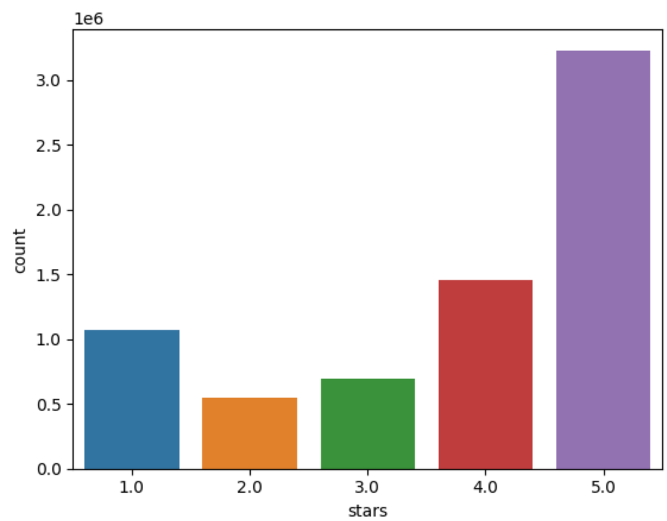
# Homework 2 Sentiment Analysis Report

*Prepared by* - **Ehsanul Kabir**

## Data Description

In this sentiment analysis project, I employed the Yelp Sentiment review dataset as my primary data source. This dataset comprises multiple columns, with our main focus being on two of them: the "text" column, which contains the submitted reviews, and the "stars" column that reflects the associated ratings. Following this, I have generated a plot that exhibits the number of samples for each rating category. Additionally, I have selected one representative record from each of the distinct rating groups.




| index | text | stars |
|---|---|---|
| 5775831 | They charge over 400$ just to recharge AC with freon. Update: Someone with Meyers contacted me and told me I was mistaken in my claim that they charged me 400$ solely for freon. They then requested my information to look up the ticket. After looking up the ticket they changed their tune, implicitly acknowledging that THEY were mistaken. So to recap, I make a claim. They deny it. Look up the information and see it's true, then give a canned "we provided you with all your options" BS answer, tacitly acknowledging their claim that I wasn't charged that much was false. | 1.0 |
| 5266345 | I asked for the Strawberry Margarita Pedicure. And he got strawberry splatter on my pants. I asked if they would pay for the dry cleaning or replace my pants if the Strawberry didn't come out. They said they would have to talk to the boss. They offered no restitution. I made it clear I wanted my nails to match my toenails. I picked out a color and asked if they had a matching shellac color for my nails. He assured me they did. He started with the first color and it didn't match but I had to stop him from continuing. On the fourth color we found one close but not the same. He dropped a cube filer on the ground and went to use it on my nails. I had to stop him and ask for a new one. He kindly obliged. The location was convenient, I got my car cleaned while I got my nails done. And the gentleman was very nice. However, it doesn't make up for the missteps. | 2.0 |
| 4365205 | Thanks to my fellow Yelpers, we headed to Stillwaters while visiting my family. It was somewhat of an odd visit. We were seated near a large barrel that was full of ice and glass bottles with table water. The spout on the barrel continually dripped behind us and splattered around our feet. Annoying! The waitress took a long time to come to our table, and when she did she asked if we were ready to order. No! We didn't even order any drinks. Well, we received our drinks, put in our order and the waitress was basically - absent. The nearby tables received plenty of attention. The food and drinks were mediocre at best, but the outside seating was awesome. | 3.0 |
| 745166 | Food - four stars Ambience - two stars Service - three stars Value for money - five stars Overall - four stars This is a hole in the wall place with yummy Indonesian food. I had the yellow rice with tempeh, fried tofu, been sprouts, and other vegetables. On top of that I tried both the green and red 'sambal', spicy sauce. If you are in need of an affordable fix of good Indonesian food, this is your place. There's not much ambience so not great for a date night. They also have a wide range of other typical Indonesian food, including fried chicken, fried fish, curries, and so on. I have lived in Indonesia for 15 years, and this is a typical tasty meal. | 4.0 |
| 3516304 | Oh god this chicken was good... it so wasn't what you would expect at a liquor store. I mean i would have NEVER gone to a liquor store for chicken. I stumbled upon this place on a drunken walk home from bourbon st. I was staying at the renaissance up the block... And i thought F+++ it! Wow... so good. I went back the next night... and the next!!! For like 9 bucks I got a 3 piece meal, a gatorade, and a little pie. dude! The batter is crispy. crunchy. full of flavor. The chicken meat was soft and moist. It burst with juices as i bit into it. insane good. :) | 5.0 |

## Subset Selection

In order to assign sentiment values to each rating, I devised the subsequent strategy. I chose a limited subset of the dataset for conducting a small-scale sentiment analysis. The sentiment mapping is as follows:

- Ratings of 1 and 2 stars are mapped to the "Negative" sentiment category,
- A rating of 3 stars is designated as "Neutral" sentiment,

- Ratings of 4 and 5 stars are categorized under the "Positive" sentiment.

For this analysis, I extracted 10,000 records from each of the three sentiment categories, resulting in a total of 30,000 records.

# Classification task

In the beginning, I planned the classification task to differentiate among all three sentiment groups. However, I later discovered that the classifiers were not performing well on the neutral class, likely because of its ambiguous nature. Consequently, I adjusted the classification task to focus on identifying positive and negative sentiments in some of the experiments.

# Data Preprocessing steps

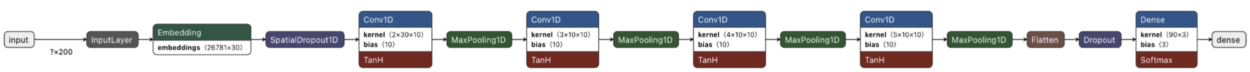
- I employed the *simple_preprocess* function from *gensim.utils* to convert words to lowercase and eliminate words with a length shorter than 3 characters or longer than 15 characters.
- For stemming purposes, I utilized the *PorterStemmer* from the *gensim.parsing.porter* module.
- Lastly, to tokenize each word into an id/token, I made use of the *tokenizer* available in the *nltk* library.
- Each sample sentence was trimmed or padded to make it long as the max sentence length (200)

# CNN Model Architecture

- The embedding layer transforms input words into continuous vectors with a fixed size of 30, capturing semantic relationships between words, while the spatial dropout layer helps with regularization by dropping entire 1D feature maps.
- The 1D convolutional layers learn to recognize local patterns in the input using different numbers of filters and kernel sizes, with the max-pooling layers reducing spatial dimensions and capturing the most important features from the previous layers.
- The flatten layer reshapes the input into a one-dimensional array suitable for the fully connected layers, and the dropout layer aids in regularization by randomly setting a fraction of input units to 0 during training.
- The dense output layer has 3 neurons, representing the three sentiment classes, and utilizes softmax activation to generate probability distributions for each class, allowing the model to predict the sentiment of a given input text.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding (Embedding)       (None, 200, 30)           803430

 spatial_dropout1d (SpatialD  (None, 200, 30)          0
 ropout1D)

 conv1d (Conv1D)             (None, 199, 10)           610

 max_pooling1d (MaxPooling1D  (None, 99, 10)           0
 )

 conv1d_1 (Conv1D)           (None, 97, 10)            310

 max_pooling1d_1 (MaxPooling  (None, 48, 10)           0
 1D)

 conv1d_2 (Conv1D)           (None, 45, 10)            410

 max_pooling1d_2 (MaxPooling  (None, 22, 10)           0
 1D)

 conv1d_3 (Conv1D)           (None, 18, 10)            510

 max_pooling1d_3 (MaxPooling  (None, 9, 10)            0
 1D)

 flatten (Flatten)           (None, 90)                0

 dropout (Dropout)           (None, 90)                0

 dense (Dense)               (None, 3)                 273

=================================================================
Total params: 805,543
Trainable params: 805,543
Non-trainable params: 0
```

Keras Model Summary

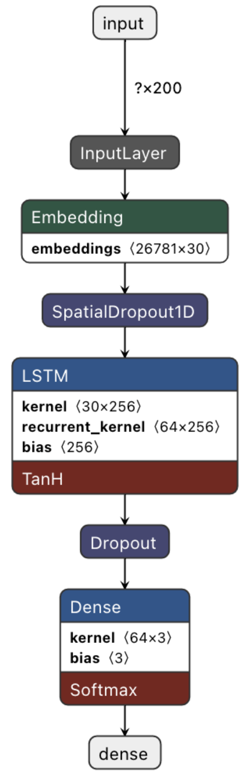



Netron Visualization

# LSTM Architecture

- The embedding layer transforms input words into continuous vectors with a fixed size of 30, capturing semantic relationships between words, while the spatial dropout layer helps with regularization by dropping entire 1D feature maps.
- The LSTM layer, with 64 hidden units, captures long-range dependencies in the input sequence, handling issues like vanishing gradients, followed by a dropout layer for regularization, and a dense output layer with 3 neurons to predict the sentiment class using softmax activation.

```
 Layer (type)                Output Shape              Param #
=================================================================
 embedding (Embedding)       (None, 200, 30)           803430

 spatial_dropout1d (SpatialD  (None, 200, 30)          0
 ropout1D)

 lstm (LSTM)                 (None, 64)                24320

 dropout (Dropout)           (None, 64)                0

 dense (Dense)               (None, 3)                 195

=================================================================
Total params: 827,945
Trainable params: 827,945
Non-trainable params: 0
_________________________________________________________________
None
```



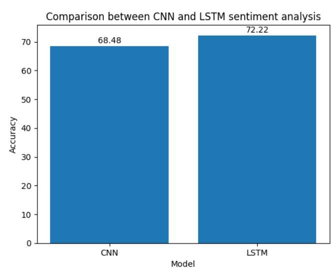

### Default Hyperparameters

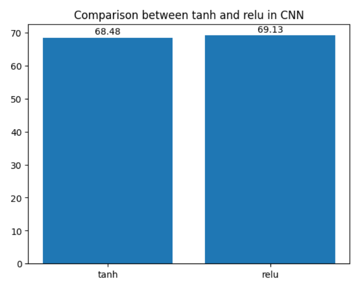
| Name | Value |
|---|---:|
| Embedding Dimension | 30 |
| Max sentence length | 200 |
| Activation | tanh |
| Optimizer | adam |
| Number of filters in CNN | 10x4 |
| LSTM number of layers | 1 |
| LSTM number of memory cells | 64 |
| Number of epochs | 20 |
| Training Batch Size | 32 |

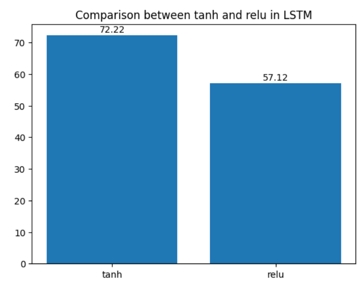
# CNN vs LSTM Comparison

| | CNN | LSTM |
|---|---:|---:|
| Trainable Params Count | 805,543 | 827,945 |
| CNN/LSTM Specific Params | 1840 | 24320 |
| Training Time Per Epoch (Mean) | 15 sec | 260 sec |

# Performance comparison (on 20% withheld test samples)



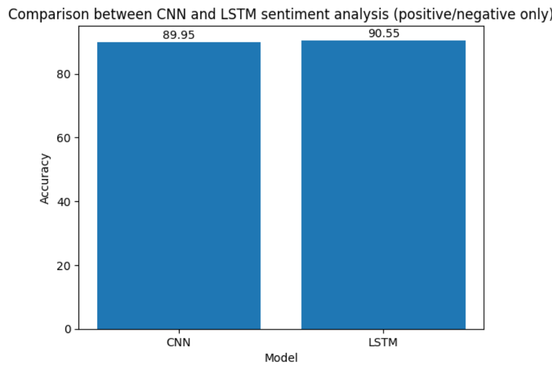



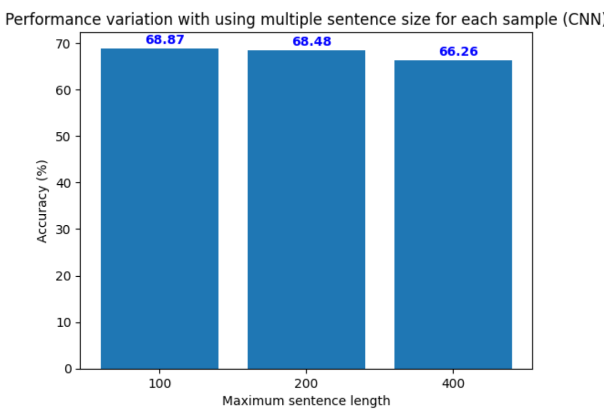

The leftmost image presents a comparison between the performance of LSTM and CNN models. In this case, LSTM slightly surpasses the performance of CNN. The middle image illustrates a comparison between the tanh and ReLU activation functions when used in a CNN. Here, ReLU demonstrates a marginally better performance. Lastly, the rightmost image showcases a comparison between tanh and ReLU when applied within an LSTM model. In this particular instance, the tanh activation function significantly outperforms ReLU.

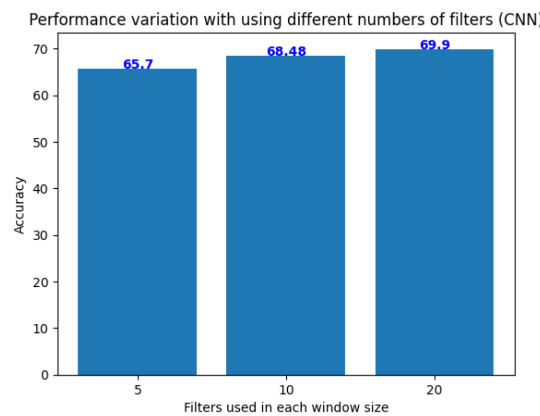
| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.66 | 0.81 | 0.73 | 1964 |
| 1 | 0.62 | 0.51 | 0.56 | 2056 |
| 2 | 0.77 | 0.74 | 0.76 | 1980 |
| accuracy | | | 0.68 | 6000 |
| macro avg | 0.68 | 0.69 | 0.68 | 6000 |
| weighted avg | 0.68 | 0.68 | 0.68 | 6000 |

The classification report for the CNN model utilizing the tanh activation function indicates that the f1-score is notably lower for class 1, which corresponds to the neutral sentiment. The reason behind this diminished performance is the inherent ambiguity of the neutral category, specifically the 3-star ratings. These ratings can be perceived as negative by some individuals, while others may consider them moderate, thus causing the classifier to struggle in accurately identifying the neutral sentiment.

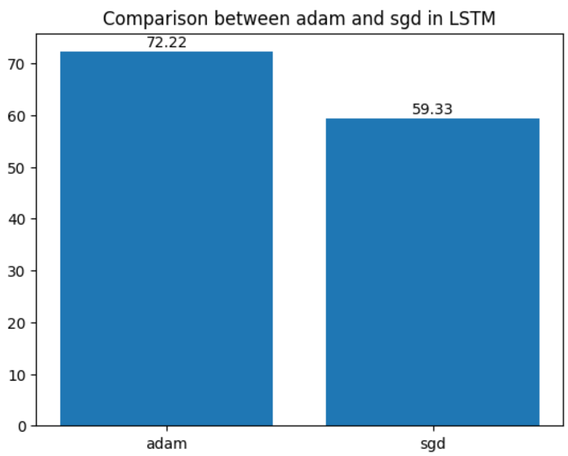


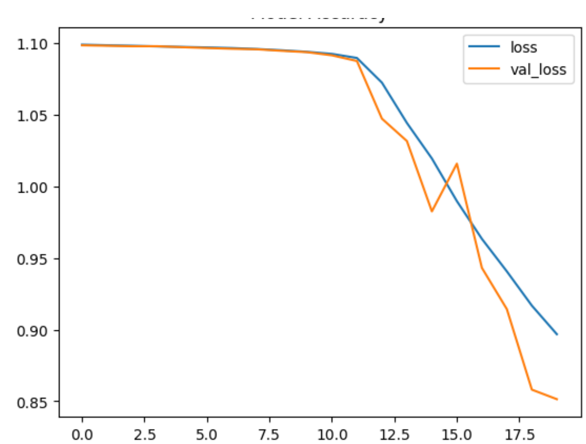

The displayed image illustrates the performance of both CNN and LSTM models in the context of the positive/negative sentiment analysis experiment. In this particular situation, both models exhibit a significant improvement in their performance. The LSTM model achieves a slightly superior outcome compared to the CNN model.




The left image presents the performance fluctuations when utilizing sentences of varying lengths. As evident from the graph, employing shorter sentences proves to be more effective. On the other hand, the right image demonstrates the impact of using different numbers of filters in a CNN model on performance. This illustration reveals that incorporating a greater quantity of filters leads to improved accuracy.




The left image illustrates that using the stochastic gradient descent (SGD) training method for LSTM results in suboptimal outcomes within 20 epochs. This may be attributed to the slower convergence typically associated with SGD. Meanwhile, the right image displays the training loss and validation loss during the SGD training process, highlighting that the loss only begins to decrease after the 10th iteration.

# Conclusion

- Through these experiments, I gained valuable insights into the various data preprocessing steps required when working with textual data.
- Additionally, I acquired considerable knowledge about exploring different architectures to determine the most suitable one for the given dataset.
- Furthermore, I observed several interesting behaviors concerning the Yelp review sentiment analysis data, such as the ambiguous nature of the neutral reviews