**Slide 1: Introduction**

Hello everyone. Today, I’m excited to present our project on sentiment analysis using Amazon product reviews. The objective of our project was to develop a model that can accurately classify reviews as positive or negative. This is particularly important in understanding customer sentiment, which can greatly influence business decisions.”

**Slide 2: Dataset Overview**

“We sourced our dataset from a collection provided by Johns Hopkins University, which consists of positive and negative reviews from Amazon. Our dataset contains approximately 200000 reviews that span multiple product categories, including electronics, books, and home appliances. This diversity allows us to train a robust model capable of generalizing well across different types of reviews.”

**Slide 3: Data Preprocessing**

To prepare the data for modeling, we underwent several preprocessing steps. First, we cleaned the text by removing punctuation and normalizing the case to ensure consistency. We also filtered out very short reviews to eliminate any noise. Tokenization followed, allowing us to split the text into individual words for numerical encoding. Finally, we employed padding and truncation to standardize the length of input sequences. These steps are crucial as they improve the quality of the data, reduce noise, and enhance the model’s performance.

**Slide 4: Text Encoding**

For the text encoding process, we utilized Keras’ tokenizer to convert our cleaned text into sequences of integers, where each unique word is represented by a distinct integer. Padding was applied to ensure uniform input length across the reviews, accommodating our model’s requirements. We also converted the sentiment labels into binary values for model input: positive reviews were labeled as 1 and negative reviews as 0. This numeric representation is critical, as it allows the neural network to process the data effectively while preserving relationships between words.”

**Slide 5: Model Architecture**

“Now, let’s discuss the architecture of our neural network. Our model features an embedding layer that transforms input integers into dense vectors, capturing the underlying meaning of words. This is followed by two LSTM layers, which are essential for processing sequences of word vectors and capturing temporal dependencies. Finally, we used a dense layer configured for binary classification, which allows the model to output sentiment probabilities. LSTMs are particularly suited for this task as they effectively handle sequential data and mitigate the vanishing gradient problem, enabling long-range context retention.”

**Slide 6: Model Training and Evaluation**

Moving on to model training and evaluation, we split our dataset into three sets: 70% for training, 15% for validation, and 15% for testing. Our model was trained using the Adam optimizer, along with the binary cross-entropy loss function. During evaluation, we achieved a test accuracy of 85%, with a final loss of 0.35. These metrics indicate that the model learned to classify sentiments effectively. By examining the confusion matrix, we gained further insights into the model’s performance, highlighting its strengths in identifying both positive and negative sentiments.”

I will pass the rest presentation to Yousuf: