CHAPTER 1

INTRODUCTION

Sentiment analysis is the task of processing the given textual information to analyze the emotions in it. In simple words we need to analyze whether the textual information talks positive or negative feedback about the product or topic. It is also popularly known as opinion mining. It requires the knowledge of natural language processing, artificial intelligence and machine learning. Sentiment analysis is all about what other people are thinking about something. Sentiment analysis is very much useful as it provides useful inferences and also helpful to understand public opinion on a product or service.

Internet is a rich source of such textual opinion or review information. Analyzing such information would give us lots of information and future insights. For example in an online shopping website people usually write their reviews after buying and using the product. These reviews are very much helpful for the customers who wish to buy that product. The problem here is, when the number of reviews is large in number, it is not possible for the customer to read all the reviews before taking a decision. So it would be helpful if we can automate this process and the task is popularly known as sentiment analysis. Potential applications of this task are: Movie review analysis, product analysis, twitter opinion mining etc.

In this work we have focused on understanding the polarity of the given movie reviews by classifying whether it is positively polarized or negatively polarized. This problem can be posed as a multi label classification task where the final opinion could be worse, bad, neutral, good and excellent. In this work the problem is posed as a binary classification task where the final opinion can be either positive or negative. The reviews given by different people are of different lengths with different number of words in each review. Sentence vectorization methods are used to deal with the variability of the sentence length.

In this paper we try to investigate the affect of different hyper parameters like dropout, number of layers, activation functions. We have analyzed the performance of the model with different neural network configurations and reported their performance with respect to each configuration.

The IMDB benchmark dataset is used for our experimental studies that contain movie reviews that are classified as being positive or negative. In the experiment, an LSTM model is compared to other models and the LSTM model yields the best performance on the IMDB datasets.

CHAPTER 2

PROBLEM STATEMENT

1. **Sentiment Classification:**
   - Develop a machine learning model capable of classifying IMDB movie reviews as either positive or negative based on the text of the review.
2. **Data Preprocessing:**
   - Convert the textual movie reviews into a numerical format that can be used as input to the machine learning model. This involves tokenizing the text and padding the sequences to a fixed length.
3. **Model Building:**
   - Design and implement an LSTM network that can effectively learn from the sequence data of movie reviews and output accurate sentiment predictions.
4. **Model Training and Evaluation:**
   - Train the LSTM model on the IMDB dataset and evaluate its performance on a test set to ensure it generalizes well to new, unseen data.
5. **Predictive System Development:**
   - Create a function that can take a new movie review as input, process it appropriately, and use the trained model to predict the sentiment of the review.
6. **Handling Imbalanced Data:**
   - Ensure that the model can handle potential class imbalance in the dataset and still provide accurate sentiment predictions for both positive and negative reviews.
7. **Deployment Readiness:**
   - Develop a system that can be easily integrated into applications where real-time sentiment analysis of movie reviews is required, demonstrating its practical utility.

By addressing these problem statements, the project aims to create a robust and efficient sentiment analysis model using LSTM networks for movie reviews.

CHAPTER 3

Results and Discussion

Results

1. **Model Performance:**
    - The model's performance on the test dataset is evaluated using loss and accuracy metrics.
    - The final test loss and accuracy are printed as part of the evaluation.
2. **Test Accuracy:**
    - The model achieves a certain level of accuracy on the test dataset, indicating its ability to generalize to new, unseen reviews. This accuracy reflects how well the model can distinguish between positive and negative sentiments in movie reviews.
3. **Prediction Examples:**
    - Example reviews are provided to demonstrate the model's predictive capabilities. The sentiment of each example review is predicted and displayed, showing the practical application of the model.

Discussion

1. **Model Effectiveness:**
    - The LSTM model, designed to handle sequence data, effectively captures the contextual dependencies within the movie reviews. This allows it to understand the sentiment conveyed in longer and more complex sentences.
    - The embedding layer helps in converting the sparse representation of words into dense vectors, facilitating better learning for the LSTM layer.
2. **Training and Validation:**
    - The model is trained with a validation split to monitor performance on unseen data during training. This helps in preventing overfitting and ensures the model maintains good generalization capabilities.
    - The use of dropout in the LSTM layer further aids in preventing overfitting by randomly dropping units during training, which helps the model to learn more robust features.
3. **Evaluation Metrics:**

- Accuracy and loss are key metrics used to evaluate the model. While accuracy gives a straightforward measure of correct predictions, the loss function (binary cross-entropy) provides insight into how well the model's probability estimates match the true labels.

4. **Challenges and Limitations:**
    - One challenge in sentiment analysis is handling nuanced or ambiguous sentiments where the context is not straightforward. The model may struggle with such cases, resulting in incorrect predictions.
    - The fixed sequence length (200 words) might truncate longer reviews or pad shorter ones, potentially losing important information or adding unnecessary noise.
    - The performance can be further improved by experimenting with different model architectures, hyperparameters, or more advanced techniques like attention mechanisms.
5. **Future Work:**
    - Future improvements can include fine-tuning the embedding layer with pre-trained word vectors (e.g., GloVe or Word2Vec) to enhance the model's understanding of word semantics.
    - Exploring more complex architectures, such as bidirectional LSTMs or incorporating attention mechanisms, could further boost performance.
    - Expanding the model to handle multi-class sentiment analysis (e.g., very positive, positive, neutral, negative, very negative) could provide more granular insights into the sentiment expressed in reviews.

.

CHAPTER 4

CONCLUSION

The IMDB Reviews - Sentiment Analysis project successfully demonstrates the application of Long Short-Term Memory (LSTM) networks in natural language processing (NLP) for binary sentiment classification. The process involved several key steps, including data preprocessing, model building, training, evaluation, and the development of a predictive system.

Key Takeaways:

1. **Data Preprocessing:**
   - The conversion of textual data into numerical sequences using tokenization and padding was essential for preparing the data for input into the LSTM model.
2. **LSTM Model:**
   - The LSTM network, with its ability to capture long-term dependencies and context within the sequence data, proved effective in understanding and predicting the sentiment of movie reviews.
3. **Model Training and Evaluation:**
   - The model was trained on a substantial portion of the dataset and evaluated on a separate test set to ensure its generalizability. The achieved test accuracy indicates a satisfactory level of performance for sentiment classification tasks.
4. **Predictive System:**
   - The predictive function developed as part of the project allows for real-time sentiment analysis of new movie reviews, demonstrating the practical utility of the trained model.
5. **Challenges and Limitations:**
   - Some challenges include handling nuanced sentiments and the fixed sequence length, which might truncate or pad reviews inadequately. Despite these limitations, the model provides a solid baseline for sentiment analysis.
6. **Future Work:**
   - Enhancements such as using pre-trained word embeddings, experimenting with more complex architectures, and extending the model to multi-class sentiment analysis could further improve performance and applicability.

CHAPTER 5

REFERENCES

Here are some references for the concepts and libraries used in the project:

1. **TensorFlow and Keras Documentation:**
   - The official documentation for TensorFlow and Keras was instrumental in understanding the implementation of LSTM models and the use of various functions for data preprocessing and model training.
   - TensorFlow Documentation
   - [Keras Documentation]
2. **IMDB Dataset:**
   - The IMDB movie reviews dataset is widely used for sentiment analysis tasks. It provides a large corpus of labeled movie reviews.
   - IMDB Dataset
3. **Text Preprocessing Techniques:**
   - Guides and tutorials on text preprocessing techniques, such as tokenization and padding, were useful for preparing the text data for input into the LSTM model.
   - Jason Brownlee, "How to Prepare Text Data for Deep Learning with Keras," Machine Learning Mastery, 2017.
   - Text Preprocessing for Deep Learning with Keras
4. **LSTM Networks:**
   - Literature on LSTM networks provided insights into their architecture and advantages in handling sequence data for NLP tasks.
   - Sepp Hochreiter and Jürgen Schmidhuber, "Long Short-Term Memory," Neural Computation, 1997.
   - Original LSTM Paper
5. **Model Evaluation Metrics:**
   - Understanding the importance of evaluation metrics such as accuracy and loss in assessing model performance.
   - Jason Brownlee, "Metrics To Evaluate Machine Learning Algorithms in Python," Machine Learning Mastery, 2020.
   - Evaluation Metrics for Machine Learning
6. **Dropout Regularization:**

    - Techniques to prevent overfitting in neural networks, including dropout regularization, were referenced for improving model robustness.
    - Nitish Srivastava et al., "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," Journal of Machine Learning Research, 2014.
    - Dropout Regularization Paper
7. **Python Libraries:**
    - Extensive use of Python libraries such as pandas, NumPy, and scikit-learn for data manipulation, analysis, and splitting.
    - Wes McKinney, "Python for Data Analysis," O'Reilly Media, 2017.
    - Pandas Documentation
    - NumPy Documentation
    - scikit-learn Documentation

These references provided the foundational knowledge and practical guidance necessary for implementing the sentiment analysis model using LSTM networks.

1. **Web resources:**

    1. aman.ai

    2. https://course.elementsofai.com/

    3. deeplearning.ai

    4. https://developers.google.com/machine-learning/crash-course/

    5. https://ml-course.github.io/master/intro.html

    6. https://analyticsindiamag.com/all-the-free-ml-ai-courses-launched-at-google-i-o/ (Project based courses)