# Movie Reviews Sentiment Analysis

**Ratna Pathak**
University Id: S220046535
Data Science (Msc)
ratna.pathak@city.ac.uk

## 1 Problem statement and motivation

The primary goal of this project is to perform sentiment analysis on the IMDB Data set of 50K Movie Reviews, which is a widely used benchmark data set for sentiment analysis. The problem being addressed is to improve the classification of movie reviews based on the sentiment expressed in the text. The main impetus for this project is to create a dependable and accurate method for analyzing large volumes of textual data to gain insights into customer sentiments towards movies.

Sentiment analysis is a task that involves the identification of subjective information in text, such as opinions, attitudes, emotions and sentiments. It has many applications, including social media monitoring, customer feedback analysis, and market research. In today's world, the importance of sentiment analysis is ever-increasing due to the proliferation of textual data from sources such as social media and online reviews.

Developing a model that can accurately classify movie online reviews in this data set can contribute significantly to this field.

Furthermore, According to Chen et al. (2019b) the project's results could benefit movie studios, producers, and distributors, as the proposed model can offer insights into customer sentiments and preferences. Additionally, online platforms such as IMDB could leverage this model to improve their recommendation systems and offer tailored content to the users. Sentiment analysis models have been used to determine the success of new products (Qazi and Qazi, 2019), track customer opinions on social media (Chen et al., 2019a), and improve customer service response times (Altalhi and Alrubaian, 2018)

## 2 Research hypothesis

The research hypothesis for sentiment analysis on IMDb are

1. By incorporating feature extraction techniques such as TF-IDF or Bag-of-Words into a machine learning model, the performance of the model could be enhanced.

2. In comparison to the above method, neural network algorithms can enhance the accuracy and efficiency of sentiment analysis on movie reviews.

Traditional methods like those above, do not consider the relationship between words or the context of the text. They rely solely on the occurrence or frequency of individual words in the text. Neural networks with word embeddings can capture the semantics and context of the text, leading to improved accuracy and efficiency in sentiment analysis on movie reviews. A comparison of neural network models with and without embeddings will be conducted.

3. Leveraging the contextual understanding BERT Transformer, a state-of-the-art pre-trained language model that has shown exceptional performance in various natural language processing tasks. BERT has the ability to capture complex relationships between words and understand the context of each word, allowing for a more accurate analysis of the sentiment in text. The use of BERT in sentiment analysis on IMDb reviews can significantly improve performance compared to traditional methods and neural network algorithms.

## 3 Related work and background

In recent years, there has been growing research using deep learning techniques for sentiment analysis, The hypothesis above is derived from reading these research papers:

1. Dang et al. (2018) paper titled "Sentiment Analysis Based on Deep Learning: A Comparative Study" provides an overview of the latest studies that have used deep learning models to solve sentiment analysis problems. The authors review various studies that have

used models based on term frequency-inverse document frequency (TF-IDF) and word embedding on different datasets. The authors suggest that deep learning models can be a promising solution to overcome these challenges.

2. Poornima and Priya (2020) compared the efficacy of term frequency to find the sentiment polarity of the sentence. From the results, it is inferred that logistic regression has achieved the greatest accuracy when used with n-gram and bi-gram models.

3. Naeem et al. (2017) optimized machine learning algorithms and concluded that term frequency-inverse document frequency performed better than other methods, but their results were still lower than those achieved by neural networks.

4. Word embedding with machine learning and neural networks has exhibited promising outcomes in sentiment analysis on IMDb movie reviews.Singhal and Bhattacharyya (2016) provides an overview of the various techniques used for feature extraction, including bag-of-words and word embeddings.

   Qaisar et al. (2019) used an LSTM model for sentiment analysis of movie reviews, achieving an accuracy of 89.4%. Similarly, Naeem et al. (2019) proposed a hybrid model combining LSTM and a convolutional neural network (CNN) for sentiment analysis. They achieved an accuracy of 92.8% on the movie review dataset. Gupta et al. (2017) combines sentiment analysis with semantic analysis to better capture the complex interplay between emotions and language.

5. Qaisar et al. (2019) and Singh et al. (2019) utilized LSTM and different structures of neural networks, respectively, to achieve high accuracy in sentiment analysis on IMDb movie reviews.

6. Wang et al. (2016) introduced a novel sentiment analysis approach that combines an attention mechanism with LSTM to perform aspect-level sentiment classification. The results showed that the proposed model outperforms several state-of-the-art models, including those based on TF-IDF and SVM. The authors conclude that their attention-based LSTM model is effective for aspect-level sentiment analysis and can help improve the performance of sentiment analysis systems.

7. As Nkhata (2021) explores the effectiveness of BERT for sentiment analysis on the movie review dataset. The study shows that BERT significantly improves the accuracy of sentiment analysis compared to other traditional methods, so we will try to see the performance achieved by the BERT transformer.

### 3.1 Accomplishments

For this coursework, the selected task is "text classification". However, the dataset chosen at the time of the proposal was found to be noisy. Therefore, it was replaced with a better dataset that has defined labels by expert annotators. The list of planned tasks is as follows:

1. Task: Data cleaning and preprocessing:– Completed.

2. Task: Feature Engineering:- Completed.

3. Task: Build and train a simple baseline model using feature engineering with a traditional machine learning classifier and examine its performance:-
   Completed. (Logistic Regression model with TF-IDF)

4. Task: Build and train some advanced deep learning models with feature engineering, with and without word Embedding and examine their performance:- Completed.
   RNN was implemented

5. Task: Model Evaluation using metrics such as accuracy, precision, recall, and F1 score. – Completed. The performance was evaluated as specified, and in addition, manual Error analysis was done.

6. Task: Hyperparameter Tuning of the selected algorithm: Completed.

7. Task: Moreover, if time permits, transfer learning, which involves pre-training a neural network on a vast corpus of text and then fine-tuning it on a smaller, labelled dataset for a specific task, will be applied as it has also demonstrated promise in the context of sentiment analysis tasks.– Not Completed. Due to

time constraints. Instead, BERT Model was implemented.

8. Task: Perform in-depth error analysis to figure out what kinds of examples our approach struggles with - Completed.

## 4 Approach and Methodology

1. Our approach to the IMDb movie review sentiment analysis task involves utilizing advanced neural network models and BERT, a state-of-the-art language model, to predict the sentiment of movie reviews. Our approach aims to address the limitations of traditional models such as logistic regression with TF-IDF vectorization, which fail to capture contextual information and relationships between words.

   Implementing a basic LSTM model captures the temporal dependencies in the sequence of words. To improve this model, we will used pre-trained static word embeddings (GloVe) to initialize the embedding layer. This will allow us to leverage the rich semantic information embedded in the pre-trained embeddings and capture the context from both directions of the sequence by implementing a bidirectional LSTM model.

   Finally, fine-tuned a pre-trained BERT model was built on our dataset to learn the patterns and features that are indicative of sentiment in movie reviews. BERT leverages a large corpus of text to learn context-aware representations of words and has shown to be effective in various NLP tasks.

2. The baseline model such as LR with TF-IDF has its own limitations, such as the inability to capture semantic relationships between words and the lack of ability to handle out-of-vocabulary words. Another limitation of the baseline is that they are not able to capture the context of a sentence or document, which can lead to inaccurate predictions in certain cases.

   LSTMs on the other hand are computationally intensive, which can make them slower to train and harder to scale to larger datasets. LSTMs are often referred to as "black box" models because it can be difficult to understand how they are making predictions. This can make it challenging to debug and refine the model

   The performance of BERT can be affected by the quality and quantity of the pre-training data. If the pre-training data is not diverse enough or if the model is not fine-tuned properly, the performance can suffer.

   Another limitation of BERT is that it requires a lot of computational resources, making it difficult to deploy in production environments that have limited resources. The optimal approach may vary depending on factors such as the size and quality of the dataset, the specific task at hand, and the computational resources available.

3. We were able to develop a functional implementation of sentiment analysis for IMDb movie reviews using neural network models and BERT. We conducted manual error analysis to continuously improve our model, and used parsing to find error patterns in the models testing different approaches and checking their performance. We implemented various models and evaluated their accuracy for the sentiment analysis task. Through this iterative process, we were able to improve the accuracy of our sentiment analysis model.

4. Libraries:

   (a) Tensorflow & Keras: These are machine learning libraries that helped in building and training neural networks.
   TensorFlow.keras.preprocessing.sequence: Used for padding sequences.

   (b) Sklearn: Scikit-learn is a machine-learning library that provides tools for data pre-processing, feature selection and evaluation.

   (c) NLTK: The Natural Language Toolkit is a library for working with human language data. It helped with tokenization, stemming, lemmatization, and more.

   (d) Collections: The collections module is a built-in Python library that was used to provide alternatives to built-in abbreviations handling.

   (e) Matplotlib, Seaborn and Plotly: Used for plotting the data distribution visualization.

   (f) Gensim: Gensim is a library for topic modelling and natural language processing, here it was used for GLoVe implementation.

(g) Torch: Used for building deep learning models.

pytorch_pretrained_bert: Used for pre-trained BERT model.

(h) BertTokenizer: Used for tokenizing text into BERT tokens.

(i) TensorDataset: Used for creating a dataset from input tensors.

(j) RandomSampler: Used for random sampling of data.

(k) SequentialSampler: Used for sequential sampling of data.

(l) Adam: Used for optimization of the model.

(m) clip_grad_norm_: Used for gradient clipping.

(n) IPython.display: Used for displaying and clearing the output in Jupyter notebooks.

(o) SpaCy: Provided tools dependency parsing.

5. To understand and perform N-gram analysis for the purpose of TF-IDF, used this blog by Albert Au Yeung's blog post "Generating N-grams,". And, I referred to the Stanford NLP Group's page on Stemming and Lemmatization, found at https://nlp.stanford.edu/IR-book/html/htmledition/stemming-and-lemmatization-1.html, and to better understand the principles behind lemmatization and decide on its use . I drew inspiration from the paper "Attention Is All You Need" by Vaswani et al. (2017) (Vaswani et al., 2017) to implement my bi-directional LSTM model. The Keras documentation on sequential models (https://keras.io/api/models/sequential/) was consulted to develop our LSTM and BiLSTM models." In addition, TensorFlow documentation: https://www.tensorflow.org/guide/keras/rnn for a better understanding of RNNs and implementation using Keras in TensorFlow.

6. Models implemented and files associated: Below 4 models are implemented in the file "these 4 models were implemented in the file "RATNA PATHAK BASELINE AND NN MODELS NLP.ipynb.ipynb".

   Baseline: LR with TFIDF The baseline model was a logistic regression model with term frequency-inverse document frequency (TF-IDF) vectorization.

   LSTM w/o Embedding The second model was a Long Short-Term Memory (LSTM) neural network without any pre-trained word embeddings.

   LSTM with Embedding The third model was an LSTM neural network with pre-trained word embeddings. GloVe embeddings was to initialize the model's embedding layer.

   Bi-direction LSTM, with static embedding: glove The fourth model was a bidirectional LSTM neural network with static GloVe embeddings.

   BERT model The fifth and final model was a BERT (Bidirectional Encoder Representations from Transformers) model. This model utilized pre-trained contextualized embeddings to capture long-range dependencies in the text. The BERT model was implemented in the file "RATNA_BERT.ipynb".

7. During the implementation of the IMDb movie review sentiment analysis some challenges and roadblocks were encountered, including:

   Data preprocessing: The IMDb dataset had duplicate reviews, which had to be removed to prevent data leakage. Additionally, the distribution of word counts was right skewed, with some reviews being extremely long. It was challenging to select a suitable maximum sequence length for the BERT model. A maximum sequence length of 512 tokens was selected, which is the maximum input length that BERT can handle.

   GPU memory constraints: Training neural networks on these large datasets was memory-intensive, especially when using BERT. The limited GPU memory made it challenging to train large models with high batch sizes. To address this challenge, the batch size was reduced, and gradient accumulation was used to simulate larger batch sizes. Additionally, mixed-precision training was utilized, which allows for larger models to be trained with less memory.

Model selection and hyperparameter tuning: There are many neural network architectures and hyperparameters to choose from, which can be overwhelming. It was challenging to select the best model architecture and hyperparameters for the IMDb dataset.

To address this challenge, various neural network architectures were compared, including LSTM, Bi-directional LSTM, and BERT. Due to time and computation constraints, Hyperparameter tuning was performed based on previous research and hit and trial methods but using grid search and random search methods would have provided an optimal performing model.

## 5 Dataset

1. The IMDB Movie Review Dataset comprises 50,000 movie reviews sourced from IMDb. With a balanced number of positive as well as negative sentiments This dataset is extensively used for natural language processing and sentiment analysis research.

2. Specific examples from the dataset that illustrate the sentiment analysis task : Positive Review "The movie is a very good movie.one of the best from Yash raj films.The direction is incredible.The screenplay is brilliant.The story is excellent.It tells about Rahul who is obssed of Kiran his college friend.He is a full blown psycho doing things like talking to his mother on a phone(anyway she died 15 years back) etc.Kiran is engaged to Sunil.Rahul does everything so he can get her.He even trys to kill Sunil but he survives it.He even goes to the place where they are going to their honeymoon.The movie is every nes delight.Shahrukh is superb,Juhi is fairly good,Sunny is average,Anupham is okay and so is Tanvi,Dalip did good.The movie belongs to Srk.The dialogues are brilliant(Shahrukh ones and a lot if not the overacting and comedy)."Jaadu Teri Nazar" and "Tu Mere Samne" are absolutely melodious tracks.!" In this example, the sentiment of the text is positive, and the task of sentiment analysis would aim to classify it as such.

3. The properties of the data which made my task challenging:

   (a) Large Reviews size: The data set has a right-skewed word count, meaning the majority of the data points lie towards the left side of the distribution, while a few data points with high values are present. There are many short reviews with a small number of words, while relatively fewer reviews have a larger number of words.

   (b) Variability: As movie reviews are given by different types of audiences, the data varies greatly differs in terms of grammar, structure, style, and vocabulary, which makes it challenging to generalize models across different domains and contexts.

   (c) Subjectivity: the data express a wide range of opinions, emotions, and attitudes, which makes it challenging to accurately classify and interpret the sentiment of the text.

   (d) Noisiness: Although the dataset is annotated by experts, It still contains a bit of noise, errors, and inconsistencies which affects the data's quality and the accuracy of models trained on it.

   (e) Presence of sarcasm and irony: Sometimes, reviewers use sarcasm and irony in their reviews, which can be difficult to detect using traditional natural language processing techniques.

   (f) Use of slang and informal language: Some reviewers use informal language, slang, short abbreviations or even create new words in their reviews, which can be difficult for machine learning models to understand.

4. Source of the dataset and basic statistics: The IMDB Movie Review Dataset was compiled by Andrew Maas and his colleagues at Stanford University. The dataset contains 50,000 movie reviews, evenly split into 25,000 training and 25,000 testing samples. The size of the dataset is approximately 80 MB, and the average length of a review is around 230 words, with a standard deviation of around 170 words. The number of sentences per review varies from one to several dozen, with an average of around 15 sentences. The dataset is available

for download from the Stanford AI website. https://ai.stanford.edu/ amaas/data/sentiment/

### 5.1 Data Preprocessing

During the preprocessing stage of the dataset, several techniques were employed to enhance the model's performance. Duplicate removal was carried out. Additionally, the distribution of the word count was visualized, and it was observed that the distribution was skewed towards the right. To address this, various preprocessing techniques and modeling strategies were employed, such as padding and selecting a word embedding later, Attention mechanisms were also incorporated into the model to help it focus on the most crucial parts of the input text, particularly for long reviews.

Gram analysis was also conducted to identify which words occur together frequently, and the analysis of grams served as the foundation for creating a TF-IDF model that includes relevant data. Data cleaning was carried out to remove punctuation, HTML, and URLs. Removing abbreviations with actual words was carried out to reduce sentence lengths and transform words to their base form. After cleaning the data, further refinement was carried out by converting words to their base form to ensure that similar words that have different forms are encoded and tokenized similarly. Lemmatization was recommended to reduce words to their root semantic form, which transforms morphological variations. In contrast, stemming was avoided as it does not consider the semantics of the sentence or surrounding words and may generate words not present in the vocabulary. Additionally, Overall, these preprocessing techniques were crucial in ensuring the dataset was ready for modelling, and improvements in model performance were observed.

## 6 Baselines

FOR the task of sentiment analysis of IMDB Movie Review Dataset, A common baseline model is a simple TF-IDF with Logistic Regression Classifier. This model represents each review as a vector of word frequencies and uses a linear classifier, to classify the review as positive or negative. This baseline model is useful because it is simple, logical and easy to implement, and they often perform well on text classification tasks.

It has been provided specific hyperparameters and it has performed exceptionally well on this dataset with an: Training accuracy: 91 % Validation accuracy: 88% and Test accuracy: 88 %

by doing manual error analysis we came to the conclusion for a few further approaches can be applied to improve the classification are: Use of advanced feature engineering techniques: Instead of using TF-IDF, we can try other feature engineering techniques such as word embeddings (e.g., Word2Vec, GloVe) or contextualized word embeddings (e.g., BERT, ELMo, GPT). These methods can capture the semantic and contextual information of the text and may lead to better classification performance. And, Experiment with different classification algorithms: Logistic regression is a popular algorithm for text classification, but we can try other algorithms such as neural networks can improve the classification.

In addition, there are more complex models that will be used to improve performance, such as neural network models like Recurrent Neural Networks (RNNs). These models can capture more complex relationships between words and perform better on the dataset than the simpler baseline models.

## 7 Results, error analysis

Results
Logistic Regression TF-IDF Model: The precision and recall values for both positive and negative sentiments are quite high, which indicates that the model is able to classify sentiments accurately. The overall accuracy of the model is 0.88, which is also good. However, the F1-score is lower compared to the other two models, which means that the model may not perform well in cases where precision and recall are equally important.

BI-DIR-LSTM: The precision and recall values for both classes are similar, which indicates that the model is able to classify sentiments accurately for both classes. The overall accuracy of the model is also 0.88, which is the same as the Logistic Regression TF-IDF model. However, the F1-score is slightly higher, which suggests that this model may perform slightly better than the Logistic Regression TF-IDF model.

BERT: The precision and recall values for both classes are high and quite similar, which indicates that the model is able to classify sentiments accurately for both classes. The overall accuracy of the model is 0.89, which is slightly higher than the other two models. The F1-score is also high, which suggests that this model may perform better than the other two models. But the point which should be taken into account is due to computational limitation for bert, only fraction of training data was taken. Hence there is a lot of possibilities to improve Bi-dir LSTM and BERT model.

Error analysis:
Baseline model manual error analysis: This was done by annotating 20 examples that were not handled correctly by each model. All the wrong examples contain grammatical errors, run-on sentences and misspellings errors. Additionally, some of the examples have very long and unclear sentences that make it difficult to understand their meaning. The lack of clarity and coherence in the sentences may be confusing for the model, resulting in incorrect predictions.

Example Ambiguity in language: The example "suspenseful subtle much much disturbing" may be linguistically ambiguous, making it difficult for the model to correctly classify. The words "subtle" and "disturbing" are opposite in meaning, and the use of "much much" could add further confusion. Feature engineering: The vectorized TIFF features may not have been the most relevant or informative features for this particular task. Different feature engineering techniques, such as using pre-trained word embeddings or extracting more specific linguistic features, may have improved the model's performance. To improve the model from such patterns in error following analysis was done: 1.Use of advanced feature engineering techniques: Instead of using TF-IDF, we can try other feature engineering techniques such as word embeddings (e.g., Word2Vec, GloVe) or contextualized word embeddings (e.g., BERT, ELMo, GPT) can be implimented. These methods can capture the semantic and contextual information of the text and may lead to better classification performance.

2. Experimenting with different classification algorithms: Logistic regression is a popular algorithm for text classification, but other algorithms such as neural networks can improve the classification. We can try these algorithms and see which one works best for your data.

Bidirectional LSTMs and BERT models can capture both forward and backward context, which can help to better understand the meaning of the words in a sentence or document.

Better handling of long sequences: LSTMs and BERT can handle longer sequences of text than logistic regression models with TF-IDF, which may be helpful for tasks that require understanding longer pieces of text.

Error analysis in Bi LSTM: Example "finally film made entertaining black comedy young girls kicking system ass football funny really funny director says places real event real characters extras chose use professional actors presence would introduced notion unless heart made stone b blind scripted film authority almost absurd grand prize berlin 2006 dear get hold"

This sentence is also difficult to understand because of its awkward phrasing. It seems to be discussing a movie that is a black comedy about young girls playing football and rebelling against the system, but the details are unclear. Manual error analysis of some reviews identified syntactic and semantic errors, as well as issues with clarity and coherence, which may have contributed to their misclassification. For example, one review discusses two different movies with unclear details, while another recommends a movie without specifying its title.

BERT: It was difficult to find patterns in misclassified examples and so Spacy parsing was applied. It looked for negation words in the parsed text and then analyze the patterns to see if there is any indication that the model is not able to handle sarcasm. But it was not very clear.

Both approaches performed slightly better than the baseline

Explain what these results mean in the context of your project and how they contribute to the overall goals of your work. These results showed that improving the accuracy just slightly required a large amount of computation and training time, also a lot of factors are important when it comes to text classification. The fine-tuned baseline model performed extremely well.

## 8 Lessons learned and conclusions

After completing the sentiment analysis project on the IMDB Movie Review Dataset, several key observations and conclusions were made:

1. Preprocessing techniques such as tokenization, lemmatization, and data cleaning significantly improve the accuracy of sentiment analysis. Understanding the data and task requirements is important to determine the most appropriate preprocessing techniques.

2. Selecting the appropriate model architecture

is crucial, considering the tradeoff between cost, time, computation, and accuracy. Different models have different strengths and weaknesses, and it is important to choose the one that best fits the task and dataset.

3. Conducting manual error analysis was a useful technique for identifying patterns and syntactic structures that influenced the model's performance and helps implement techniques as per the pattern of error found.

4. The sentiment analysis model achieved competitive results with state-of-the-art models built on the same dataset, with an accuracy of 89%. This is a slight improvement over the baseline models used.

5. However, misclassified reviews still existed, particularly those with sarcasm or subtle irony. These cases require more sophisticated approaches that take into account the context and speaker's intentions.

Overall, the project achieved its objectives of building and improving the sentiment analysis model and provided new knowledge and insights into the challenges and opportunities of sentiment analysis on natural language data. It was concluded that appropriate preprocessing techniques, model architecture, and error analysis can result in highly accurate and effective sentiment analysis models. However, there is still room for improvement, especially in dealing with subtle nuances of language and context. Future research could explore more sophisticated approaches such as contextual dynamic embeddings and transformer models to further enhance the accuracy and robustness of sentiment analysis.

## References


Ayman M Altalhi and Majdi Alrubaian. 2018. A hybrid sentiment analysis approach for arabic customer service tweets. *Journal of King Saud University-Computer and Information Sciences*, 30(2):171–181.

Xiaodong Chen, Qiang Li, and Bin Li. 2019a. Sentiment analysis of social media for stock market prediction. *Information Sciences*, 479:19–31.

Yiran Chen, Xiangnan He, Lishan Nie, Shengchuan Liu, and Zheng Hu. 2019b. Exploiting review contents and user behaviors for online movie recommendation. *Information Sciences*, 501:315–329.

Nhan Cach Dang, María N Moreno-García, and Fernando De la Prieta. 2018. Sentiment analysis based on deep learning: A comparative study. *Sensors*, 18(8):2667.

Udit Gupta, Abhishek Chatterjee, Raghavendra Srikanth, and Puneet Agrawal. 2017. A sentiment-and-semantics-based approach for emotion detection in textual conversations. *arXiv preprint arXiv:1707.06996*.

Muhammad Naeem, Muhammad Farooq, Ali Khawar, and Sher Afzal Khan. 2017. Sentiment analysis of imdb movie reviews using machine learning techniques. *International Journal of Advanced Computer Science and Applications*, 8(4):102–108.

Muhammad Aamir Naeem, Nauman Aslam, Faizan Ahmad Khan, and Tahir Hameed. 2019. Hybrid sentiment analysis using convolutional neural network and long short-term memory. *Cognitive Computation*, 11(6):1012–1025.

Gibson Nkhata. 2021. Movie reviews sentiment analysis using bert. *Journal of Information Technology and Software Engineering*, 11(3):1–7.

A. Poornima and K. Sathiya Priya. 2020. A comparative sentiment analysis of sentence embedding using machine learning techniques. In *2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS)*, pages 493–496.

Syed Muhammad Qaisar, Muhammad Shahzad, Syed Hassan Bashir, Ali Raza, and Shahzad Hassan. 2019. Sentiment analysis of imdb movie reviews using long short-term memory. *Journal of Intelligent & Fuzzy Systems*, 36(1):1037–1043.

Abroon Muhammad Qazi and Anum Muhammad Qazi. 2019. Sentiment analysis of customer product reviews using machine learning approaches. *PloS one*, 14(9):e0221972.

Abhishek Singh, Chinmay Kulkarni, and Navpreet Amrinder Ayan. 2019. Sentiment analysis of imdb movie reviews. In *2019 11th International Conference on Communication Systems Networks (COMSNETS)*, pages 730–734. IEEE.

Pulkit Singhal and Pushpak Bhattacharyya. 2016. Sentiment analysis and deep learning: A survey.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

Yequan Wang, Minlie Huang, Li Zhao, and Xiaoyan Zhu. 2016. Attention-based lstm for aspect-level sentiment classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 606–615. Association for Computational Linguistics.