Sentiment Analysis From Text

by Team 8

Convolutional Neural Networks (CNN)

CNN are a specific type of neural network that can be useful for tasks such as classification or segmentation. Convolutional layers use only certain connections from previous layer .

This method causes the layer to gain more of an understanding of the general view of the inputs.

Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) is a type of Recurrent Neural Network (RNN). RNN neurons have a connection to the previous neuron state . RNNs are particularly beneficial to data that is sequential or that can value a contextual view; therefore, RNNs can be impressive in text classification .

LSTMs are a form of RNNs where newer information in the neurons is more critical than older information .They are useful on sequential text data.

DESCRIPTION OF DATASET

The ACL Internet Movie Database (IMDb) dataset used was created in for learning word vectors. The dataset consists of (50,000) reviews are paired with a label of 0 or 1 to represent negative and positive sentiment, respectively. These labels were linearly mapped from the IMDb's star rating system where reviewers can rate a movie a certain number of stars from 1 to 10. The reviews with labels are split in half; each set has 12,500 positive reviews and 12,500 negative reviews to keep the data balanced

The Proposed Method

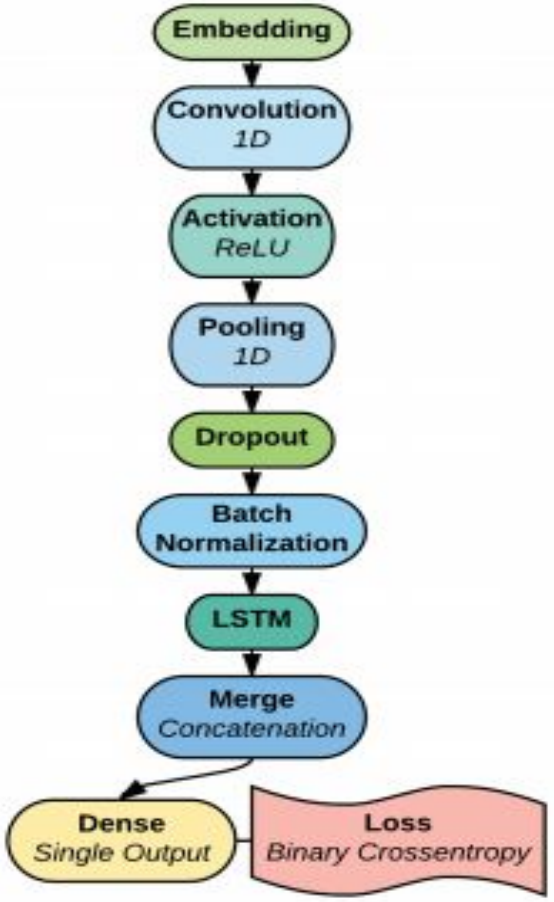


Figure 3. Diagram of basic structure of CNN based LSTM network. We use this in our experiments as a baseline model to compare with our novel model.

Convolution

The output of the embedding layer is given to each branch. Each branch starts with a 1-dimensional convolution layer of kernel size c specific to that branch. The kernel during 1-dimensional convolution is of shape kernel size by embedding size (c x e). The kernel will then perform convolution on whole words instead of the typical 2- dimensional convolution that would filter partial widths and cut words into pieces. The purpose of the CNN layer is to view word combinations of the kernel size c. The result is an understanding of words when used with other words.

Activation

Each branch applies a rectified linear unit (ReLU) activation of the CNN layer’s output; this layer replaces any negative outputs with zero. The ReLU layer is used in order to introduce non-linearity into the network. The output of this layer is the same shape as the input shape.

Max Pooling

Each branch undergoes 1-dimensional max pooling following ReLU activation; this layer converts each kernel size of the input into a single output of the maximum observed number. The result is a reduced, down-sampled version of the input. The purpose of the layer is to reduce overfitting, while allowing for further processing. This technique allows pooling to be applied with the understanding that data is composed of whole words. The output of this layer is a reduction in height according to kernel size p (input height ÷ p).

Dropout

After max pooling, each branch goes through a dropout layer; this layer randomly sets a portion of the inputs to 0. The dropout is applied to specified d fraction of the inputs. This layer serves to generalize the network to not focus on specific pieces of input. The output shape is equivalent to the input shape.

LSTM

The final layer for each branch is a LSTM layer with a specified number of units l. The LSTM is used because of the nature of sequential data. The layer’s persistence allows knowledge of previous input (convoluted word combinations) to influence subsequent input. The output has a length of the number of units l.

Dense

The layer is followed by a simple sigmoid activation function to conform the output between 0 and 1. The final yield is a single output.

Loss Function

The network is compiled with a binary cross entropy loss function, this loss calculates loss with two classes 0 and 1.

0 represents negative sentiment and 1 represents positive sentiment. The loss is calculated on the single and final output of the dense layer.

Optimizer

The network is also compiled with an optimizer; Adam was the optimizer used for testing during the experiments. The optimizer was used with varying learning rates and learning rate decay parameters.

EXPERIMENTAL SETUP

The IMDb review sentiment dataset was used for all experiments. The dataset was preprocessed to a dictionary size of 5,000 words with a zero-padded maximum sequence of 500 words per a review; anymore data became insignificant to the networks objective.

Embedding

An embedding size of 32 best fit the dataset. Attempts to use GoogleNews word2vec pre-trained embeddings and Wikipedia fastText pre-trained embeddings (as untrainable word embeddings) proved to have no positive effect. Therefore we do not include it in our proposed model

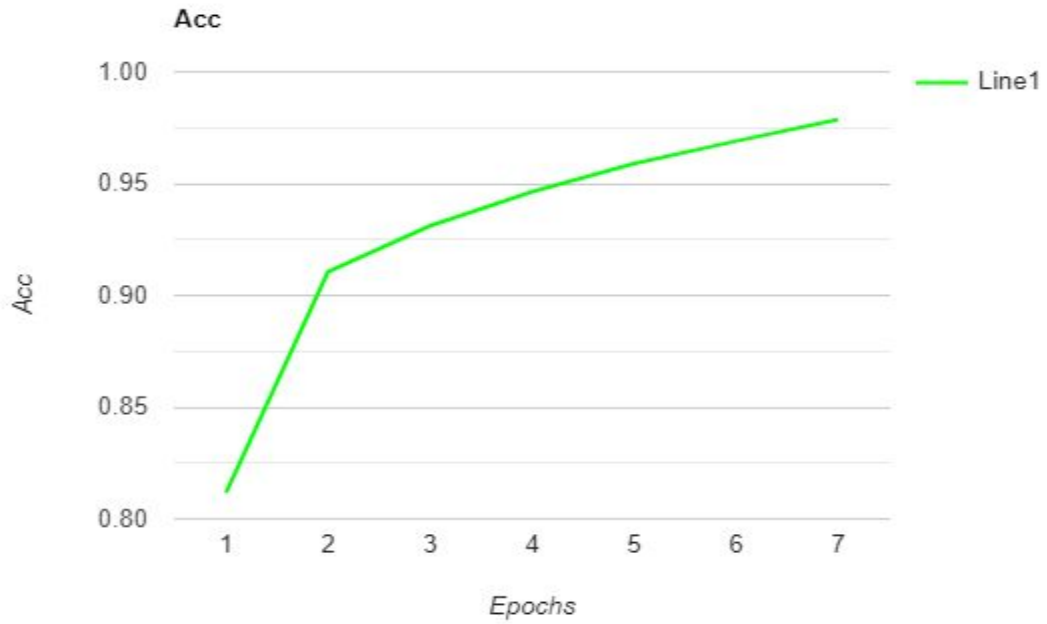
Max Pooling & Dropout

Although max pooling assisted with the chief issue of overfitting, large pooling sizes proved to decrease accuracy. The optimum kernel size during the experiments was 2; reducing the height of the input by half.

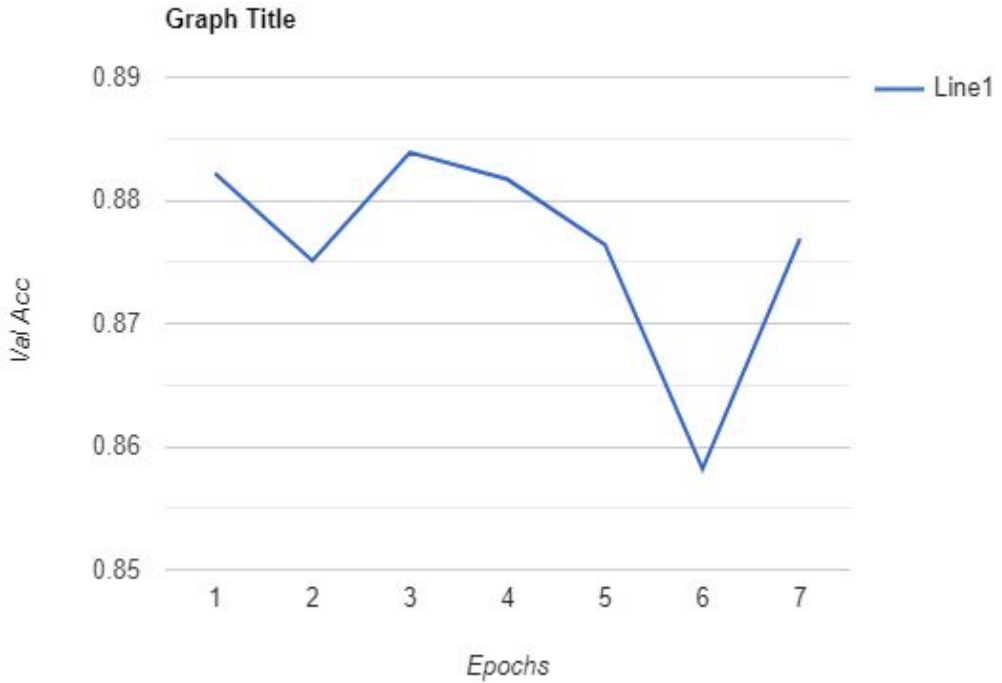
The dropout layer was recognized to be the best option to reduce overfitting.

RESULTS AND ANALYSIS

Accuracy :



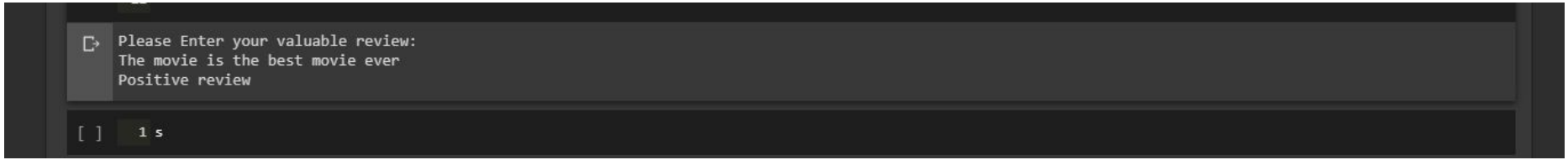
Overfitting :



Result


Please Enter your valuable review:
1 s


Please Enter your valuable review:
The movie is the best movie ever
1 s


Please Enter your valuable review:
The movie is the best movie ever
Positive review
1 s