Multi-class Sentiment Analysis using Deep Learning

Kevinkumar Patel
Student Id: 1107736
Lakehead University

***Abstract*—The purpose of this project is to apply deep learning methods to the challenge of analysing the sentiment of film reviews. For fact, there are several platforms for film rating, such as Rotten Tomatoes, IMDB and Flixster, which people can rate depending on their own feelings. My sentiment analysis project aims at using movie review text with associated labels from Rotten Tomatoes to classify phrases on a scale of five classes: negative, somewhat negative, neutral, somewhat positive, positive (I use numerical value 0, 1, 2, 3, 4 to represent them respectively).In this paper I will be giving robust solution based on Convolutional Neural Network(CNN) and will be be taking advantage of the Count Vectorizer to count the term frequency. In this task generated model will be evaluated based on Accuracy, Recall, Precision, and Figure-of-Merit (f-1) score.**

***Index Terms*—Sentiment analysis, Natural Language Processing(NLP), 1D-CNN, Recall, precision, figure-of-merit (f-1) score, Deep Learning, Neural network.**

I. Introduction

Throughout the Big Data period, there is an enormous growth in the volume of different data, such as image, video, sound and text. Text is the largest among them so studies relating to text research were carried out extensively from the past to the present. For example, the consumers of Social Network Services (SNS) often reflect their nostalgic feeling, and they also exchange some opinions regarding regular news with the public or friends. Emotional analysis includes emotion categories (e.g., pleasure, excitement, satisfaction, angry), while attitude analysis contains categories such as positive, neutral, and negative.One of NLP's most important areas is the study of emotions. Analyzing sentiment is the method of unravelling or extracting concrete patterns from text details. Examination of emotions will help us gain the mindset and atmosphere of the general population which will then help us collect valuable background knowledge. It will then allow one to anticipate and make the measured choices correctly. If such a sentiment is accurately predicted, then it will be applicable to the idustry for movie recommendation, personalized newsfeed.

Machine Learning(ML) techniques are quite usefule for the sentiment analysis but in ML feature definition requires much effort of domain experts. Recently, deep learning approaches have gained interest for this purpose, because they can minimise the effort for identifying the function and obtain fairly high accuracy.In this paper, i aim at classifying sentiments for text data and suggest a CNN architecture, which is a form of deep learning model.

In This paper I used Rotten Tomatoes Reviews dataset for training and testing. For this i used preprocessing to clean the data and then i split the data into 70% training and 30% for testing.I also used Count Vectorizer to count the frequency of the words.

This paper can be summarized as follows: (1)I build two consecutive convolution layers of architecture to increase the output for long and complex texts. (2)I applied the CNN model to multiclass sentiment classification and obtained F1 scores, accuracy, recall and precision of the model.

II. Background

A. Deep Learning for Sentiment Classification

The deep learning method, one of the methods of artificial intelligence, has been commonly used lately to characterise emotions. Dong et al.[1] developed a new model, the Adaptive Recursive Neural Network (AdaRNN), which classifies the Twitter texts into three categories of sentiment: positive, neutral and negative. The AdaRNN obtained 66.3 per cent accuracy by experimental tests. Tang et al.[2] implemented a different version of the RNN model, the Gated Recurrent Neural Network (GRNN), which obtained accuracies of 66.6% (Yelp results for 2013–2015) and 45.3% (IMDB results). Each of these experiments generally believed three or four emotion labels were in effect. Kim [3] had a result of a maximum of 89.6% accuracy with seven different types of data through their CNN model with one convolutional layer. Deriu et al. [4] trained the CNN model that had a combination of two convolutional layers and two pooling layers to classify tweet data of four languages sentimentally and obtained a 67.79% F1 score. I performed studies with some of CNN models that had various configurations, and my model is similar to [4] model.

When mentioned above, two prevailing forms of deep learning methodology occur for the classification of sentiments: RNN and CNN. Throughout this study, I am proposing a CNN model, whose structure was specifically crafted for efficient classification of sentiments. Throughout the next chapter we clarify the CNN's advantages of text processing.

B. Convolutional Neural Network for Text Classification

To provide reliable and effective data analysis, the rapid increase in the amount of complicated data sets per year demands further improvement of machine learning methods. Lately, deep learning methods on tasks such as image analysis, natural language processing, object recognition, etc. are producing improved performance relative to previous machine learning algorithms. These deep learning algorithms performance depends on their ability to model complex and nonlinear interactions within the data.It has been shown that even without any special hyperparameter modification a single convolutional layer, a combination of convolutional filters, could achieve comparable performance[3]. In fact, the CNN does not require specialist information on the linguistic structure of a target language[5]. The CNN has been effectively extended to various text analyses due to these advantages: textual filtering, quest by email, sentence modelling.

In this paper, we design a CNN model for the sentiment classification and show that our network is better than other deep learning models through experimental results.

III. Data Preprocessing

A. Dataset

There are several platforms for film rating, such as Rotten Tomatoes, IMDB and Flixster, which people can rate depending on their own feelings.I used Rotten Tomatoes Reviews dataset for the text classification. In this dataset phrases are classified into this classes: negative, somewhat negative, neutral, somewhat positive, positive (I use numerical value 0, 1, 2, 3, 4 to represent them respectively). Table 1 shows the first five rows of the dataset.

TABLE I
First Five row of the dataset

PhraseId	SentenceId	Phrase	Sentiment
1	1	A series of escapades demonstrating...	1
2	1	A series of escapades...	2
3	1	A series	2
4	1	A	2
5	1	series	2

Data Analysis is an essential aspect of market practises because companies collect a massive volume of data today. Worldwide, sensors capture temperature data, user data by taps, car data for steering wheel prediction etc. Many of these gathered data contain important lessons and visualisations for companies, allowing these observations simple to understand.

Data visualizations in python can be done via many packages. I used matplotlib library for the data visualization. Matplotlib is a 2-D plotting library that aids in figure visualisation. Including diagrams and visualisations.Matplotlib is used in Python is a stable, free and fast framework for visualising results. Fig. 1 represents the categorised dataset which is shown below in which 0,1,2,3,4 are for the negative, somewhat negative, neutral, somewhat positive, positive.There are 7072,27273,79582,32927,9206 data are for respected category. So basically its around 156060 data that is huge number of review.So for training this much data i need high power Graphics processing unit(GPU) so i used google colab to train my model.Colab Notebooks is a Jupyter notebook environment that requires no setup to use and runs entirely in the cloud.

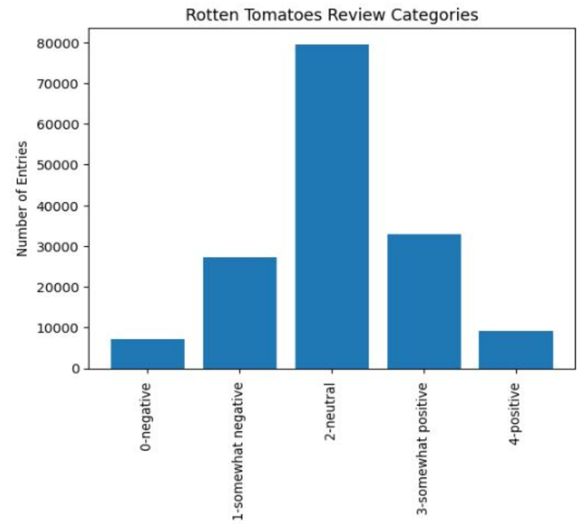

Fig. 1. categories of the data

B. Data Cleaning

Preprocessing was carried out to modify the text data appropriately in the experiment. I used decapitalization and did not mark the start and end of the sentences.Since the original dataset only contains raw text with its name, i need to convert the natural language shape and delete noises to suit our model better. The preprocessing portion of the data thus comprises the following steps: Tokenization and Segmentation, Noise Removal(Remove stop words), Lemmatization and Normalization.

With the progression of the popular Python NLP library-NLTK, we can tokenize and segment our reviews of the film. Tokenizers explicitly partition the strings into substring sets. Tokenizers may be used to locate the terms and punctuation in a list (from the dataset) for example:”This movie is one time watch” can be tokenized into something like [’This’,’movie’,’is’,’one’,’time’,’watch’].

When the text from our film analysis dataset has been extracted, the task is to make sense of the raw material. Text cleaning is used broadly for most text cleaning, depending on the source of the data, output filtering, external noise and so on. So then after i used NLTK library to remove the stopwords from the dataset.Stop words are basically a set of commonly used words in any language, not just English.For example, in this ”How to develop the java application” in which ’develop’, ’java’, ’application’ are the most focused words in the sentence but ’how’, ’to’, ’the’ are the stopwords which are no more useful to get the better result. Thus with the use of the python NLTK I remove those kind of stopwords.

The goal of stemming in English text processing is to minimise inflectional forms and often derivationally linked types of a word to a common type of base. There are various combinations of terms and phrases in my data set, such as’ movies,” film’ and’ movie’ have the same meaning-’movie’ but they can be viewed separately by computer.

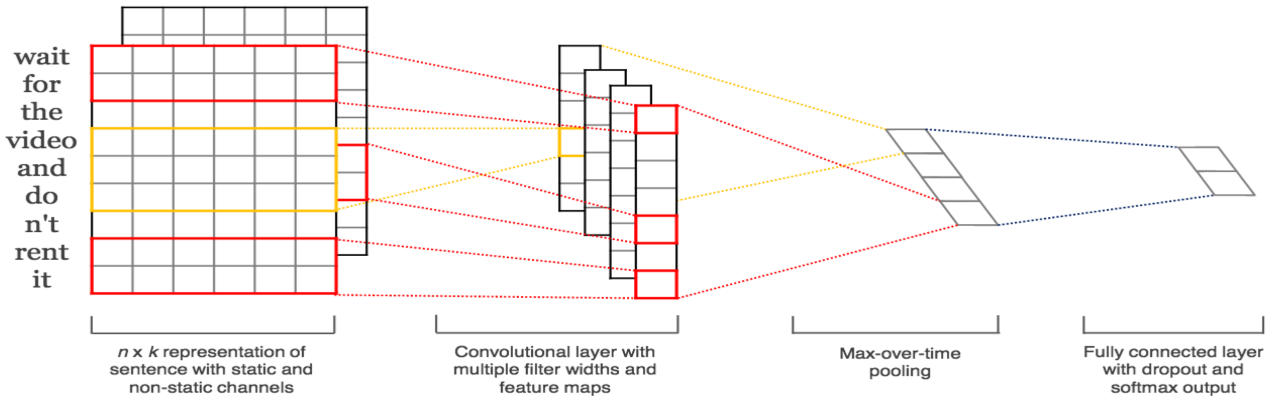

Fig. 2. CNN model

Therefore, efficient mapping of different terms to the same stem is necessary, even though this stem is not a true root in itself. In this task, I used The Porter Stemming Algorithm as the basic method of stemming. As an example I take a sentence from my film reviews:"A series of escapades demonstrating the adage" will be converted into "seri escapad demonstr adag good goo als." Result is after using removing the stopwords and tokenization and stemming.

As we know robots, as sophisticated as they can be, can not comprehend vocabulary and sentences in the same manner that humans can. To make the corpora of papers more accessible to machines, first they have to be translated into some numerical form. To do this, there are a few techniques employed. Selection of features plays a very important role in the processing of machine learning and natural language which serves two key purposes. Firstly, by decreasing the size of the successful vocabulary it allows training and implementing a classifier more productive. Second, collection of features also improves the precision of classification by reducing the noise characteristics. There are multiple technqiues available for the feature selection like Bag of Words, TF-IDF, Word Embedding and Count Vectorizer. I used Count Vectorizer the most straightforward one, it counts the number of times a token shows up in the document and uses this value as its weight.

IV. Proposed Model

CNN, commonly used on image databases, removes the image's important attributes, while the "convolutional" filter passes around the picture. If the input data are given as one-dimensional, the same CNN feature may also be used in the document. Local text knowledge is retained in the text field, as the philtre travels, and essential features are removed. Then it is efficient to use CNN for classification of texts. A graphical representation of the image of the proposed network is shown in Figure 2.The network consisted of an two convolutional layers, a pooling layer and a layer that was completely connected. Basic working of a CNN in which tensor is translated into a simple CNN for image processing with a collection of kernels of the sized by d. Such layers of convolution are called feature maps, which can be stacked to have several input filter. CNNs utilise pooling to minimise the numerical overhead, and decreases the performance size from one layer to the next inside the network. Various pooling methods are used to reduce performance while retaining critical characteristics. In this model the first convolutional layer is used to look at basic qualitative details when looking at the matrix, while the second convolutional layer is used to grab while remove key features (e.g., bad, great) that include emotions that influence classification.

The matrix going through the consecutive convolution layer is used as the reference to the pooling layer. Although average pooling and L2-norm pooling were used as the pooling layer location, I used max pooling in this project, which is a technique for selecting the greatest value as a representation of peripheral values.As the emotion is always defined by a mixture of many terms rather than voicing the meaning in each word in the paragraph, we have followed the methodology of max-pooling. The pooling layer slides with an optional phase all of the matrix values, which is the contribution of the second convolutional layer, resulting in contribution vectors. Because max-pooling is the layer that transfers the largest value of many values to the next layer, this results in far smaller performance vectors. The convolutional layer, in other words, looks at the meaning and selects the key elements, and the pooling layer plays a part in choosing the most popular features.A flattening method is conducted after going through the pooling layer to transform the two-dimensional function map from the display into a one-dimensional format and transfer it to a Fully-Connected (FC) layer of F-dimensions. Because the FC layer takes a one-dimensional vector as the origin, it is essential to flatten the two-dimensional vector supplied from the pooling layer. The FC layer binds both neurons at the input and the exit. A vector moving through the FC layer generates an contribution marked as positive or negative. The softmax activation feature helps to distinguish various groups in the FC network.In this paper, I argue that data which passes through two consecutive convolution layers and then passes through the pooling layer is efficient in storing contextual information and extracting prominent features.

V. Experimental analysis

I conducted experiments on multiclass classification. This task is based on mulitclass classification, where the each instance classified to the either of these catories "negative","somewhat negative", "neutral", "somewhat positive","positive" classes respectively. The evaluation of the model is based on loss, accuracy, F1 score, precision measure and recall respectively. All of these attributes depends on the characteristics of the model in which number of parameters used for testing, batch size, kernel size, structure of the model with different models, activation functions used in each layers and optimizer used.

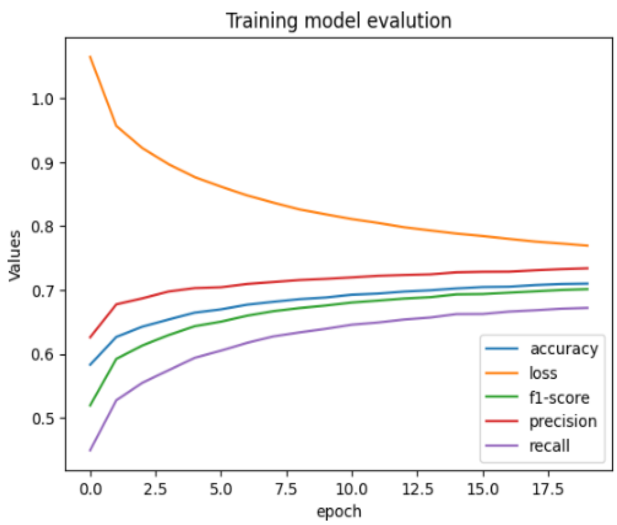

Fig. 3. Training results

After all posibilities of vectorization modules and all other parameters of model, the optimal solution for the proposed model was given by using ReLU and softmax as activation layers and Adadelta as an optimizer. The result provided from this combination is shown in the figure 3. As an evaluation result, the loss, accuracy, F1 score, precision measure and recall obtained were 0.98, 0.6417, 0.630, 0.662 and 0.60 percent respectively for the 20 epochs. Time taken to train this model is around 736.84 seconds.

VI. Conclusion

I have developed a convolutional neural network for the classification of sentiments in this paper. Through experimental findings, I found that on fairly long text the consecutive convolutional layers led to a better efficiency. For the multiclass classification the theoretical CNN models obtained around 64% accuracy. I will extend my research, as potential study, to certain classification activities (e.g., gender classification). I will also try to find improved mechanisms for classifying sentiments; for example, for piling more levels of the convolutional layers, residual relations.

References

[1] Dong, L.; Wei, F.; Tan, C.; Tang, D.; Zhou, M.; Xu, K. Adaptive recursive neural network for target-dependent twitter sentiment classification. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Baltimore, Maryland, USA, 23–25 June 2014; Volume 2, pp. 49–54.

[2] Tang, D.; Qin, B.; Liu, T. Document modeling with gated recurrent neural network for sentiment classification. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 17–21 September 2015; pp. 1422–1432.

[3] Kim, Y. Convolutional neural networks for sentence classification. arXiv 2014, arXiv:1408.5882

[4] Deriu, J.; Lucchi, A.; De Luca, V.; Severyn, A.; Müller, S.; Cieliebak, M.; Hofmann, T.; Jaggi, M. Leveraging large amounts of weakly supervised data for multi-language sentiment classification. In Proceedings of the 26th International Conference on World Wide Web, Perth, Australia, 3–7 April 2017; pp. 1045–1052.

[5] Zhang, X.; Zhao, J.; LeCun, Y. Character-level convolutional networks for text classification. In Advances in Neural Information Processing Systems 28; Neural Information Processing Systems Foundation, Inc., Montreal, Quebec, Canada, 2015; pp. 649–657