Problem Statement: SENTIMENT ANALYSIS OF IMBD Movie REVIEWS USING LTSM (Long Short-Term Memory) Neural Networks

Develop a sentiment analysis system using Long Short-Term Memory (LSTM) neural networks to analyze customer reviews scraped from websites like Amazon and IMDB. This system will automatically classify the sentiment of reviews between a scale of **-1** to **+1** where a score near **–1** is very negative, **+1** is very positive & score near **0** would be neutral.

Team: List the members of your team (names and student IDs)

1. **MEMBER 1**
   Name: Kabiir Krishna
   Email: k7krishn@uwaterloo.ca
   WatIAM: k7krishn
   Student Number: 21106092

2. **MEMBER 2**
   Name: Akshat Baheti
   Email: abaheti@uwaterloo.ca
   WatIAM: abaheti
   Student Number: 21100660

Data: What data will you use? If you are collecting new datasets, how do you plan to collect it?

We will perform web scraping (using BeautifulSoup) to scrape Movie/TV Show reviews from IMDB website. The scraped data will consist of the review text and any relevant metadata (e.g., rating stars, date). We aim to collect at least 10,000 labeled reviews with continuous scores between –1 & 1 (1 = highly positive, -1 = highly negative, 0 = neutral) to ensure sufficient training data for the LSTM model.

Methodology/Algorithm: What method or algorithm are you proposing? If there are existing implementations, will you use them and how? How do you plan to improve or modify such implementations?

1. **Data Extraction & Preprocessing:**
   a. **Extraction:** We will scrape the data from IMDB using BeautifulSoup.

    b. **Pre-Processing:** Clean the scraped reviews by removing special characters, converting text to lowercase, and potentially stemming / lemmatization.
    c. Convert text reviews into numerical sequences using techniques like word embedding.
2. **Synthesizing training Dataset:**
    a. We would score the scraped reviews using VADER, create a training dataset
    b. **Bonus:** To augment VADER's scores, we plan to pair it with a Neural Network to further augment its scoring & create a strong model.
3. **LSTM Model Development:**
    a. Build an LSTM Neural Network from scratch for testing architecture with layers for embedding, LSTM units (potentially bidirectional), and final classification layers.
    b. Train the LSTM model on the preprocessed training data developed from VADER, aiming to improve its ability to predict the correct sentiment labels for latest reviews.
4. **Evaluation:**
    a. Evaluate the model's performance on a hold-out test set using metrics like accuracy, precision, recall, and F1 score for each sentiment class.
    b. Visualize the model's performance using techniques like confusion matrices or sentiment distribution plots.

Related Work (3+ prior works): Which papers will you read to inform your understanding of the problem, and the appropriate method to tackle it?

We referred to the following papers to understand the problem statement & selecting our topic & approach:

1. **Title: Comparison of Accuracy between Long Short-Term Memory-Deep Learning and Multinomial Logistic Regression-Machine Learning in Sentiment Analysis on Twitter**
   Authors: Aries Muslim, Achmad Benny Mutiara, Rina Refianti, Cut Maisyarah, Karyati, Galang Setiawan
   Summary: Comparison of LSTMs with other Machine Learning methods
   Learning: Helped us decide which model we can use for our project & verify how LSTMs fare against other approaches

2. **Title: Sentiment analysis using neural network**
   Authors: Akshi Kumar, Ritu Rani

Summary: Uses PNN (Probabilistic Neural Networks) for sentiment analysis on twitter data
Learning: Different approaches used for sentiment analysis of customer/user provided data.

3. **Title: Learning Word Vectors for Sentiment Analysis**
   Authors: Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, Christopher Potts
   Summary: The paper presents a model that uses a mixture of unsupervised and supervised techniques to find word vectors capturing linguistic nuances.
   Learning: The benefits of a vector-based approach for NLP & How to correlate string data with Neural Network models for effective usage.

4. **Title: Explaining Recurrent Neural Network Predictions in Sentiment Analysis**
   Authors: Leila Arras, Grégoire Montavon, Klaus-Robert Müller, Wojciech Samek
   Summary: The paper aims at applying LRP to a word-based bi-directional LSTM model on a five-class sentiment prediction task for emotion detection.
   Learning: Working with LSTMs coupled with other models for Sentiment and emotion analysis

5. **Title: Deep CNN-LSTM with combined kernels from multiple branches for IMDb review sentiment analysis**
   Authors: Alec Yenter, Abhishek Verma
   Summary: This paper describes a unique sentiment analysis approach using combined kernel from multiple branches of convolutional neural network (CNN) with long remembering (LSTM) layers including multiple alternative approaches.
   Learning: Possibly using a 3-black-box Model to do Sentiment analysis.

6. **Title: Attention Is All You Need**
   Authors: Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin
   Summary: The paper that the Transformer generalizes well to alternative tasks by applying it with success to English body parsing each with massive and restricted coaching information.
   Learning: Alternatives to LSTMs and it’s variants like new ‘Transformer’ architecture as introduced by the paper

Evaluation Plan: How will you evaluate your results? Qualitatively, what kind of results do you expect (e.g., plots or figures)? Quantitatively, what kind of analysis will you use to evaluate and/or compare your results (e.g., performance metrics or statistical tests)?

We will evaluate the model's performance on a separate test set not used for training. We will measure accuracy, precision, recall, and F1 score for each sentiment class (positive, negative, neutral). Additionally, we will visualize the model's performance using confusion matrices and sentiment distribution plots to gain insights into its strengths and weaknesses. Our expected output is a model that can score a review between –1 and +1 with high accuracy & confidence along with analysis (graphs etc.) for a given movie.

Timeline:

- Week 1: Data scraping, cleaning, and training data synthesis.
- Week 2-3: LSTM model development and training.
- Week 3-4: Model evaluation and visualization.
- Week 4: Report writing and finalization.

Tools and Technologies:

- Web scraping: BeautifulSoup
- Data analysis/cleaning: Pandas, NumPy
- Machine learning/Deep Learning: TensorFlow/Keras or PyTorch
- Visualization: Seaborn / Matplotlib
- Language Processing: Spacy, NLTK, Gensim

Note:

This proposal provides a high-level overview. We will refine/modify the details as we progress through the project.

Minimal Requirements: Does your project meet the following requirements?

Requirement	Mapping
Dataset with at least 10,000 labeled examples if you propose a supervised method. More will be needed for some tasks like machine translation,	• The proposal outlines scraping customer reviews from IMDB website (Section: Data).

conversational systems and 100-way classification.	• We are plan to scrape **11,000+ reviews** for multiple movies with a functional spider (Section: Data). • We will scrape 10,000 + reviews from the same source (IMDB or Amazon) to create our training corpus, which is to be fed to VADER
You will be able to collect the complete dataset by the project milestone due date.	We have created a working web-crawler to scrape data. We have allotted **Week 1** for data scraping, cleaning, and labeling aiming for data collection & training data synthesis within that period (Section: Timeline).
Task is feasible: either prior work on the dataset exists or a human can get good accuracy on it.	• For scraping, we are take a simple website (review page of a movie on IMDB) & traverse it's HTML tree, for the same type of data on the same website (i.e. IMDB review page for other movies), the reviews will be at the same level/structure/node in the HTML tree regardless, this makes our web scraper scalable, allowing us to quickly collect data points • The proposal references existing research using LSTMs for sentiment analysis on similar data (Section: Related Work). • This shows the task's feasibility with LSTMs.
You have identified an automatic (i.e., can be computed by a computer) evaluation metric for the task.	• The proposal mentions using established metrics like accuracy, precision, recall, and F1 score (Section: Evaluation Plan). • These are all automatic metrics computable by the machine learning model. • Moreover, we would visualize the inferences from out testing dataset to give a better idea about what people think about the movie/product

If no automatic evaluation is possible (rare tasks must be justified), you must have a clear understanding of how to perform a reliable human evaluation of your system, and it must be possible within the time limit of this course.	This is not applicable in this case. The proposal focuses on automatic evaluation metrics & visualization of inferences obtained. Since sentiment about a movie is a subject criterion
Using NLP you must get good performance on the task. For example, predicting stock prices from twitter data would not fit this requirement because the strongest	We are performing several NLP techniques crucial for the task (Section: Methodology/Algorithm): • Text cleaning (removing special characters, converting to lowercase) • Word embedding (converting text to numerical sequences) • Sentiment Scoring(assigning sentiment scores)

----------THE END ----------