep learning use in this case .

## 2.1.Sentiment Dictionary

The method based on sentiment dictionary for sentiment analysis is the traditional method for sentiment analysis. It according to the constructed sentiment dictionary to calculate the emotional tendency of the text by processing the text of extracting emotional words. Furthermore, the final classification effect depends on the perfection of the sentiment dictionary.[1]

Dictionary-based sentiment classification is simple and easy, and versatility can be guaranteed. But there are still many shortcomings: low accuracy, Language is a highly complex thing, and using a simple linear overlay obviously causes a large loss of precision. Word weights are also not static, and it is difficult to be accurate; new words appeared, for this shortcoming, for example, if there are new emotional words in a text not included in the sentiment dictionary, the result will be wrong; difficult to build a dictionary, the core of dictionary-based sentiment classification lies in the emotional dictionary. The construction of emotional dictionary requires strong background knowledge and requires a deep understanding of language.

## 2.2. Machine Learning

The essence of sentiment analysis is classification process. Machine learning is very suitable for classification, machine learning method use learning algorithm and classifier model to train the dataset[2]. Basically, the machine learning classifier need vector import, so we need to preprocessing the text data first and extract the sentiment words as features. The next step is transform the words as word vector or matrix and put it into classifier such as Naïve Bayes classifier, Support Vector Classifier, Logistic Classifier and K-NN classifier. Besides, the final classification effect depends on the choice of training text and the correct emotional labeling.

## 2.3 Naïve Bayes Classifier

Naive Bayes classifiers are simple, probabilistic classifiers based on Bayes’ Theorem. In addition, it is a generative model, which uses a method of directly modeling the joint probability P(x, c) to obtain the target probability value. [3] For all Naïve Bayes classifier, when Naïve Bayes classifier is given the class variable, it assume that the particular feature’s value is independent of others.

The limited of Naïve Bayes is that in theory, the naive Bayesian model has the smallest error rate compared to other classification methods. But this is not always the case. This is because the naive Bayesian model assumes that the attributes are independent of each other. This assumption is often not true in practical applications. When there are many attributes or the correlation between attributes is large, Bad classification.[5]

## 2.4 K-NN Classifier

K-nearest neighbor (K-NN) is a learning algorithm based on distant which can be used for classification and regression. The point or vector which need to predict is depend on the neighbor vectors. In a limited range, the predict result is same with the most neighbor vectors, in other words, it is the same with the most nearest one. This technique can also use the weight to optimize the algorithm, for example, the nearer neighbors have more weight than the distant ones. [4]

But the K-NN algorithm is limited. The high computational complexity and high spatial complexity are the disadvantage of it. Besides, the distance value must be calculated for each data in the dataset we use, it will cost a lots of time.

## 2.5 Support Vector Classifier

Support-vector classifier (SVC) is a based on Support Vector Machine(SVM) which is used for classification. It is one of the most popular classifiers in the world and most widely use. It improve the generalization ability of learning machines by seeking the minimum structural risk, and minimize the empirical risk and confidence range, so as to achieve good statistical rules in the case of less statistical sample size. The core concept of Support-vector machine (SVM) is to divide the data into two parts in a hyperplane and classify the data by using -1, 0, 1.

This technique solves the problem of requiring an infinite number of samples in the past. It only needs to abstract a certain amount of text into vectorized training text data, which improves the accuracy of classification. [6]

The main problem of SVM is only in binary classification has the best application performance, and other prediction problems are not very good.

## 2.6 LSTM Neural Network

LSTM (Long Short-Term Memory) is a long-term and short-term memory network. It is a time recurrent neural network suitable for processing and predicting important events with relatively long intervals and delays in time series.[7] In contrast to the RNN model, all RNNs have a chained form of a repetitive neural network module.

The main advantage of LSTM on sentiment analysis is LSTM can capture the changing sentiment in a tweet.[8] For example, when a sentence contain conflicting sentiments words which is difficult to be inferred accurately by others neural network. LSTM can do this by learning the sentiments expressed towards the end of the sentence would be more important context compared with the words which at the start.

## 2.7 Rotten Tomatoes Dataset

The Rotten Tomatoes Dataset is the data set of movie review from Kaggle competition which public for sentiment analysis research and study. For every movie review in the dataset, it has five ranks: 0,1,2,3,4 to replace Negative, Somewhat Negative, Neutral, Somewhat positive and Positive.

# 3.Decide the Algorithm to use

## 3.1 Naïve Bayes

The limitation of Naïve Bayes is that it is more suitable for small scale dataset and in this case the dataset is too large and cannot ensure the accuracy of prediction.

## 3.2 K-NN Classifier

Although K-NN is easier to understand and implement, but it looks like not suitable in this assessment. If we use K-NN Classifier to do this, it cannot identify the relationship between the emotion words correctly and it will drop some important information to reduce the accuracy.

## 3.3 Support Vector Classifier

The main problem of SVM is only in binary classification has the best application performance, and other prediction problems are not very good. Obviously, it is not a binary classification in this case.

## 3.4 LSTM Neural Network

From the previous study, I can know that LSTM neural network is very suitable in sentiment analysis. Furthermore, LSTM network can do other typical neural networks cannot do. So I decided to use this algorithm as the main method to finish the predict and the problem is how to set the best parameters and build the perfect network.

# 4.Implement the Object

## 4.1Data loading

The data we use are similary. Below is the data we use for training and test in Figure 4.1 and Figure 4.2

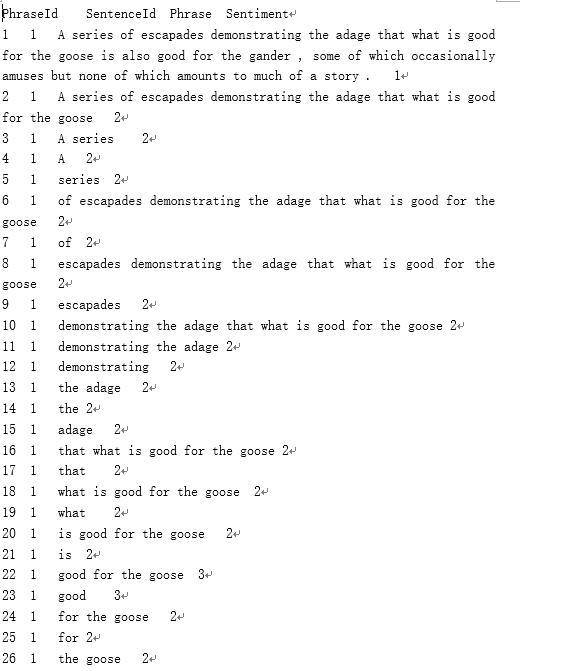


Figure 4.1 Training data

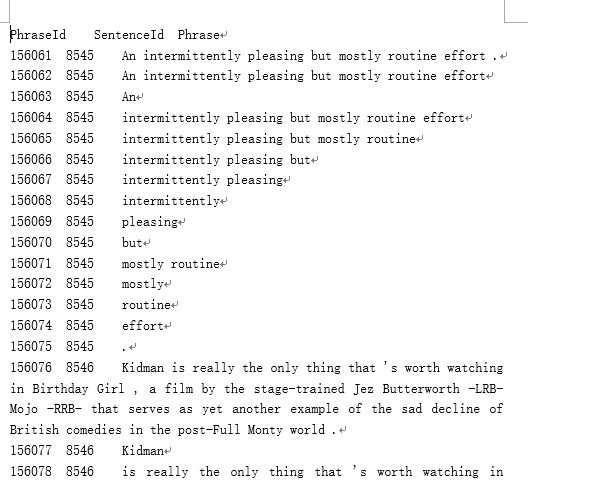


Figure 4.2 Test data

The classifiers I used need digital type data as the input and the data we get is string type. I need to transform these data. Besides the structures of these two data files are different. I need to use two functions and pandas to deal with this problem. In order to load the training data file, I know the feature I need is the ‘Phrase’ column and ‘sentiment ’ column. So after I use pandas to read all the data and get the two lists data. Then return these. It is the same for test data. But the only different is that there is no sentiment in test data and it need the phrase id to predict and output the result.

The following figure is how can I implement the Load file: Figure4.3 and Figure 4.4

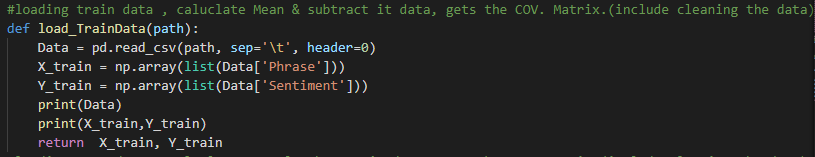


Figure 4.3 load\_train data function

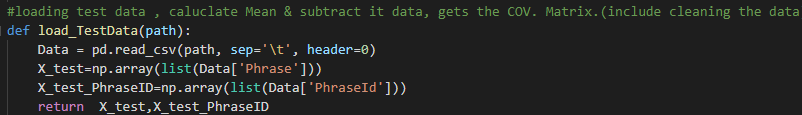


Figure 4.4 load\_test data function

## 4.2Feature Extraction

After reading the file, the next problem is how to transform the string type data to digital data which can be suit for every classifier. LSTM neural network is different from other classifier in feature extraction. As for the feature extraction of LSTM, it need to use ***Tokenizer.fit\_on\_texts*** to create token dictionary to make every element be a document and remove the particular symbol. Then use ***Tokenizer.word\_index*** to calculate the word dictionary size which exclude the same word for the embedding input size. The next step is split the data to train data and validation data. In this case, I use 20% of the whole data as validation data. Then transform the document to vector and set the data shape. Because every sentence has the different length and I want to trim each sentence to its first 60 words. I need to use ***tokenizer.texts\_to\_sequences*** to pad all text to the same size in order to feed them into neural network. Using one hot encoding to do the same thing on the text. The final step is that in order to improve the accuracy of classifier, shuffle the data is necessary. Creating a function named ***shuffle\_data*** by using ***np.random.shuffle*** to shuffle the data randomly. The Figure4.5 below show the part of the code I wrote for feature extraction for LSTM:

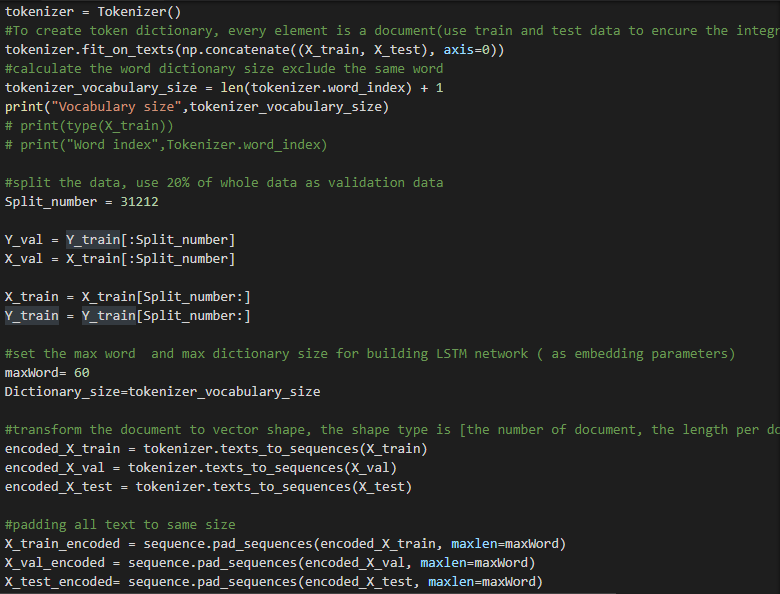
****

Figure 4.5 Feature Extraction for LSTM

As for other classifier, the method of feature extraction I used is the same which is ***TfidVectorizetr*** to transform the data from string to digital. Only use the default the setting of this function is not enough, turning the parameters is necessary. So after many attempts, I found the most efficient parameters’ value which are ***min\_df*** is 5, ***max\_df*** is 0.5, analyzer is word, ***strip\_accents*** is unicode, ***ngram\_range*** is (1,3), ***sublinear\_tf*** is true, ***smooth\_idf*** is true. The following Figure 4.6 below show this part of code of mine:

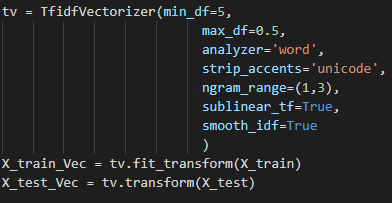


Figure 4.6 Feature Extraction of other classifiers

## 4.3Training Model and Evaluation

After I tried the several kinds of methods, the LSTM neural network had the best performance. So in this part, I will introduce the training model and evaluation steps of LSTM.

The construction process of LSTM neural network was based on Keras. All code I wrote was based on keras functions. My LSTM network has six layers, one input layer, one embedding layer, one LSTM layer, three dense layers. The LSTM layer has 64 outputs neuron. There is a dropout step between the first LSTM layer and first dense layer, and the drop ratio is 0.5 to freeze some neuron to avoid overestimate. The first dense layer has 1200 outputs and use the RELU activation function. The second dense layer has 500 outputs and use the same activation function with the first one. The third layer has 5 outputs which map the five sentiment labels and use the softmax function as activation function.

Using SGD algorithm and callback method can make benefit for model. It can implement that realize observation and interference in the training process and improve the training accuracy. The Figure 4.7 below show the setting of SGD and callback:

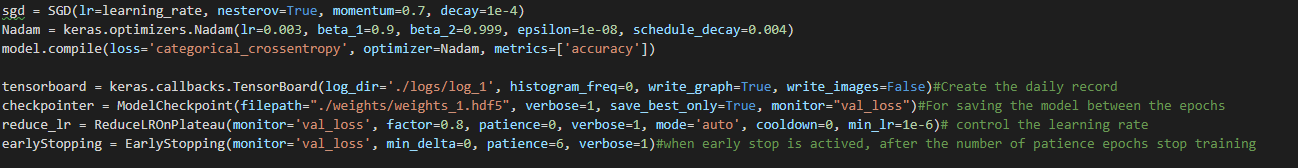


Figure 4.7 SGD and Callback setting

For training, it need to do the data load first and data preprocessing first, then training the model in 10 epochs again and again to find the best parameters to get the best score of the result. After getting a considerable score, I input the test data into the trained model to get the result and submitted it to the Kaggle to get the predict accuracy of my model.

Obviously, the evaluation method is the accuracy from Kaggle of the predict result of my model. The result of my LSTM model is about 63% which is shown in the Figure 4.8 below:



Figure 4.8 The accuracy of LSTM result

Furthermore, there are some other classifiers which I used for sentiment analysis. Such as Naïve Bayes, Logistic Regression, Support Vector Machine and K-NN. I carried out all the algorithm and each one I created a function to use as Figure 4.9 below and one example of SVC is like Figure 4.10. The only difference is the algorithm different and the code is similar.

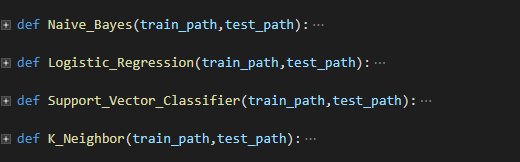


Figure 4.9 Other Classifiers

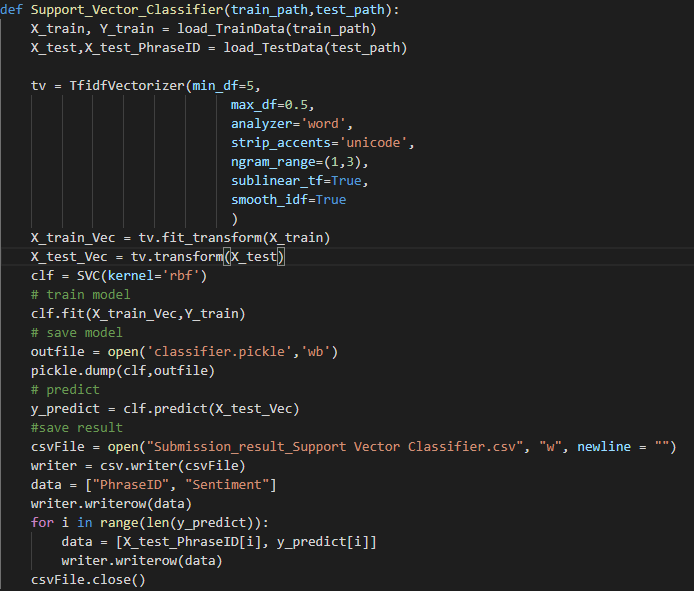


Figure 4.10 The example of SVM implement code

Besides, I submitted all the classifier’s predict results to Kaggle and get the accuracy report of Figure 4.11 below:

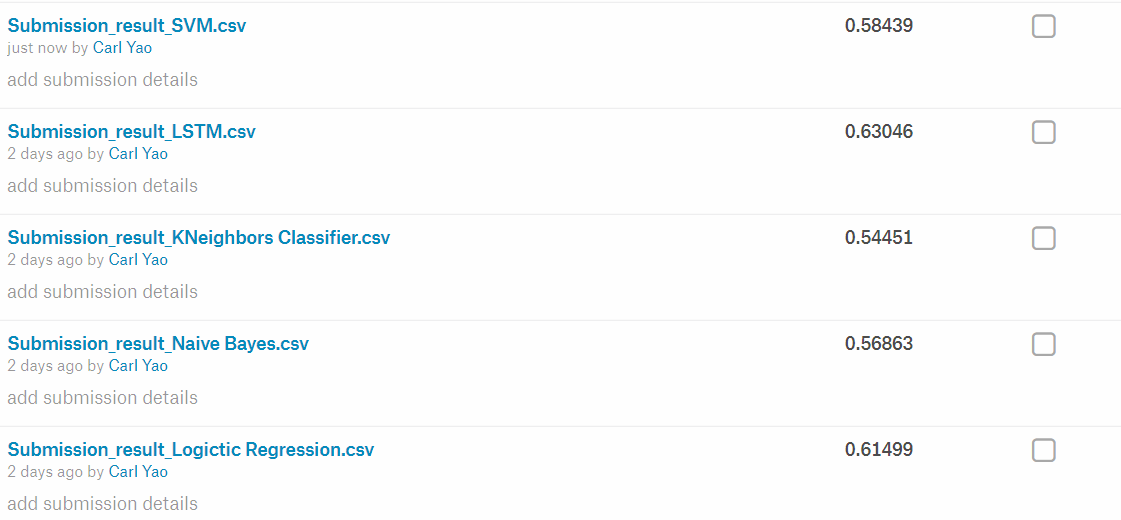


Figure 4.11 Other classifier accuracy on Kaggle

# 5.Comparison with the state of the art

With the development of Nature Language Processing (NLP), there are several new methods appeared for sentiment analysis. But as for classification, all current state of art approaches use neural network. Context Modelling Using Hierarchical Attention Networks for Sentiment and Self-assessed Emotion Detection in Spoken Narratives by Lukas Stappen’s team.[9] They use the Ulm State-of-Mind in Speech(USoMS) corpus and combined with Self-Assessed Affect Sub-Challenge of the Interspeech to help participants in Computational Paralinguistic Challenge to predict self-assessed valence.

According to this research, task for predicting narrative emotions, the results of GRU based encoders are considerably superior, with the strongest test UAR, 91.0 % achieved by the attention enhanced AttBiGRU.[9] For the more complex task of prediction the self-assessed labels, the accuracy of the result is 73.7%.

Compare with the traditional one method based sentiment analysis, this research combined the sentiment dictionary and machine learning and get a high accuracy result. It is a big step forward in sentiment analysis research.

# 6.Lessons Learned

Through the assignment, this assignment was done by myself and I found that it will be an important experience for me. Compared with the assignment 1, this assignment is more difficult, it need to use more complex algorithm and exercise my programming skills. In this assignment, I have more time to learn and try different methods even deep learning.

Furthermore, it was the first time I used the Tensorflow and Keras, and give me a new understanding of these. It will be a good start for me to continue my study.

As for the parameters tuning. I found that there is a function included in Scikit-Learin which can tune the parameter automatically. This function is GridSearchCV, it was really an important discover for me. It can greatly shorten the time when I adjust the parameters. Besides, it is the first time I used the shuffle function, in the past I don’t know there is a convenience way to cross validation the model and the previous method I use is much more complex.

The problem I still have is that I know how to build a LSTM neural network and how to use it by Keras. But I cannot understand the principle of LSTM clearly. I think I have to spend more time to put the whole of it principle in my head in the future.

# 7.Conclusion

The LSTM is really have the best performance in this situation. For other classifiers, because of the time limited, I only use the default setting to train. Maybe in the future, I will try to turning other classifier’s parameters to improve the accuracy.

**Reference**

[1]. C. Strapparava and A. Valitutti, "Wordnet-affect: an affective extension of wordnet," in Proceedings of the 4th International Conference on Language Resources and Evaluation, 2004.

[2]. G. Vinodhini, R. Chandrasekaran, "Sentiment analysis and opinion mining: A survey", International Journal of Advanced Research in Computer Science and Software Engineering, vol. 2, no. 2277128X, 2012.

[3] En.wikipedia.org. (2019). *Naive Bayes classifier*. [online] Available at: https://en.wikipedia.org/wiki/Naive\_Bayes\_classifier [Accessed 22 Apr. 2019].

[4] "K-nearest neighbors algorithm", *En.wikipedia.org*, 2019. [Online]. Available: https://en.wikipedia.org/wiki/K-nearest\_neighbors\_algorithm. [Accessed: 22 Apr 2019].

[5] LANGLEY, P. and SAGE, S. (2019). Induction of Selective Bayesian Classifiers. Institute for the Study of Learning and Expertise, pp.399-403.

[6] Yang, Z., Salakhutdinov, R. and Cohen, W. (n.d.). [online] Arxiv.org. Available at: https://arxiv.org/pdf/1603.06270.pdf [Accessed 24 Feb. 2019].

[7] Sepp Hochreiter and Jurgen Schmidhuber, "D LONG SHORT-TERM MEMORY," Neural Computation 9(8):1735-1780, 1997

[8] Raaj, V. (2018). Sentiment Analysis on movie reviews using CNN-LSTM architecture. [online] Medium. Available at: https://medium.com/@vivanraaj/sentiment-analysis-on-movie-reviews-using-cnn-lstm-architecture-81c6305d4605 [Accessed 22 Apr. 2019].

[9] Stappen, L., Cummins, N., Meßner, E., Baumeister, H. and Dineley, J. (2019). Context Modelling Using Hierarchical Attention Networks for Sentiment and Self-assessed Emotion Detection in Spoken Narratives - IEEE Conference Publication. [online] Ieeexplore.ieee.org. Available at: https://ieeexplore.ieee.org/document/8683801 [Accessed 22 Apr. 2019].