[[1]](#footnote-1)

Sentiment Analysis of sentences

Based on CNN and LSTM

Vivek Reddy Doudagiri, Venugopal Gonela, Ramcharan Varma Golla

*Abstract*—Sentiment analysis is a sub-domain of opinion mining where the analysis is focused on the extraction of emotions and opinions of the people towards a topic from a structured, semi-structured or unstructured textual data. In the recent years, sentiment analysis has emerged as a major research problem in the field of Natural Language Processing. In this project, we try to focus our interest on one special area where Sentiment analysis is critical i.e., movie reviews. Here we consider binary (positive and negative) classification based on Recurrent Neural Networks.

*Keywords—Convolution Neural Network, Opinion Mining, Recurrent Neural Network, Structured data*

# INTRODUCTION

World Wide Web has become the biggest database being fed with data by every internet user around the world. A lot of data in the form of text, images, music that shares people’s ideas, interests and emotional experiences is being exchanged. This makes Opinion Mining or Sentiment analysis an important area of research.

The database has been growing at an exponential rate giving rise to new challenges. Opinions which are being expressed in the form of reviews provide an opportunity for new explorations to find collective likes and dislikes of the users. One such domain of reviews is the domain of movie reviews which affects everyone from audience, film critics to the production company. Movie reviews posted in user forum or blogs are usually informal and unstructured. The reviews convey the true emotion of a person about the topic of interest, in this case it is the emotion towards a movie.

# Related Research

Sentiment analysis of online user generated content is

important for many social media analytics tasks. In the context

of social media, there are several challenges. First, there are huge amounts of data available. Second, messages on social networks are by nature informal and short. Therefore, sentiment analysis is a very challenging task. Xi Ouyang, Cheng Hua Li, Lijun Liu [1] proposed a framework called Word2vec + Convolutional Neural Network (CNN). They first used Word2vec of google to compute word vectors which are input to CNN. An architecture consisting of 3 Convolution layers and Pooling layers is used and the accuracy of the model is improved by considering Parametric Rectified Linear unit is used as the activation function, Normalization and Drop out technology.

Andrew M. Dai and Quoc V. Le from Google have proposed two approaches [3], one of them is to use a Sequence auto encoder and another is to predict what comes next in a sequence. Their experimental result with long short term memory recurrent networks after pretraining using the above proposed models shows that strong performance is achieved in many of the classification tasks.

# Sentiment Analysis

Sentiment analysis aims to determine the attitude of a speaker or a writer with respect to some topic or the overall contextual polarity of a document. The attitude may be their judgment or evaluation, affective state which is the emotional state while writing, or the intended emotional communication which is the emotional effect the writer wishes to have on the reader. Sentiment is defined as any kind of emotion and in the context of sentiment analysis is the opinion that is being expressed in the form of text or speech.

Sentiment Analysis is a technology that will be very important in the next few years. With opinion mining, we can distinguish poor content from high quality content. With the technologies, available we can know if a movie has more good or bad opinion and find the reasons why those opinions are positive or negative. Much of the early research in this field was centered around product reviews, such as reviews on different products on Amazon.com, defining sentiments as positive, negative, or neutral. Most sentiment analysis studies are now focused on social media sources such as IMDB, Twitter and Facebook, requiring the approaches be tailored to serve the rising demand of opinions in the form of text. Furthermore, performing the phrase-level analysis of movie reviews proves to be a challenging task.

# Neural Network

As modern computers become ever more powerful, scientists continue to be challenged to use machines effectively for tasks that are relatively simple for humans. More experience allows humans to refine responses and improve their performance. Although eventually, we may be able to describe rules by which we can make such decisions, these do not necessarily reflect the actual process we use. Yet another common human activity is trying to achieve a goal that involves maximizing a resource while satisfying certain constraints. Each of these types of problems illustrates tasks for which computer

solutions may be sought.

An artificial Neural Network is an information processing system that has certain performance characteristics in common with biological neural networks. They have been developed as generalization of mathematical models of human cognition of neural biology based on assumptions that information processing occurs at many simple elements called neurons, signals are passed between neurons over connection links, each connection link has an association weight, each neuron applies an activation function to its net input to determine output signal.

A neural network is characterized by its architecture, training or learning algorithm and its activation function. Most neural networks use ReLU or tanh as activation function.

Neural networks are sometimes described in terms of their depth, including how many layers they have between input and output, or the model's so-called hidden layers. They can also be described by the number of hidden nodes the model has or in terms of how many inputs and outputs each node has. Variations on the classic neural-network design allow various forms of forward and backward propagation of information among tiers.

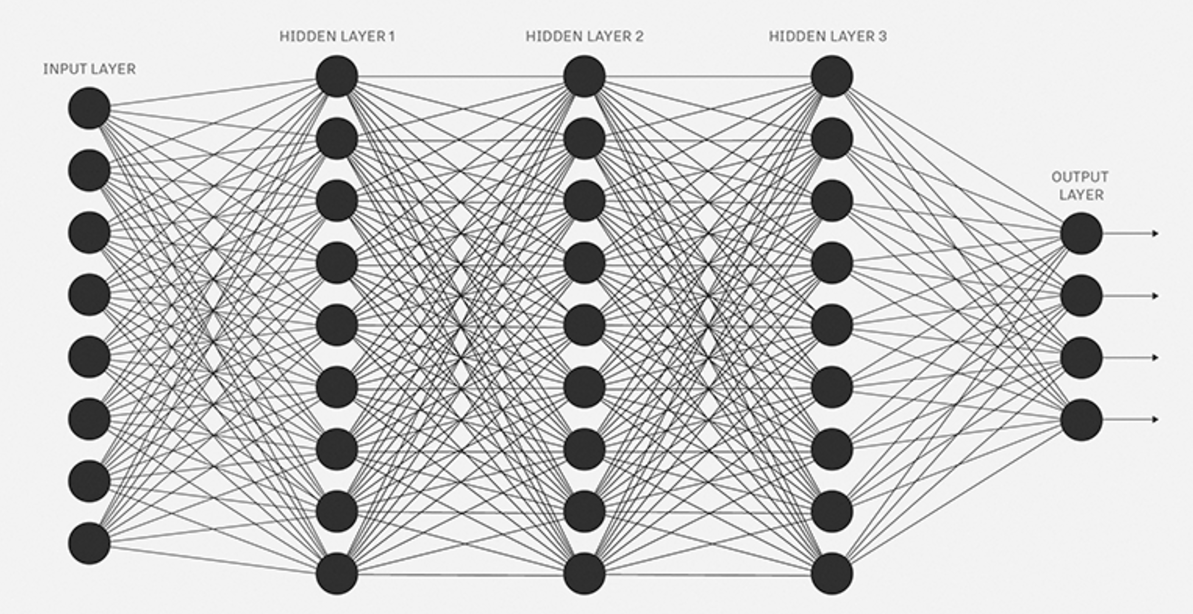


Fig 1: Neural Network with 3-hidden layers

## Convolution Neural Networks (CNN)

Convolutional neural network (ConvNet) also known as shift invariant or space invariant artificial neural network is a type of [feed-forward](https://en.wikipedia.org/wiki/Feedforward_neural_network) [artificial neural network](https://en.wikipedia.org/wiki/Artificial_neural_network) that are made up of neurons that have learnable weights and biases. Each neuron receives some inputs, performs a dot product and optionally follows it with a non-linearity. The whole network still expresses a single differentiable score function: from the raw data on one end to class scores at the other. And they still have a loss function (e.g. SVM/Softmax) on the last (fully-connected) layer.

A simple ConvNet is a sequence of layers, and every layer of a ConvNet transforms one volume of activations to another through a differentiable function. We use three main types of layers to build ConvNet architectures: **Convolutional Layer, Pooling Layer**, and Fully Connected Layer.

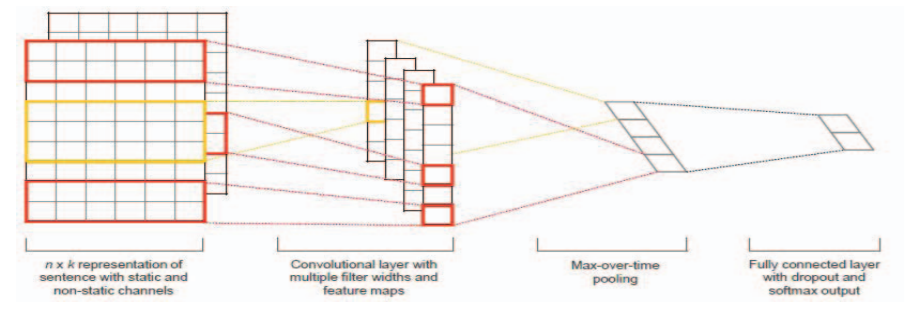


Fig 2: Convolutional Neural Network

## Recurrent Neural Network (RNN)

A recurrent neural network (RNN) is a class of [artificial neural network](https://en.wikipedia.org/wiki/Artificial_neural_network) where connections between units form a [directed cycle](https://en.wikipedia.org/wiki/Directed_cycle). This creates an internal state of the network which allows it to exhibit dynamic temporal behavior. Unlike feedforward neural networks, RNNs can use their internal memory to process arbitrary sequences of inputs. This makes them applicable to tasks such as unsegmented connected handwriting recognition or speech recognition.

The idea behind RNNs is to make use of sequential information. In a traditional neural network, we assume that all inputs (and outputs) are independent of each other. But for many tasks that’s a very bad idea. If you want to predict the next word in a sentence you better know which words came before it. RNNs are called recurrent because they perform the same task for every element of a sequence, with the output being depended on the previous computations. In theory RNNs can make use of information in arbitrarily long sequences, but in practice they are limited to looking back only a few steps.

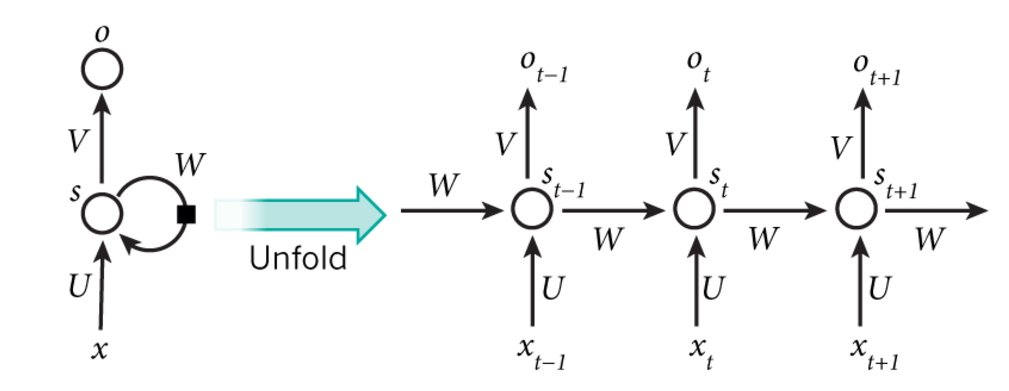


Fig 3: Typical RNN and foldings in time

## LSTM

## The LSTM addresses the problem of “vanishing gradient”

by replacing the self-connected hidden units with memory blocks as illustrated in Fig. 4. The memory units enable the network to be aware of when to learn new information and when to forget old information. Our LSTM implementation is standard and has input gates, forget gates, and output gates

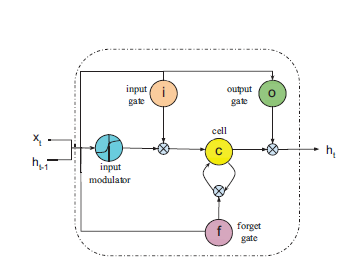


Fig 4: Long Short Term Memory Recurrent Network

# Literature Survey

## Word Embedding

Word embedding is a technique of representing a word in terms of numerical vectors usually of lower dimension compared to vocabulary. The words that have similar meaning can be made to correspond to close vectors to obtain meaningful results. Word Embedding plays a crucial role in all natural language processing applications like POS tagging. There are two embedding’s that are commonly used, Word2Vec and GloVe.

*Word2Vec:*

Word2Vec by google provides an efficient implementation of the continuous bag-of-words and skip-gram architectures for computing vector representations of words. The word2vec tool takes a text corpus as input and produces the word vectors as output. It first constructs a vocabulary from the training text data and then learns vector representation of words. The resulting word vector file can be used as features in many applications.

*GloVe:*

GloVe is an unsupervised learning algorithm for obtaining vector representations for words. Training is performed on aggregated global word-word co-occurrence statistics from a corpus, and the resulting representations showcase interesting linear substructures of the word vector space. It is a count based model.

The GloVe model is trained on the non-zero entries of a global word-word co-occurrence matrix, which tabulates how frequently words co-occur with one another in given corpus. Populating this matrix requires a single pass through the entire corpus. This is computationally expensive as the size of data grows, but it is a one-time cost. Subsequent training iterations are much faster because the number of non-zero matrix entries is typically much smaller than the total number of words in the corpus.

## Drop out

Over fitting is a serious problem in machine learning models. Large networks are also slow to use, making it difficult to deal with overfitting by combining the predictions of many different large neural nets at test time. Deep neural networks contain multiple non-linear hidden layers and this makes them very expressive models that can learn very complicated relationships between their inputs and outputs. With limited training data, however, many of these complicated relationships will be the result of sampling noise, so they will exist in the training set but not in real test data even if it is drawn from the same distribution. This leads to overfitting and many methods have been developed for reducing it. Dropout is a regularization technique in neural networks which avoids model from overfitting by preventing co-adaptation on training data.

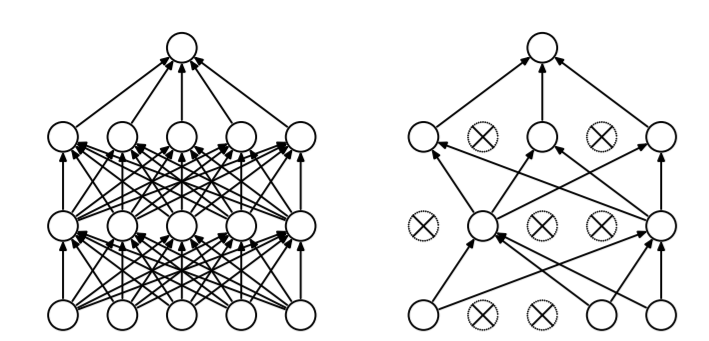


Fig 5: a) Standard Neural Network b) Neural net after Drop Out

## Activation function

In computational networks, the activation function of a node defines the output of that node given an input or set of inputs. The role of activation function in a neural network is to make it non-linear. There are many Activations that are in use. Some of them are Identity function, Tanh, Rectified Linear Unit (ReLU), Soft plus, Sinc, sigmoid etc.

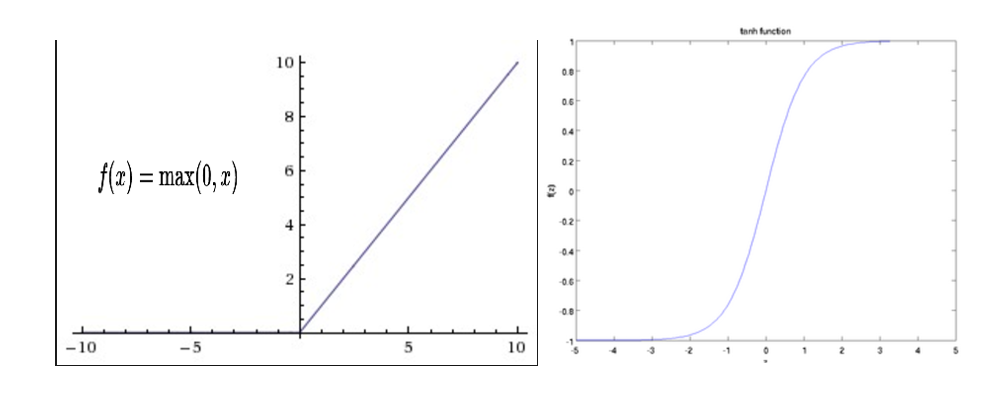


Fig 6: a) ReLU function b) Tanh function

## ADAM optimizer

ADAM optimization algorithm [4] for first-order gradient-based optimization of stochastic objective functions, based on adaptive estimates of lower-order moments. The method is straightforward to implement, is computationally efficient, has little memory requirements, is invariant to diagonal rescaling of the gradients, and is well suited for problems that are large in terms of data. The method is also appropriate for non-stationary objectives and problems with very noisy and/or sparse gradients. The hyper-parameters have intuitive interpretations and typically require little tuning. The method computes individual adaptive learning rates for different parameters from estimates of first and second moments of the gradients.

# DataSet

The dataset we used is the [Movie Review data from Rotten Tomatoes](http://www.cs.cornell.edu/people/pabo/movie-review-data/). The data set is originally collected by Pang and Lee [5]. It contains 10,662 example review sentences, half positive and half negative. The dataset has a vocabulary of size around 20k. Since this data set is small we’re likely to overfit with a powerful model. Using Random Sub-sampling the whole data set is divided into validation set, Training Set and Test Set. 10% of the data is taken as a validation set, 70 % for training and 20% for testing.

# Experiment

# Conclusion

Sentiment Analysis is a challenging task. In our project, we considered two different models CNN and RNN to classify the given movie review dataset. By properly tuning the parameters the maximum accuracy in each model is observed.

# References

[1] Ouyang Xi, Cheng Hua Li, Lijun Liu, “Sentiment Analysis

Using Convolution Neural Networks”, *In proceedings of*

*the International Conference on Computing and*

*Information technology;* *Ubiquitous Computing and*

*Communications; Dependable, Autonomic and Secure*

*Computing; Pervasive Intelligence and Computing,*

Pages: 2539-2364, 2015.

[2] Yoon Kim, “Convolutional Neural Networks for Sentence

Classification”, WildML.com, Referenced on 10th march,

2017.

[3] Andrew M. Dai, Quoc V. Le, “Semi-Supervised Sequence

Learning”, *In proceedings of the Neural Information*

*Processing Systems Conference.*

[4] Diederik P. Kingma, Jimmy Lei Ba, “ADAM: A Method

For Stochastic Optimization”, *In proceedings of*

*International Conference on Learning Perspective (ICLR),*

2015.

[5] Pang B, Lee L, “Seeing Stars: Exploiting Class Reation-

Ships For Sentiment Categorization with respect to rating

scales”, *In proceedings of the 43rd Annual Meeting on*

*Association for Computational Linguistics,* PP. 115-124,

2005.

1. [↑](#footnote-ref-1)