|  |  |
| --- | --- |
| Name Of The Student | Aryan Arun Kadam |
| Internship Project Title | RIO-125: Automate sentiment analysis of textual comments and feedback |
| Name of the Company | TCSiON |
| Name of the Industry Mentor | Debashis Roy |
| Name of the Institute | TKIET Warananagar |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Start Date | End Date | Total Effort (hrs.) | Project Environment | Tools used |
|  |  |  |  |  |
| **Project Synopsis:**  In today's digital age, online movie reviews hold immense power. They influence audience decisions, shape marketing strategies, and provide valuable feedback to filmmakers. However, manually analyzing vast amounts of movie reviews is a laborious task. This project proposes an automated sentiment analysis system dedicated to movie reviews.  This system will leverage Artificial Intelligence (AI) to analyze the emotional tone of textual reviews. By classifying reviews as positive, negative, or neutral, it will provide insights into:  **Solution Approach:**   * Rule Based Systems:  1. Rule-based approaches classify text into organized groups by using a set of linguistic rules. 2. Each rule comprises of a pattern based on semantics and its predicted category.  * Machine Learning based Systems:  1. Text classification based on past observations. 2. By using training data, the algorithm can learn the different associations between pieces of text and that a particular output (i.e. tags) is expected for a particular input (i.e. text). 3. Feature extraction: Transforms each text into a numerical representation in the form of a vector. E.g. bag of words [a vector represents the frequency in a predefined dictionary of words] 4. The algorithm is fed with training data consisting of feature sets. 5. Once trained with enough training samples, the machine learning model can begin to make accurate predictions on unseen text with similar feature sets. | | | | |
| **Data Preprocessing**  Movie Review Data Collection: Movie review data will be gathered from trustworthy sources.  **Text Cleaning:**  The text data will undergo cleaning to remove punctuation, stop words, and convert it to lowercase. Additionally, normalization techniques like stemming or lemmatization can be applied to enhance consistency.  **Feature Engineering**  **Sentiment-indicating Features:**  Relevant features will be extracted from the cleaned text to signify sentiment. Here are some examples:  **Word n-grams:** Sequences of words that capture sentiment patterns. For instance, a 3-gram like "worst movie ever" indicates negative sentiment.  **Sentiment Lexicons:** Pre-defined lists of positive and negative words commonly associated with movie reviews (e.g., "thrilling" vs. "disappointing").  **Focus on LSTMs:**  Since LSTM models inherently possess the capability to comprehend intricate sentence structures and capture subtle nuances of sentiment, explicit part-of-speech (POS) tagging may not be deemed essential. LSTM networks excel at processing sequential data and have demonstrated proficiency in deciphering context-rich information, particularly in longer and more complex reviews. Their ability to implicitly learn syntactic and semantic patterns within textual data mitigates the necessity for additional linguistic features like POS tagging. However, it remains prudent to maintain an exploratory stance regarding the inclusion of POS tagging within the analysis framework. Should a comprehensive evaluation reveal discernible improvements in model performance attributed to the incorporation of POS information, its integration can be judiciously pursued. This adaptive approach ensures that the analysis framework remains dynamic, capable of accommodating refinements based on empirical evidence and emerging insights. Thus, while not initially imperative, the inclusion of POS tagging stands as a potential avenue for enhancing the LSTM model's efficacy in capturing sentiment nuances within IMDb movie reviews.  **Model Training with LSTM**  **Labeled Dataset:** A machine learning model will be trained on a dataset of movie reviews with corresponding sentiment labels (positive, negative, or neutral).  **LSTM Model Selection:**  This report focuses on utilizing an LSTM network for sentiment analysis. LSTMs are particularly adept at handling sequential data like text, making them well-suited for this task.  **Training Process:**  The chosen LSTM model will be trained on the labeled dataset to establish the connection between extracted features and sentiment labels.  **Sentiment Classification:**  Once trained, the LSTM model can analyze new, unseen movie reviews. Based on the extracted features from the review text, the model will predict its sentiment (positive, negative, or neutral). | | | | |
| **Assumptions:**  The project operates under the assumption that IMDb movie reviews are in English and labeled with sentiment polarity (positive, negative, or neutral). However, the model's accuracy may be influenced by sarcasm, slang, or informal language. Techniques will be explored to address these challenges, such as:   * Adapting the model to handle sarcasm and informal language. * Incorporating sentiment analysis of slang terms. * Employing context-aware approaches to discern nuanced expressions. By acknowledging these factors, the project aims to enhance the model's robustness in accurately capturing sentiment across diverse linguistic nuances. | | | | |
| Project Diagrams:   * The image visualizes the training and validation performance of a deep learning model. * It displays changes in a specific metric (e.g., accuracy) over multiple epochs. * The x-axis represents the number of epochs, while the y-axis denotes the metric's value. * Both training and validation metric values are plotted on the same graph for comparison. * The image facilitates analysis of the model's training progress and generalization ability. | | | | |
| **Algorithms:**  Text Classification Algorithms  Some of the most popular machine learning algorithms for creating text classification models include the naive bayes family of algorithms, support vector machines, Regressions, and deep learning algorithms with CNN and RNN. Metrics and Evaluation Cross-validation is a common method to evaluate the performance of a text classifier.   1. It consists in splitting the training dataset randomly into equal-length sets. 2. For each set, a text classifier is trained with the remaining samples (e.g. 75% of the samples). 3. The classifiers make predictions on their respective sets and the results are compared against the human-annotated tags. 4. With these results, a performance metrics is built, that are useful for a quick assessment on how well a classifier works.   Performance metrics normally includes:   * Accuracy: the percentage of texts that were predicted with the correct tag. * Precision: the percentage of examples the classifier got right out of the total number of examples that it predicted for a given tag. * Recall: the percentage of examples the classifier predicted for a given tag out of the total number of examples it should have predicted for that given tag. * F1 Score: the harmonic mean of precision and recall. | | | | |
| **Outcome:**  Our project focused on leveraging Long Short-Term Memory (LSTM) neural networks to analyze sentiments expressed in IMDb movie reviews. Through meticulous preprocessing and architectural design, our LSTM model adeptly classified each review into positive, negative, or neutral categories. This enabled us to extract valuable insights into audience reactions towards movies, empowering filmmakers and enthusiasts with a nuanced understanding of viewer sentiments. The model demonstrated robust performance in sentiment classification, achieving notable accuracy and providing a comprehensive perspective on audience perceptions. | | | | |
| **Exceptions considered:**  Throughout the implementation of our sentiment analysis project on IMDb movie reviews using LSTM neural networks, we considered and addressed several notable exceptions, including:  **Noisy or Ambiguous Data:** Acknowledging the presence of noisy or ambiguous data within the IMDb review dataset, we implemented rigorous data preprocessing techniques:   * Tokenization * Stop word removal * Sequence padding   **Overfitting Mitigation:** Given the complexity of LSTM architectures and the limited dataset size, we recognized the potential for overfitting during model training. To address this, we employed:   * Regularization techniques such as dropout * Early stopping to prevent the model from memorizing the training data   **Hyperparameter Tuning:** We acknowledged the importance of hyperparameter tuning in optimizing the model's performance. Through systematic experimentation and validation on the validation set, we fine-tuned:   * Learning rate * Batch size * LSTM layer configurations   By conscientiously addressing these exceptions, we ensured the robustness and reliability of our sentiment analysis model for IMDb movie reviews. | | | | |
| Enhancement Scope:  Expanding the scope of the sentiment analysis project on IMDb movie reviews using LSTM neural networks presents several exciting opportunities for enhancement. Here are some potential avenues for further exploration:  **Multimodal Analysis:** Incorporate additional modalities such as images, audio, or metadata associated with movie reviews to perform multimodal sentiment analysis. This could provide a more comprehensive understanding of audience sentiment by considering multiple sources of information.  Fine-grained Sentiment Analysis: Instead of classifying reviews into broad categories (positive, negative, neutral), enhance the model to perform fine-grained sentiment analysis. This could involve identifying specific aspects of movies (e.g., acting, plot, cinematography) and analyzing sentiments associated with each aspect individually.  **Aspect-based Sentiment Analysis:** Develop a model capable of identifying and analyzing sentiments expressed towards specific aspects or features mentioned in movie reviews. For example, categorize sentiments towards character development, plot twists, or visual effects, providing filmmakers with granular insights into audience preferences.  **Temporal Analysis:** Explore how sentiments expressed in movie reviews evolve over time, considering factors such as release dates, trends in movie genres, or cultural events. This could involve analyzing sentiment trends over different time periods or identifying correlations between sentiment fluctuations and external factors.  **User Personalization:** Implement techniques for user-specific sentiment analysis, considering individual preferences and biases. This could involve building user profiles based on past review data and tailoring sentiment analysis results to each user's unique perspective.  Cross-domain Sentiment Analysis: Extend the analysis to incorporate reviews from multiple domains beyond IMDb, such as social media platforms, forums, or news articles. This could provide a broader understanding of public opinion towards movies across different online platforms.  Interactive Visualization: Develop interactive visualization tools to present sentiment analysis results in an intuitive and engaging manner. This could include sentiment heatmaps, sentiment timelines, or interactive dashboards that allow users to explore and interact with sentiment data dynamically. | | | | |
| Link to Code and executable file: | | | | |