Deep Learning Project Report

Group Members:

1. Reihaneh Moghisi
2. Shakiba Mashini
3. Jinendra Shrimal
4. Muhammad Harris
5. Hussain Kaide Johar Manasi

Table of Contents

Part 1

CinemaScope: Automated Sentiment Analysis of IMDb Movie Reviews for Strategic Entertainment Marketing

Executive Summary

With the help of deep learning, CinemaScope is an endeavor that will help the entertainment industry's marketing and content development by leveraging IMDb movie ratings. Our goal is to convert unprocessed movie review data into useful insights using a cutting-edge LSTM-based sentiment analysis algorithm. This will help us make more strategic decisions and gain a competitive advantage.

Background:

Comprehending audience mood is crucial for those working in the film production and distribution industries. IMDb, social media, and other platforms provide access to viewer input, which is a rich source of information that, with careful analysis, can highlight patterns and preferences. But the manual analysis of this input that is done by hand is not only sluggish and subjective, but it cannot keep up with the amount of data that is released so quickly every time a new movie comes out.

Problem Statement

In order to enable fast and precise categorization of viewer attitudes as either positive or negative, our aim is to automate the sentiment analysis of movie reviews. With the use of automation, we will be able to quickly gauge public sentiment, identify changes in viewer preferences, and modify our marketing tactics to increase effectiveness and engagement.

Objectives

- **Automate Sentiment Analysis:** Create a machine learning model that can categorize movie reviews' sentiment automatically, doing away with the necessity for laborious manual analysis.
- **Optimize Marketing Strategies:** Use sentiment analysis results to tailor marketing campaigns, emphasizing aspects that get good feedback.
- **Content Development Insight:** Make use of analytic findings to give content producers useful criticism in an effort to raise viewer satisfaction levels for next initiatives.
- **Competitive Analysis:** Monitor opinion patterns about our films as well as those of our rivals to facilitate tactical market placement.

Expected Impact

- **Enhanced Efficiency:** It is anticipated that automating sentiment analysis will greatly reduce the time and labor presently devoted to manual evaluation.

- **Enhanced Marketing Effectiveness:** We expect higher viewer engagement and better conversion rates when we apply marketing methods that are in line with audience emotion.
- **Data-Driven Content Creation**: By using sentiment analysis insights, content producers can produce works that are more successful and more relatable to their target consumers.
- **Enhanced Competitive Edge:** Staying up to date with sentimental trends will help us make strategic decisions that will keep us competitive in a changing market and maintain a strong standing within the entertainment sector.

Data Acquisition

TensorFlow Datasets provides the IMDb movie reviews dataset, which is the dataset used for this research. It includes fifty thousand IMDb movie reviews, categorized as either favorable or negative. It is ideal for training and testing our sentiment analysis model because of its binary classification. In order to ensure a balanced approach to both the learning and assessment stages, the dataset is pre-divided into 25,000 reviews for training and 25,000 for testing.

Model Architecture

Our LSTM model architecture captures both short- and long-term dependencies in movie reviews, efficiently managing the sequential structure of text data:

- **Embedding Layer:** Compared to one-hot encoded vectors, the richer representation provided by the first layer's transformation of each token (word) into a 128-dimensional vector is possible. In order to prevent overfitting and lower computing complexity, the model concentrates on the most frequently occurring terms, handling a vocabulary size of up to 10,000 words.
- **Layers of LSTM:** There are two LSTM layers used:
- The first layer of the LSTM is made up of 64 units and returns sequences, which enables it to send its outputs for additional temporal processing timestep by timestep.
- The second LSTM layer, which has 64 units as well, only transmits the final output and reduces the temporal information to a single context vector that summarizes the full review rather than returning sequences.
- **Output Layer:** A probability indicating whether the review is positive or negative is produced by a dense output layer consisting of a single neuron and a sigmoid activation function.

Training and Compilation

- **Compilation:** The Adam optimizer, renowned for its effectiveness in managing sparse gradients on noisy tasks like text classification, is used to create the model. For our binary classification task, binary cross entropy is the suitable loss function, and accuracy is monitored as the performance parameter.
- **Training:** A batch size of 32 people is trained across ten epochs. This strikes a compromise between model performance and training speed, enabling the network to update its weights frequently enough to achieve efficient convergence without updating them too quickly. To track performance and prevent overfitting, validation is carried out on a portion of the training set.

Model Evaluation

To make sure our model generalizes far beyond the data it was trained on, the last assessment of the model is performed on the unseen test set. Accuracy and loss score are the primary metrics used to assess the model's performance. Accurate sentiment classification would be confirmed by the model's high accuracy and minimal loss on the test set.

Conclusion

By adding these specifics to the report, we are able to provide readers with a clearer knowledge of how the LSTM model works inside the CinemaScope project and how it contributes to our strategic goals in the entertainment sector. CinemaScope hopes to use state-of-the-art technology to better content development, optimize marketing techniques, and keep a competitive edge in a field that is changing quickly by automating the sentiment analysis of movie reviews.

Final Project Part-II

Concept

In the popular arcade game "Space Invaders," players take control of a sleek spaceship positioned at the bottom of the game screen. The primary goal is to fend off an invading army of aliens that move horizontally and descend slowly towards the player's position. The objective is to eliminate all the aliens before they reach the bottom of the screen. Players can skillfully maneuver their spaceship to the left or right and launch projectiles upwards to destroy the advancing aliens.

Actions

- **Move Left**: Move the spaceship one unit to the left.
- **Move Right**: Move the spaceship one unit to the right.
- **Fire**: Shoot a projectile upwards to destroy aliens.
- **Do Nothing**: Skip the turn.

States

- **Player State**: The position of the spaceship.
- **Alien State**: Positions of all aliens in the grid and their movement direction (left or right).
- **Projectile State**: Position of any projectiles fired by the player

Rewards/Penalties

- **+10 points** for each alien destroyed, encouraging the player to eliminate threats.

- **-50 points** if an alien reaches the bottom of the screen, heavily penalizing failures in defense.
- **-1 point** for each shot fired, promoting efficient shooting and avoiding spamming.

Hypothesis for Action Rule and Reward/State Distribution

Network Architecture

- **Input Layer**: State representation, including the positions of the player, aliens, projectiles, and the current direction of alien movement.
- **Hidden Layers**: Multiple layers of neurons with ReLU activation functions to process the spatial relationships and dynamics of the game.
- **Output Layer**: Softmax layer that predicts a probability distribution over possible actions based on the estimated Q-values for each action.

Learning Strategy

- **Deep Q-Network (DQN)**: A neural network that approximates the optimal action-value function, predicting Q-values for each possible action given the current state.
- **Experience Replay**: Implements a replay buffer to store and reuse past experiences, aiding learning stability and efficiency.
- **Target Network**: A secondary network that provides stable targets during temporal difference learning, updated less frequently to stabilize the updates.

Detailed Analysis Procedure

Define the State and Action Spaces:

- **State Space**: Quantify and encode all possible configurations of player, aliens, and projectiles into a neural network-compatible format.
- **Action Space**: Enumerate all possible actions, including moving left, moving right, firing, and doing nothing.

Initialize Networks:

- Configure the primary DQN and a similarly architected target network with identical initial weights.
- Set up the optimizer, loss function, and learning rate.

Simulate Gameplay:

- Utilize an epsilon-greedy policy for action selection to balance exploration of new strategies and exploitation of known effective strategies.
- Update the game state based on the actions taken and the inherent game dynamics.

Train the Model:

- Record new states, rewards, and actions at each step, storing these transitions in the replay buffer.
- Periodically train the network on a batch of experiences sampled from the buffer.
- Use the target network to estimate target Q-values and update the primary network based on these targets.

Iterative Learning and Evaluation:

- Continuously assess performance based on average rewards and the number of aliens successfully destroyed.
- Tune hyperparameters and modify the neural network structure as necessary based on performance metrics and observed learning behavior.

Deployment and Continuous Improvement

- Deploy the trained model in a live game environment.
- Allow the model to keep learning from new gameplay experiences, adapting to potentially evolving game dynamics, and increasing difficulty levels.

Final Reward

The final reward in this game is a combination of positive rewards for destroying aliens and negative penalties for failing to defend against aliens reaching the bottom of the screen or inefficiently using projectiles. It is cumulative, as it reflects the player's performance over the course of the game. It incentivizes the player to strategically manage resources (projectiles) and prioritize actions (defending against advancing aliens) to maximize overall reward while minimizing penalties.

Conclusion

Reinforcement learning is a powerful technique for training AI models to perform strategic moves in various applications. In the timeless arcade game "Space Invaders," the use of reinforcement learning requires careful consideration of state and action representations, learning approaches, and continuous refinement. This framework serves as a practical tool for both educational and gaming purposes in the AI field, providing a structured approach for effectively training models to execute strategic moves in "Space Invaders.”