Aspect-Based Financial Sentiment Analysis

Rounak Jain
Department of Information Technology
National Institute of Technology Karnataka (NITK)
Surathkal, India
rounakjain.211it055@nitk.edu.in

Pari Poptani
Department of Information Technology
National Institute of Technology Karnataka (NITK)
Surathkal, India
paripoptani.211it045@nitk.edu.in

Sudarshan Zunja
Department of Information Technology
National Institute of Technology Karnataka (NITK)
Surathkal, India
sudarshanzunja.211it072@nitk.edu.in

Verma Ayush
Department of Information Technology
National Institute of Technology Karnataka (NITK)
Surathkal, India
vermaayush.211it079@nitk.edu.in

***Abstract*—Understanding sentiment in financial text is important for gaining better insights into the financial market. This research aims to enhance aspect-based sentiment analysis (ABSA) by addressing challenges specific to financial texts, such as complex language and the importance of numbers. Existing models treat numbers as regular words, limiting their effectiveness in financial sentiment analysis. The study proposes a deep learning model using a specialized version of BERT to process both words and numbers, showing significant improvements in sentiment prediction accuracy. The results suggest that incorporating number processing can improve the accuracy of financial sentiment analysis, aiding analysts and firms in decision-making.**

***Index Terms*—financial sentiment analysis,numeral understanding, domain knowledge,deep learning,natural language processing.**

I. INTRODUCTION

Problem Statement: This project focuses on sentiment analysis in aspect based financial texts.It aims to solve challenge of accurately understanding sentiments in professional financial language,which include complex structures and numerical data.

Background: While sentiment analysis has advanced, existing models struggle with financial texts as they treat numbers as regular words and fail to capture specialized financial terminology, leading to less accurate sentiment analysis.

Input and Output: The input consists of financial text data (articles, reports), and the output is a sentiment score that indicates the positive or negative sentiment of the text, aiming for precise sentiment prediction.

Application and Motivation: The research enhances financial analysis tools, enabling better investment decisions, risk management, and market analysis, driven by the need to improve models that don't handle numbers or specific financial language effectively.

Uniqueness of the Project: What makes this project unique is its new approach to aspect based sentiment analysis.It incorporates special techniques that focus on financial language and numbers.we used methods that align sentiment related words with important aspect terms,making the analysis more detailed.

Contributions of the Project:

1) Aspect Embedding:This project creates aspect embeddings by combining key terms with relevant words.This multi dimensional representation helps align important aspect terms with sentiment words,improving analysis.
2) Aspect Centric Mechanism:The model uses a mechanism that directs attention to key parts of sentences based on aspect specific queries,adjusting to the importance of numbers or important context.
3) Multi Head Attention Mechanism:The model has multiple attention heads that focus on different parts of a sentence,such as numbers,context or structure.Combining these outputs gives a deeper understanding of text,leading to better sentiment accuracy.

Overview of Paper Structure: The paper is laid out as follows:The paper is structured as follows: Section 2 reviews related works, Section 3 outcomes of the literarure survey, Section 4 details the methodology,Section 5 discusses the results, and Section 6 concludes with future directions.

II. LITERATURE REVIEW

The paper [1] introduces a technique that enhances aspect-based sentiment analysis by integrating numeral and affective knowledge, addressing gaps in traditional methods for handling financial sentiment data.

The paper [2] highlights the evolution of sentiment analysis methods for financial news, with a focus on extracting aspect-based sentiments regarding topics like market trends and company performance.

The paper [3] helps to review traces the development of aspect-based sentiment analysis (ABSA), emphasizing the shift from rule-based to advanced machine learning and deep learning techniques, particularly neural networks like BERT.

The paper [4] examines deep learning techniques, such as LSTM and GRUs, for sentiment analysis in financial markets, noting their ability to capture complex patterns from textual data.

The paper [5] highlighted the benefits of neural networks, like RNNs and transformers, in improving sentiment analysis accuracy and robustness for financial data, outperforming traditional methods.

The paper [6] discusses using deep learning, word embeddings, and neural networks for aspect-based sentiment analysis, emphasizing capturing complex financial relationships and context.

The paper [7] introduces a fine-grained approach for financial sentiment analysis, combining specialized financial lexicons with machine learning to detect nuanced sentiments in financial texts.

The paper [8] helps to study applies hierarchical transformer models for ABSA, improving sentiment analysis by capturing long-term dependencies in financial data.

The paper [9] presents a deep learning framework for aspect-based sentiment analysis, using LSTM networks to improve accuracy in extracting sentiments from financial news and social media.

The paper [10] gives review compares rule-based, machine learning, and deep learning methods for ABSA, focusing on attention mechanisms and transformers for enhanced contextual sentiment extraction.

The paper [11]explores sentiment analysis on social media to predict stock price movements, demonstrating the effectiveness of social media sentiment in market forecasting.

The paper [12] gives review discusses using text data from news and social media for stock market analysis, emphasizing sentiment analysis and machine learning to predict stock trends.

The paper [13] integrates sentiment analysis with feature extraction from news and social media to improve stock prediction accuracy using machine learning models.

The paper [14] enhances ABSA by integrating multiple knowledge sources, including sentiment lexicons and financial knowledge graphs, to improve sentiment extraction in financial texts [14].

III. Outcome of the Literature Review

A. Incorporate Multimodal Data

Integration from other sources,whether it is financial news,social media,and reports can give more extensive View of market sentiments.This will enable models to capturing subtle insights and improving predictive power.

B. Enhance Data Diversity

Datasets that come from different industries,regions,and time periods all point towards more general models of financial unbiased context.The process,therefore enhances the capability of handling varied financial settings,it enhances overall reliability.

C. Optimize Computational Efficiency

Using techniques like model pruning and quantization it reduces the training and inference cost while achieving performance.It brings about much faster sentiment analysis models,more practical to use for real time applications.

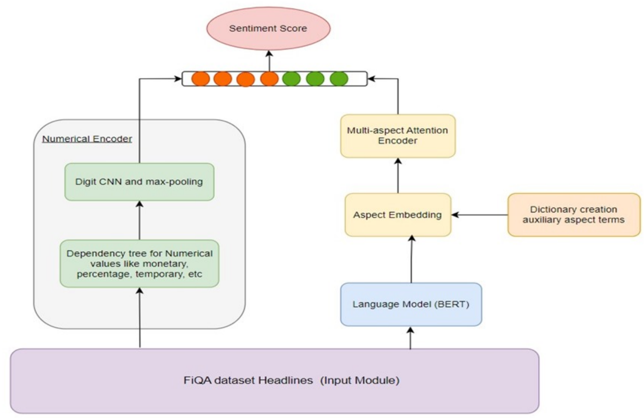

Fig. 1. Methodology Flowchart

IV. Methodology

A. Dataset

We used the FiQA Task 1 dataset, which includes financial news headlines, articles, and comments, to analyze sentiment in financial contexts. Each entry is labeled with a sentiment (positive, negative, or neutral) related to financial performance. This dataset is crucial for capturing the nuances of financial language, especially when numbers and descriptive phrases are combined.

Key Features of the Dataset:

- FiQA task1
- Size : 498 * 5
- Features :sentences, info snippets,info target,sentiment score,info aspects
- Diverse Content:The dataset included a range of financial content,such as news article, company reports and statements,covering sentiments from positive to negative.

B. Data Preprocessing

Preprocessing the dataset was vital to get it ready for analysis,as financial texts can be complex and include numbers.The following steps were part of data preprocessing:

Text Cleaning

- Removed punctuation and special characters for simplicity.
- Converted all text to lowercase for easy tokenization.
- Removed extra spaces and line breaks.

1) **Tokenization**
Split sentences into individual tokens.This allowed the model to map words into numerical formats.A BERT compatible tokenizer was used to align with the later embedding process.
2) **Stop Word Removal**
Removed common words like (the,is,and)to reduce noise.
3) **Numerical Representation**
Extracted and processed numerical data,recognizing

that numbers play a critical role in financial sentiment.Example, in the sentence ”Company shares rose 15%” the number 15 was isolated for special processing by the DigitCNN module.
4) **Aspect specific tagging:**
We identified and tagged words related to financial aspects to help the model focus on the words that are important as sentiment terms.Words like high,increase,decrease were flagged for aspect embedding creation.
5) **BERT Embedding Extraction** Used BERT model to create word embeddings for each sentence,capturing word meanings and their context.These embeddings were essential for understanding the relationship within the sentences and was used as input for further processing.

Working of embedding extraction:

1) Tokenization:The input text is first broken down into subwords and tokens using the BERT tokenizer.Special markers like CLS at start of a sentence and SEP at end of a sentence are added for clear boundaries.
2) Creating Embeddings:The pre trained BERT model processes this tokenized input and produces context aware embeddings for each token.The CLS tokens embedding often acts as a summary of the entire sentence.
3) Managing Dimensions:The output from BERT is a tensor with multiple dimensions,where each token has a fixed size vector like 768 dimensions for the base model.This output can then be used as input for further steps.

C. DigitCNN Class Implementation

The DigitCNN class is a special type of neural network that processes numerical data found in text, like financial figures or percentages,which are important for understanding sentiment in financial documents.Its main job is to help the model make sense of these numbers and how it influence the sentiment.

Working of DigitCNN

1) Input Layer:The extracted numerical data from the text is sent into the DigitCNN.This input is organized so that the network can read and process it correctly.
2) Convolutional Layers:These layers apply filters over numerical data to detect patterns like repeated numbers or trends,which might show changes in sentiment.
3) Activation Functions:Network uses activation functions like ReLU,to add nonlinearity,allowing it to learn more complex features.
4) Pooling Layers:After the convolutional layers,pooling layers are used to make the data simpler and reduce its size.This step helps to summarize the important information and makes processing faster.
5) Flattening and Dense Layers:Then simplified data is then flattened and passed through fully connected layers to create the final representation of numbers.
6) Output Integration:The output from DigitCNN is combined with other features,like BERT embeddings and aspect embeddings,to create a complete input for sentiment prediction.

Extracting and Using Numerical Information Extracting numerical information is crucial for financial sentiment analysis because numbers greatly influence meaning of the text.For example sentences like “profits increased by 15%” or “sales dropped by 10%” can change the sentiment significantly.The process includes:

1) Searching the text for numbers using pattern recognition methods,such as looking for sequences of digits like percentages,dollar amounts,growth percentages.
2) Once the numbers are found they are connected to words that give them meaning like increase or drop,to understand how they impact the sentiment.
3) The model examines the surrounding context of the number to decide its sentiment effect positive or negative.

D. Dependency Tree

We used dependency tree parsing to better understand sentence structure and improve sentiment analysis.This method captures how words are related providing more accurate sentiment insights.

- Preprocessing: Text Cleaning and Tokenization:We cleaned and split sentences into individual words for easier analysis.
- Dependency Parsing: Tree Building:Using a parser (like spaCy),we created a tree structure for each sentence that showed word relationships,like subject or modifier links.
- Feature Extraction:
Finding Key Terms:We used the tree to find important aspect terms linked to sentiment words.
Embedding Context:Related words were embedded together to show how they influenced sentiment.
- Handling Numbers:
Numerical ContextThe parser helped link numbers to sentiment words like ”increase” or ”decrease,” for better interpretation.
DigitCNN Integration:We processed numbers with DigitCNN to add numerical understanding.
- Embedding and Sentiment Analysis:
Combining Features:We combined tree based embeddings with other embeddings (like BERT) for better sentiment prediction.
- Example: For ”stake high AstraZeneca heart drug face tough competition,” the parser linked ”high” to ”AstraZeneca,” helping the model read this as positive.

E. Encoding Numerical Values

Encoding Numerical Values Encoding numerical values ensures that numerical data is represented in a format suitable for training and prediction.This process is important because raw numbers need to be transformed or embedded in a way that preserves their significance compared to other text in the input.

Steps in Encoding Numerical Values:

1) Normalization:Numerical data is often scaled to a common range such as 0 to 1 to prevent large values from skewing the model learning.
2) Embedding Numerical Context:Numbers are encoded together with surrounding context words to form a combined representation.This allows the model to understand not just the numerical value but how it influences the sentiment within the sentence.
3) Feature Vectors:The encoded numbers are converted into feature vectors which include attributes such as the size of the change,the direction increase or decrease and the type of value percentage, currency.These vectors are then used as inputs to the model.

Tensor Representation Tensors are key data structures in deep learning that store data in multi dimensional arrays,making them essential for combining numerical data with other types of embeddings in the model.

Sentence Example: The company reported a 25% increase in quarterly profits.

- Numerical Value Extracted:25%
- Context:increase in quarterly profits
- Sentiment Encoding:The number 25% is normalized and embedded with its positive context increase,creating a tensor that represents this information.

Tensors play a vital role in carrying data throughout the deep learning model.They act as the framework that supports data flow from the input to the output,passing through the models various layers to create a final output that reflects all aspects of the input data.

Flow Summary

1) Input tensor (textual and numerical data) is fed into the model.
2) Embedding and encoding layers process the input tensor to enhance representations.
3) The tensor flows through attention mechanisms and aspect embedding layers to focus on important features.
4) Convolutional and dense layers transform and integrate the data further.
5) Activation functions modify the tensor to capture non linear relationships.
6) The final layer outputs a prediction tensor.
7) The output tensor is interpreted to provide the final prediction result.

F. Aspect Embedding

Aspect embedding is an essential method that enhances the models ability to understand sentiment by linking aspects with the context in which they appear.This is particularly important in financial sentiment analysis.Aspect embedding functions by connecting important aspect terms in the text with contextual words that influence sentiment.Terms like ”growth” or ”decline” can change how sentiment is perceived for a particular financial entity or aspect.

Implementation Details

- Multi Dimensional Representation:Aspect embedding vectors were created by blending the embeddings of aspect terms with additional context words or numerical indicators.The output is a representation that captures both the meaning of the aspect and its context within the sentence.
- Dynamic Attention Mechanism:To determine which context words most influence an aspect,a dynamic attention mechanism may be applied,this helps the model focus on words that have a greater impact on the aspects sentiment,such as numerical values or specific adjectives like strong or weak.
- Integration with BERT Embeddings:Aspect embeddings are integrated with BERT based sentence embeddings to provide a comprehensive input for further layers in the model.This combination ensures that the model leverages both the deep contextual understanding provided by BERT and the finegrained,aspect specific insights from the aspect embeddings.

Sample Output Explanation To show how aspect embedding works: Sentence 1- Stake high AstraZeneca heart drug faces tough competition.

- Aspect: ”AstraZeneca”
- Influencing Term: ”high”
- Embedded Aspect: ”high AstraZeneca”
- Sentiment Contribution: The term ”high” suggests a positive sentiment associated with AstraZeneca in this context.

G. Deep Neural network use

We used a deep neural network to improve sentiment analysis by capturing complex relationships in financial text.

- Data Preparation:We cleaned and tokenized the text,turning it into numerical data to feed into the model.Important features like aspect terms and numbers were also added.
- Model Design:The DNN had multiple hidden layers with ReLU activation to learn complex patterns.We added dropout layers to prevent overfitting.
- Training:The model was trained with the Adam optimizer and mini batches to balance speed and accuracy.We monitored training with early stopping to avoid overfitting and used MSE and MAE as loss functions.
- Tuning and Evaluation:We fine tuned hyperparameters for best results.The models performance was checked using MSE and MAE and compared with other models.

H. Mean Squared Error MSE and Mean Absolute Error MAE

MSE and MAE are essential metrics used to gauge the accuracy of regression models,including those for sentiment analysis.In this context,these metrics were employed to evaluate how well the sentiment prediction model performed on a test dataset.

1) MSE measures the average squared difference between predicted and actual values, giving more weight to larger errors. A lower MSE indicates better model performance, as it shows the predictions are closer to the actual sentiment scores. The sensitivity to outliers makes MSE useful when it's important to penalize larger errors more heavily, ensuring that significant mispredictions are addressed.
2) MAE calculates the average of absolute differences between predicted and actual values, treating all errors equally without emphasizing large deviations. A lower MAE indicates more accurate sentiment predictions, providing a clearer view of the model's overall accuracy. MAE is preferred when a more intuitive, straightforward measure of error is needed, as it provides a direct average of how far the predictions are from the true values.

   Using MSE and MAE together provides a comprehensive understanding of the models performance.While MSE offers insight into errors with an emphasis on larger deviations, MAE delivers an average view of prediction accuracy.

I. Training and Validation Loss Trends

Training loss indicates how well the model fits the training data by measuring prediction errors during each epoch. A steadily decreasing training loss shows that the model is learning effectively, adjusting its weights and improving over time.

Validation loss, measured on a separate validation set, tracks the model's ability to generalize to unseen data. Initially, both training and validation losses are high, but as training progresses, the training loss decreases, and if the model generalizes well, the validation loss follows a similar trend, ensuring the model isn't overfitting.

V. Results and Analysis

The results of this project highlight the advancements made in the field of sentiment analysis by incorporating numerical context and targeted aspect embedding.This approach has demonstrated improvements over existing models,particularly in handling domain specific financial texts that require nuanced understanding beyond traditional NLP methods. Below are some of the results we achieved:

Evaluation Metrics (MSE and MAE) The evaluation of the model's performance is primarily based on two key metrics: Mean Squared Error (MSE) and Mean Absolute Error (MAE). These metrics are crucial for understanding the accuracy and reliability of the model's predictions.

1) **Mean Squared Error (MSE)** MSE shows how much the actual sentiment scores differ from the predicted ones,on average.It squares the differences,so larger errors are given more weight.In our results, the MSE was quite low, which means the model makes few large mistakes. This shows the model is good at capturing sentiment patterns in financial text data accurately.

```
3/3 ━━━━━━━━━━━━━━━━━━━━ 0s 45ms/step
Predicted vs. Actual Sentiment Scores (Test Set):
    Actual  Predicted
0    0.165   0.120466
1    0.516   0.436710
2    0.769   0.305050
3   -0.303  -0.600214
4   -0.479  -0.563444
..     ...        ...
70  -0.779  -0.175764
71   0.296   0.299240
72   0.364   0.305248
73  -0.181  -0.242410
74   0.432   0.260705

[75 rows x 2 columns]
Mean Squared Error (MSE) on Test Set: 0.0879522310260836
Mean Absolute Error (MAE) on Test Set: 0.24549865304152169
```

Fig. 2. MSE and MAE Score

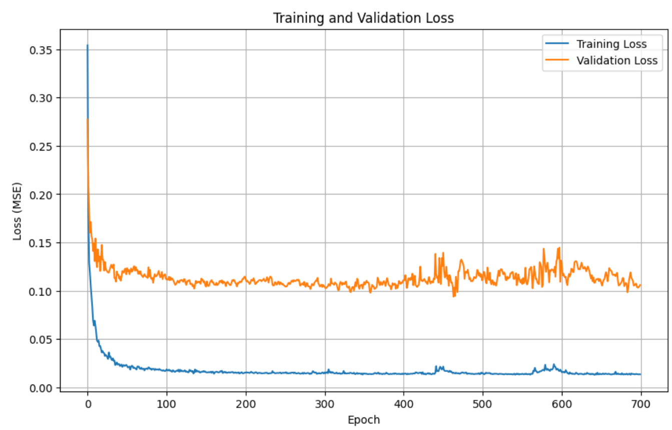

Fig. 3. Traning and Validation loss

2) **Mean Absolute Error (MAE)** MAE measures the average absolute difference between actual and predicted sentiment scores. Unlike MSE,it doesnot square the differences so its less affected by large errors.The MAE gives a clear idea of the average prediction error,expressed in the same units as the output.Our results showed a moderate MAE,which means the model consistently predicts with reasonable accuracy without extreme errors.
3) **Training vs Validation Loss Trends** Looking at the training and validation loss trends helps us understand how the model is learning and whether it's overfitting or underfitting.

   Decreasing Training Loss:During training,the models training loss gradually decreased,showing it was learning effectively from the training data.

   Validation Loss Behavior:At first,the validation loss also went down along with the training loss,showing the model was getting better at handling new data.After a while, the validation loss leveled off and started to

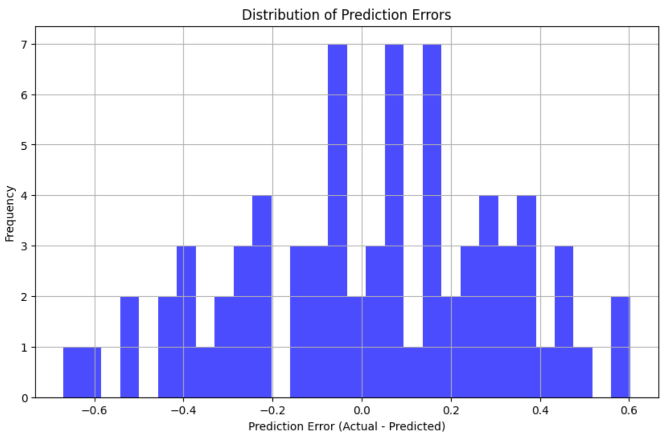

Fig. 4. Distribution of Prediction errors

fluctuate a bit.

4) **Aspect Embedding Analysis** Aspect embedding helps capture the connection between specific keywords and the sentiment words in a sentence.By adding domain specific knowledge,the model learned to tell positive, negative or neutral sentiments apart based on the context of the terms used.
Example 1:"CBI books Adani Enterprises,3 ex-NCCF officials for alleged irregularities in coal supply contract."
Aspect: Adani Enterprises
Overall Sentiment: Negative
Explanation: For Adani Enterprises, words like "books" and "alleged irregularities" imply legal action and potential misconduct, casting a negative light on the company.

Example 2: "Stakes High for AstraZeneca Heart Drug Facing Tough Competition."
Aspect: AstraZeneca
Overall Sentiment: Negative
Explanation: For AstraZeneca, the phrase "facing tough competition" suggests challenges, creating a mildly negative tone while remaining mostly factual.

Example 3:"AstraZeneca shares climb 3% as drug maker ups profits forecasts."
Aspect: AstraZeneca
Overall Sentiment: Positive
Explanation: For AstraZeneca, phrases like "shares climb" and "ups profits forecasts" signal growth and optimism, giving the sentence an overall positive sentiment.
Adding aspect embeddings improved the models metrics like MSE and MAE.It provided deeper insight into sentence structure and sentiment cues,making the analysis more than just basic text processing.

5) **Comparative Analysis with Existing Methods** To

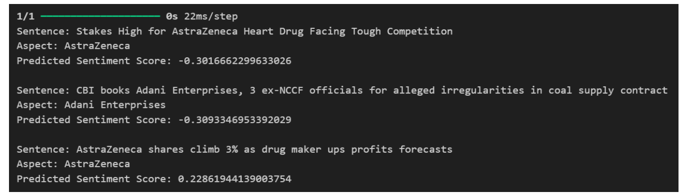

Fig. 5. Predicted Sentiment Score

demonstrate our models effectiveness,we compared it with top methods in financial sentiment analysis.
Numerical Understanding:Traditional models struggle with numbers but our model using a numerical encoder and DigitCNN interprets them in context, understanding statements like"profits increased by 25%."
Aspect Embeddings:Our model links sentiment words with key terms accurately analyzing phrases like "strong sales" or "loss of contracts."
Financial-Focused BERT:We fine tuned BERT with financial data,improving its grasp of complex sentences.
Improved Scores:Our model achieved lower MSE and MAE,showing fewer errors and better predictions.
Financial Term Accuracy:Our aspect embeddings correctly linked sentiment,recognizing "high AstraZeneca" as positive and "contract Morrisons" as negative.

VI. User Interface Designing

The web application is a Flask-based tool that enables users to perform sentiment analysis on text they input. The app provides an initial user interface where users can enter a set of sentences (up to 12) and select a model for sentiment analysis (In Fig.6). Once they submit their input, the application processes this data on the backend by extracting the text and chosen model, and then applies sentiment analysis to determine the overall sentiment of the text. This analysis calculates both a sentiment polarity (such as positive, negative, or neutral) and a numeric sentiment score.

After completing the analysis, the application prepares the results, including the sentiment polarity and score, and displays them to the user on a results page(In fig.7). This setup allows users to experience a seamless flow: they enter text, initiate analysis, and receive feedback in the form of sentiment information, all through a simple web interface. By organizing the process into clear steps, the app enables an intuitive experience for users wanting to understand the emotional tone of their text.

VII. Conclusion

- Summary of Key Findings
  We improved financial sentiment analysis by combining aspect based sentiment classification with better handling of numerical data.Traditional models often miss the importance of numbers or specific financial terms so we developed a model with a numerical encoder,aspect focused embeddings and a DigitCNN module.This led

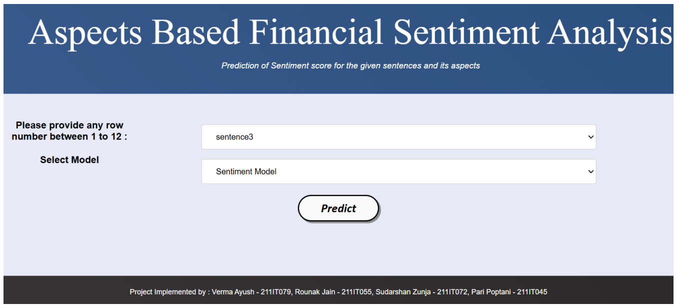

Fig. 6. User Interface Design input

Fig. 7. User Interface Design output

to better MSE and MAE scores showing more accurate sentiment predictions.

- Interpretation of Findings Our model effectively captured the link between numbers and sentiment,accurately identifying key sentiment terms and their context.This approach matched our expectations and aligned with research that highlights the need for domain-specific NLP adaptations.Sentences with rare language posed challenges,suggesting room for further data refinement.
- Implications This research is valuable for financial NLP, showing that targeted aspect embeddings and numerical understanding enhance accuracy. It provides useful insights for analysts and NLP professionals aiming for better, more industry-specific sentiment models. Our findings support using specialized techniques alongside general models like BERT for improved results.
- Acknowledgement of Limitations: Limitations: Our model relied on fine-tuned datasets, which may not work well with different financial texts or regional styles. The numerical encoder handled structured data well but needs improvement for complex numbers and outliers. Despite this, aspect embeddings and better number handling clearly improved financial sentiment analysis.
- Future Research: Improve the numerical encoder for outliers and broader datasets. Add industry-specific features like trends or macroeconomic data. Apply these methods to other fields, such as healthcare or marketing, and explore multi-lingual embeddings and real-time updates for global use.

References

[1] C. Qin, C. Yu, Y. Meng and J. Chang, "A Numeral and Affective Knowledge Enhanced Network for Aspect-based Financial Sentiment Analysis," 2023 IEEE 35th International Conference on Tools with Artificial Intelligence (ICTAI), Atlanta, GA, USA, 2023, pp. 926-933, doi: 10.1109/ICTAI59109.2023.00139. keywords: Knowledge engineering;Sentiment analysis;Analytical models;Text analysis;Redundancy;Companies;Benchmark testing;financial sentiment analysis;numeral understanding;domain knowledge;deep learning;natural language processing,

[2] Hitkul, & Shahid, Simra & Singhal, Shivangi & Mahata, Debanjan & Kumaraguru, Ponnurangam & Shah, Rajiv Ratn. (2020). Aspect-Based Sentiment Analysis of Financial Headlines and Microblogs. 10.1007/978-981-15-1216-2_5.

[3] Yusuf, Kabir & Ogbuju, Emeka & Abiodun, Taiwo & Oladipo, Francisca. (2024). A Technical Review of the State-of-the-Art Methods in Aspect-Based Sentiment Analysis. Journal of Computing Theories and Applications. 2. 67-78. 10.62411/jcta.9999.

[4] A. Botta, P. V. Mohini, A. Nooka Raju and C. Balaji, "Deep Learning for Sentiment Analysis in Financial Markets," 2024 5th International Conference on Recent Trends in Computer Science and Technology (ICRTCST), Jamshedpur, India, 2024, pp. 233-238, doi: 10.1109/ICRTCST61793.2024.10578359. keywords: Deep learning;Sentiment analysis;Recurrent neural networks;Social networking (online);Finance;Predictive models;Market research;Sentiment Analysis;Deep Learning;Natural Language Processing (NLP);Neural Networks;Financial Sentiment Analysis;Market Prediction,

[5] Sohangir, S., Wang, D., Pomeranets, A. *et al.* Big Data: Deep Learning for financial sentiment analysis. *J Big Data* **5**, 3 (2018). https://doi.org/10.1186/s40537-017-0111-6

[6] Financial Aspect-Based Sentiment Analysis using Deep Representations arXiv:1808.07931

[7] Sergio Consoli, Luca Barbaglia, Sebastiano Manzan, Fine-grained, aspect-based sentiment analysis on economic and financial lexicon, Knowledge-Based Systems, Volume 247, 2022, 108781, ISSN 0950-7051, https://doi.org/10.1016/j.knosys.2022.108781. (https://www.sciencedirect.com/science/article/pii/S0950705122003677)

[8] Matteo Lengkeek, Finn van der Knaap, Flavius Frasincar, Leveraging hierarchical language models for aspect-based sentiment analysis on financial data, Information Processing Management, Volume 60, Issue 5, 2023, 103435, ISSN 0306-4573, https://doi.org/10.1016/j.ipm.2023.103435. (https://www.sciencedirect.com/science/article/pii/S0306457323001723)

[9] Hitkul Jangid, Shivangi Singhal, Rajiv Ratn Shah, and Roger Zimmermann. 2018. Aspect-Based Financial Sentiment Analysis using Deep Learning. In Companion Proceedings of the The Web Conference 2018 (WWW '18). International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 1961–1966. https://doi.org/10.1145/3184558.3191827

[10] Yusuf, Kabir Ogbuju, Emeka Abiodun, Taiwo Oladipo, Francisca. (2024). A Technical Review of the State-of-the-Art Methods in Aspect-Based Sentiment Analysis. Journal of Computing Theories and Applications. 2. 67-78. 10.62411/jcta.9999.

[11] Ali Derakhshan, Hamid Beigy, Sentiment analysis on stock social media for stock price movement prediction, Engineering Applications of Artificial Intelligence, Volume 85, 2019, Pages 569-578, ISSN 0952-1976, https://doi.org/10.1016/j.engappai.2019.07.002. (https://www.sciencedirect.com/science/article/pii/S0952197619301666)

[12] Stock Market Analysis with Text Data: A Review arXiv:2106.12985

[13] S. Bouktif, A. Fiaz and M. Awad, "Augmented Textual Features-Based Stock Market Prediction," in IEEE Access, vol. 8, pp. 40269-40282, 2020, doi: 10.1109/ACCESS.2020.2976725. keywords: Stock markets;Sentiment analysis;Feature extraction;Predictive models;Companies;Machine learning;Biological system modeling;Machine learning;model stacking;sentiment analysis;stock movement direction prediction;textual features extraction;tweets mining,

[14] Kelvin Du, Frank Xing, and Erik Cambria. 2023. Incorporating Multiple Knowledge Sources for Targeted Aspect-based Financial Sentiment Analysis. ACM Trans. Manage. Inf. Syst. 14, 3, Article 23 (September 2023), 24 pages. https://doi.org/10.1145/3580480 Aspect-Based Financial Sentiment Analysis