Emotion Detection and Sentiment Analysis in Texts

Naman Pundir
University at Buffalo, New York, US
namanpun@buffalo.edu

Abstract

Emotion Detection and Recognition from images can be determined by computer vision and deep learning algorithms and is being used in many industry but emotion detection in texts is a new field where we can detect the sentiment of the people by processing public opinion in various platforms like Twitter, Facebook and more. There are 6 emotions in humans according to the facial expression namely happy, sad, fear, disgust, surprise, and anger. This paper explains how we can use various machine learning and deep learning algorithms to detect emotions from the texts. We will be working on 1D-CNN, LSTM and BERT transformer, discussing the algorithms, comparing their performances and discussing the future scope of emotion detection in text on stock analysis and prediction.

[GitHub Repository for the paper: https://github.com/namanpundir/Emotion_detection_in_text]

1 Introduction

There are 6 emotion categories that are widely used to describe humans' basic emotions, based on facial expression: anger, disgust, fear, happiness, sadness and surprise. The question remains, however, how much of an emotion we can convey via text? How we can determine the emotion behind the text? Can we detect emotions in text like in images?

This is particularly intriguing because in spoken language, face expression and voice tone convey about 70% of the intended feelings. Some computer vision and deep learning approaches can be used to detect emotions in photos or movies. Here, we'll look at how we can use neural networks and transformers to accomplish this goal in text, with a focus on deep learning.

We will be using 1D-CNN (Convolution Neural Network), LSTM (Long Short-Term Memory) and BERT (Bidirectional Encoder Representations from Transformers) transformer to detect emotion in the texts and comparing these models based on their f1 score.

2 Datasets

To train these models we need some datasets which contain some sentences and labelled emotions. We are using three datasets to train our models which are Daily-Dialogue[link], Emotion-Stimulus[link], ISEAR[link].

All three datasets were merged and split into train and test dataset to get the variety of data to train the model.

Dataset	Year	Size
Daily-Dialogue	2017	102k lines
Emotion-Stimulus	2015	2.5k lines
ISEAR	1990	7.5k lines

Datasets are labeled with 5 labels sad, anger, joy, fear and neutral. Text in the datasets are simple dialogue or small texts. We will use these datasets to train our models and to test them.

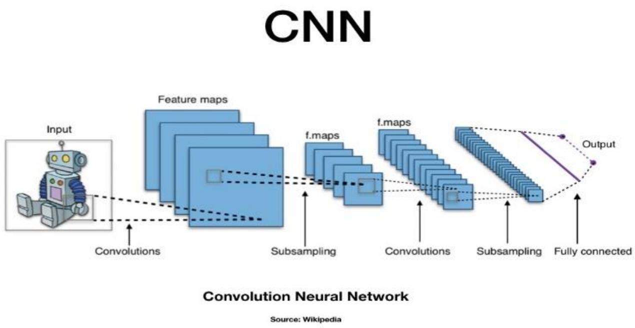
Anger	Text
neutral	There are tons of other paintin
sadness	Yet the dog had grown old and l
fear	When I get into the tube or the t
fear	This last may be a source of con
anger	She disliked the intimacy he sho
sadness	When my family heard that my l
joy	Finding out I am chosen to colle
anger	A spokesperson said : ` Glen is fu
neutral	Yes .
sadness	When I see people with burns I
fear	I had gone to the hospital for m
sadness	One day I heard from a friend th

3 Models

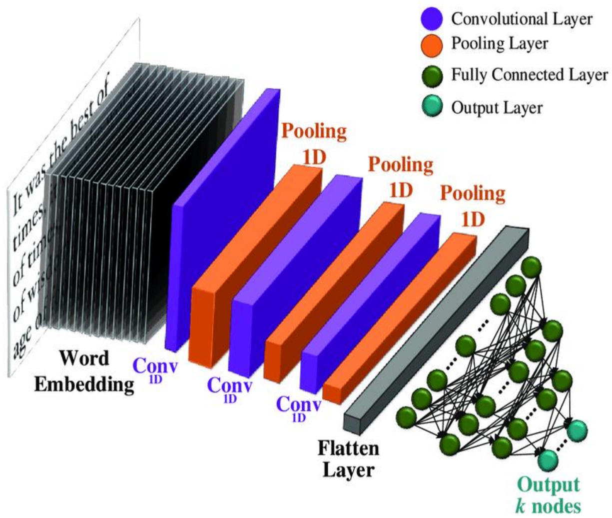
We are using three models CNN, LSTM(RNN) and BERT to detect the emotions in the text.

1. CNN (Convolution Neural Network)

Convolutional neural network (CNN), a class of artificial neural networks that has become dominant in various computer vision tasks, is attracting interest across a variety of domains, including radiology, medical image classification.



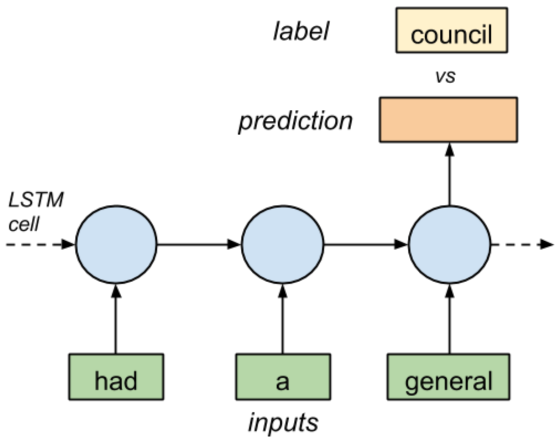
CNN is designed to automatically and adaptively learn spatial hierarchies of features through backpropagation by using multiple building blocks, such as convolution layers, pooling layers, and fully connected layers. CNN nowadays is used in many image and videos classification tasks including medical imaging and sports video analysis.



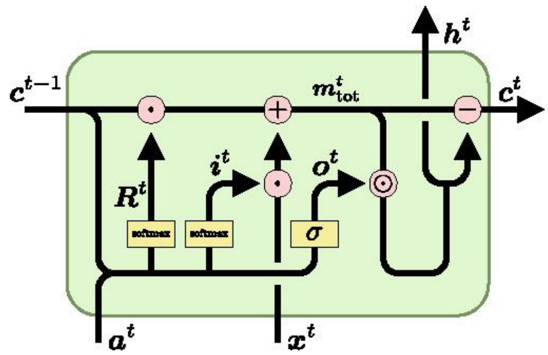
2. LSTM (Long Short-Term Memory)

Long Short-Term Memory networks (LSTM) are a special kind of RNN (Recurrent Neural Network), capable of learning long-term dependencies. The basic idea behind LSTM is to keep the important context in the long-term memory for further use and less important contextual data in the short memory which might not be used less in the future process. They were introduced by Hoch Reiter & Schmid Huber (1997), and were refined and popularized by many people in following work.

They work tremendously well on a large variety of problems and are now widely used.

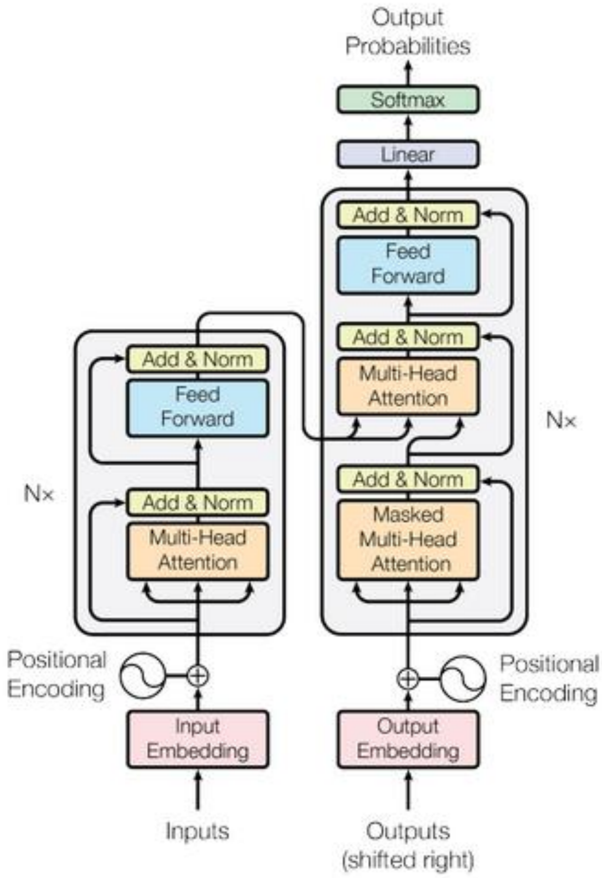


LSTMs are specifically developed to prevent the problem of long-term dependency. They don't have to work hard to remember knowledge for lengthy periods of time; it's nearly second nature to them. All recurrent neural networks are made up of a series of repeated neural network modules. This repeating module in ordinary RNNs will have a relatively simple structure, such as a single tanh layer.

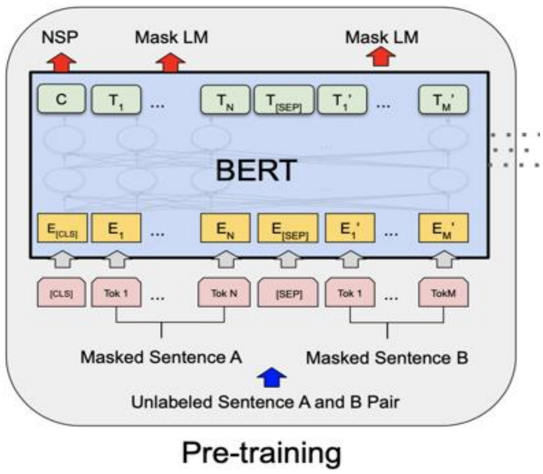


3. BERT (Transformer):

The Transformer in NLP is a novel architecture that aims to solve sequence-to-sequence tasks while handling long-range dependencies with ease. The Transformer was proposed in the paper Attention Is All You Need [link]. Attention is used in the transformers block and brought the revolutionary change in the NLP industry. This is the quote from the above-mentioned paper: “The Transformer is the first transduction model relying entirely on self-attention to compute representations of its input and output without using sequence-aligned RNNs or convolution.”



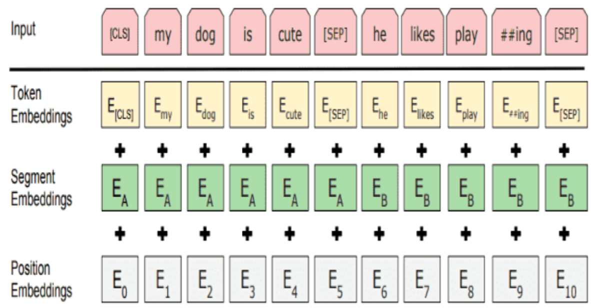
BERT (Bidirectional Encoder Representations from Transformers) is a recent paper published by researchers at Google AI Language. It has caused a stir in the Machine Learning community by presenting state-of-the-art results in a wide variety of NLP tasks. It is the most widely used transformer in the industry.



The use of the bidirectional training of Transformer, a popular attention model, to language modeling is BERT's fundamental technical breakthrough. Previously, researchers looked at a text sequence from left to right or a combination of left-to-right and right-to-left training.

As opposed to directional models, which read the text input sequentially (left-to-right or right-to-left), the Transformer encoder reads the entire sequence of words at once. Therefore, it is considered bidirectional, though it would be more accurate to say that it's non-directional. This characteristic allows the model to learn the context of a word based on all its surroundings (left and right of the word).

Bert use three embeddings to compute the input representations.



4 Working with CNN Model:

[https://github.com/namanpundir/Emotion_detection_in_text]

Architecture:

(X) Text

Embedding (W2V pretrained on wiki)

Deep Network (CNN 1D)

Fully connected (Dense)

Output Layer (SoftMax)

Emotion class (Y)

Embedding Layer:

Word Embedding is a representation of text where words that have the similar meaning have a similar representation. We will use 300-dimensional word vectors pre-trained on Wikipedia articles. We can also train the w2v model with our data, however our dataset is quite small and trained word vectors might not be as good as using pretrained w2v.

Deep Network:

Though text data is one-dimensional, we can use 1D convolutional neural networks to extract features from our data. The result of each convolution will fire when a special pattern is detected. By varying the size of the kernels and concatenating their outputs, we are allowing ourselves to detect patterns of multiples sizes. Patterns could be expressions like "I love", "very less" and therefore CNNs can identify them in the sentence regardless of their position.

Fully Connected Layer:

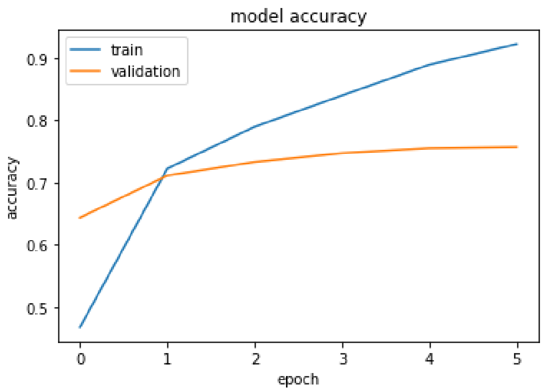
The output from the convolutional layers represents high-level features in the data. While that output could be flattened and connected to the output layer, adding a fully connected layer is a (usually) cheap way of learning non-linear combinations of these features.

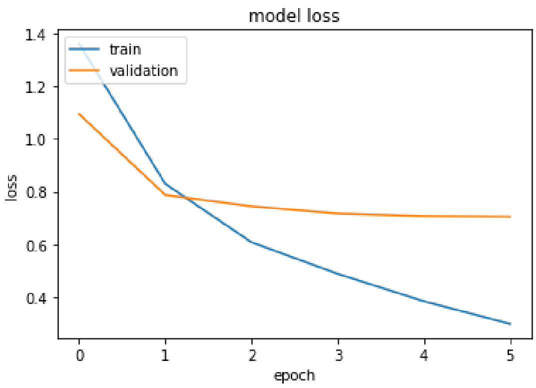
Output Layer:

Based on the problem at hand, this layer can have either Sigmoid for binary classification or SoftMax for both binary and multi classification output. These are the activation functions which are responsible to help the network to learn the complex data.

Model:

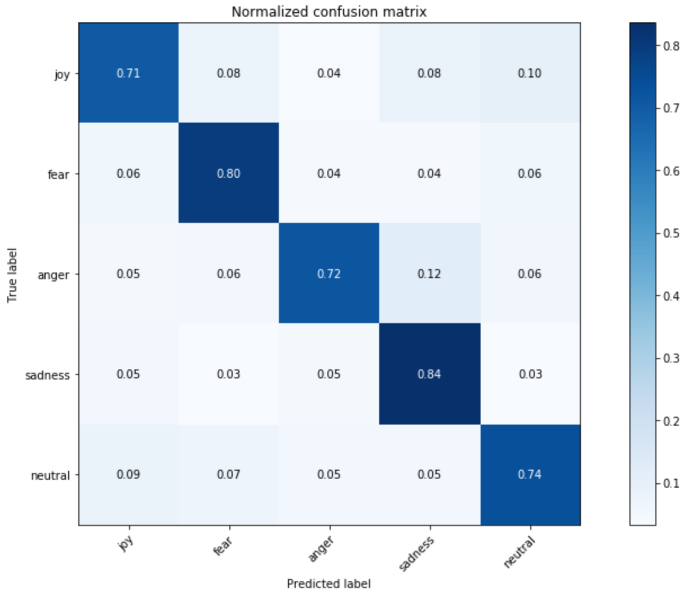
```
Layer (type)                 Output Shape              Param #
=================================================================
embedding (Embedding)        (None, 256, 300)          3626700

conv1d (Conv1D)              (None, 254, 256)          230656

global_max_pooling1d (Globa  (None, 256)               0
lMaxPooling1D)

dense (Dense)                (None, 256)               65792

dense_1 (Dense)              (None, 5)                 1285

=================================================================
Total params: 3,924,433
Trainable params: 297,733
Non-trainable params: 3,626,700
```





Results: F1 Score: 75.6

Confusion Matrix:



5 Working with LSTM Model:

(X) Text

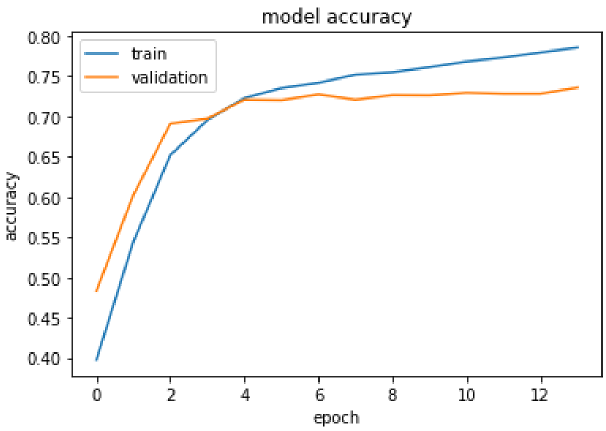
Embedding (W2V pretrained on wiki)

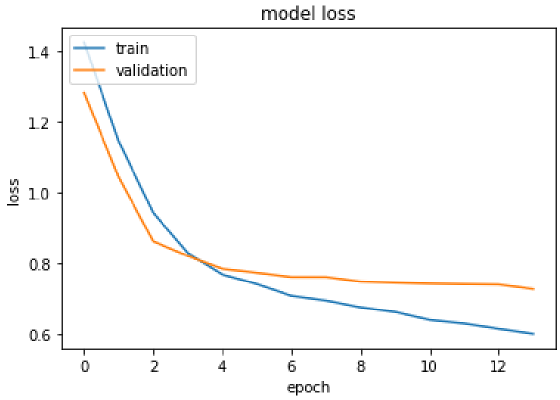
Deep Network (LSTM)

Fully connected (Dense)

Output Layer (SoftMax)

Emotion class (Y)

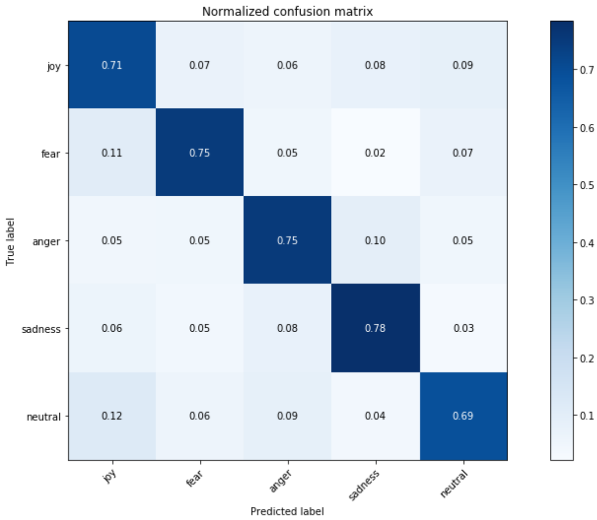




Results: F1 Score: 73.56

Example of detecting emotions from our LSTM trained model:

```
Message: ['delivery was hour late and my pizza was cold!']
predicted: anger (0.06 seconds)
```

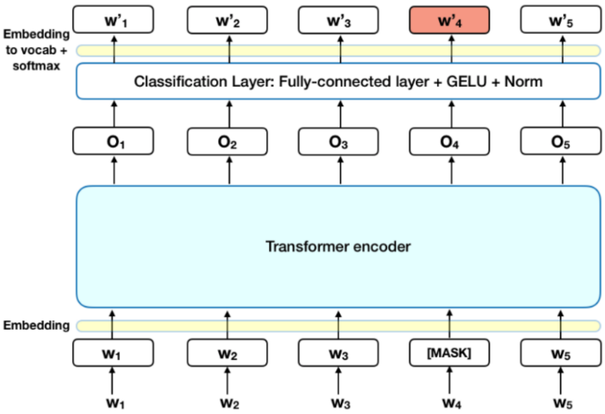


6 Working with BERT Model:

Training a Bert model takes a lot of time and computational power. We just ran model for 1 epoch with maximum learning rate of 2e-5 on google colab GPU for 11 hours.

Results: F1 Score: 0.83

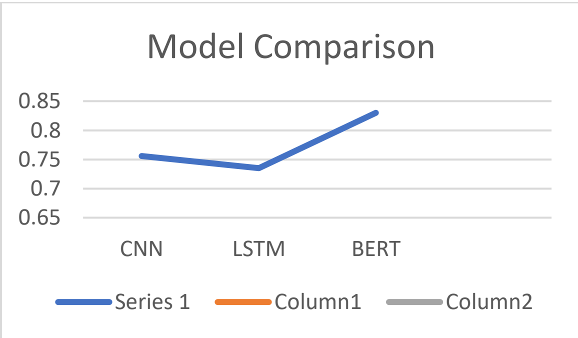
	precision	recall	f1-score	support
joy	0.83	0.85	0.84	707
sadness	0.81	0.79	0.80	676
fear	0.85	0.85	0.85	679
anger	0.81	0.77	0.79	693
neutral	0.79	0.83	0.81	638
accuracy			0.82	3393
macro avg	0.82	0.82	0.82	3393
weighted avg	0.82	0.82	0.82	3393



7. Result Table:

Model	F1 Score
CNN	0.756
LSTM	0.735
BERT	0.830

We can observe that BERT transformer gives highest F1 score among all the models. Which clearly shows that the BERT model is the best model among CNN, LSTM and BERT to detect emotions in text.



8. Conclusion

As we know there are different kind of data i.e. 1D vector data, 2D data, Sequential data and 3D data. Text is kind of a sequential data and to process the sequential data we have three methods 1D CNN, LSTM, and Transformers, which we used with these datasets.

We can clearly see that BERT transformer is more accurate predicting emotions as compare to CNN and LSTM in the above three datasets.
BERT is better than other as it is pre trained on an absurd amount of data. BERT model comes in two sizes one is BERT-base (800million words) and BERT-large (2500million words). That makes BERT better than other models.
BERT is the first deeply bidirectional, unsupervised language representation, pre trained only a plain text corpus.

9. Applications

Emotion detection and sentiment analysis using transformers is one of the hottest topics in the tech industry.
Applications like **social media monitoring** which help companies about some of the most truthful points of view about products, services, and business because users offer their opinions unsolicited.

Sentiment analysis **Customer Support** reads regular human language for meaning, emotions, tone, and more, to understand customer requests, just as a human would.

Sentiment analysis can be used to gain insights from the **Customer Feedback** available and save employee hours and can also work with common mistakes like misspelled and misused words.

8. Scope of Work

Emotion detection and sentiment analysis can be used to detect sentiment behind the stock market prediction which can lead to better investment opportunity.

```
message = 'Tesla is overbought. You should wait for this stock as market is volatile.'

start_time = time.time()
prediction = predictor.predict(message)

print('predicted: {} ({:.2f})'.format(prediction, (time.time() - start_time)))
```

```
predicted: fear (2.32)
```

Lot of people put their views on social media platforms like Twitter and Facebook and detecting emotions behind these views by scraping the data and using models like BERT, which can give an insight about the stocks and future prediction. The movement of any stock depend on two factors one is the company work, policies and revenue and the other is public sentiment on that stock.
There is a scope of work of creating a model which can scrape the data from social media platform about the stocks and detect the major emotions behind the stock which can be used to predict the future performance of the stocks.

9. Reference:

1. Emotion Detection from Text: Shiv Naresh Shivhare and Prof. Saritha Khethawat

https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.301.9872&rep=rep1&type=pdf#:~:text=Emotion%20Detection%20in%20text%20documents,in%20emotion%20detection%20are%20discussed.

2. Text-based emotion detection: Advances, challenges, and opportunities https://onlinelibrary.wiley.com/doi/full/10.1002/eng2.12189

3. Emotion detection in Text: Armin Seyeditabari, Narges Tabari, Wlodek Zadrozny https://arxiv.org/abs/1806.00674

4. Classify text with BERT https://www.tensorflow.org/text/tutorials/classify_text_with_bert

5. Why and how to use BERT for NLP Text Classification? https://www.analyticsvidhya.com/blog/2021/06/why-and-how-to-use-bert-for-nlp-text-classification/

6. Daily Dialogue - https://arxiv.org/abs/1710.03957

7. Emotion-stimulus - https://metatext.io/datasets/emotion-stimulus

8. Isear(International Survey on Emotion Antecedents and Reactions)- https://paperswithcode.com/dataset/isear

9. Attention is all you need: https://arxiv.org/abs/1706.03762

10. Chowdhury GG. Natural language processing. *Ann Rev Inf Sci* https://asistdl.onlinelibrary.wiley.com/doi/full/10.1002/aris.1440370103?casa_token=cvvV-T3fVmQAAAAA%3A_1-M4d-lYvjNkdrhFeDDtwz08n_sUhVT_5Dd5rms6TT1PNynXTGDZxuOt5ZA6elTzz_Gp_I41oFdsg .

11. https://devblogs.microsoft.com/cse/2015/11/29/emotion-detection-and-recognition-from-text-using-deep-learning/