Real-time Twitter data sentiment analysis to predict the recession in the UK using Graph Neural Networks

Avleen Malhi
WMG
University of Warwick
United Kingdom
avleen.malhi@warwick.ac.uk

Mohammad Naiseh
Department of Computing and Informatics
Bournemouth University
United Kingdom
mnaiseh1@bournemouth.ac.uk

Kunal Jangra
Department of Computing and Informatics
Bournemouth University
United Kingdom
s5435089@bournemouth.ac.uk

***Abstract*—The global economy is rapidly contracting, and as more countries experience recessions, it is crucial to have reliable tools for predicting them and figuring out the variables that influence how well an economy is doing. Twitter and other social media platforms have become important informational resources for forecasting market movements and identifying potential future threats. In this study, sentiment analysis is performed on real-time Twitter data to forecast a recession for the UK economy. Twitter feeds on recession are fed into Azure to pre-processes the data, and to produce insights on the recession's contributing factors. A model to accurately forecast the recession using a Graph Neural Network (GNN) is created on the processed data. The main contribution of the research is the creation of a framework for forecasting the UK recession using GNN on real-time Twitter data. The purpose of the study is to give insights into the variables influencing the UK's economic health and to pinpoint reliable recession forecasting techniques. Policymakers, economists, and companies wishing to track the UK economy in real time may find the research findings interesting. Overall, the work has important ramifications for forecasting recessions and keeping tabs on economic circumstances utilising Twitter, GNN, and Azure Databricks.**

***Index Terms*—Sentiment analysis, UK recession, real-time twitter data, Graph neural networks**

I. Introduction

The world economy is contracting quickly, and it is anticipated that this trend will continue as more countries experience recessions. High energy prices, skyrocketing inflation, rising interest rates, and a weak global economy have all conspired to raise the possibility that the UK economy may go into recession. Therefore, it is crucial to create reliable tools for recession forecasting and figuring out what influences the health of the economy. Twitter and other social media platforms have become important informational resources for forecasting market movements and identifying potential future threats. Twitter is the ideal source of data since it allows users to express ideas and engage in real-time debate among economists.

The necessity to forecast and identify the variables that influence the health of the UK economy, particularly during recessions, served as the impetus for this research article. A particular type of neural network called graph neural network (GNN) can be used to model complex relationships between entities and to make predictions based on these relationships. This research aims to forecast the UK recession and identify the factors affecting the likelihood of a recession by utilising real-time Twitter data and modelling the relationships among data using Graph Neural Networks (GNN) for sentiment analysis. The major contributions of the work are as follows:

- Developing GNN-based sentiment analysis to predict the onset and severity of UK recession.
- Performance comparison of GNN-based sentiment analysis with LSTM-based sentiment analysis in predicting economic recessions using Twitter feeds.
- Identifying the key factors affecting the economic recession in the UK, and their correlation based on the insights generated from the data pipeline.

II. Related Research

Sentiment analysis, commonly referred to as opinion mining, is a branch of natural language processing that focuses on locating and extracting subjective data from text, such as feelings, views, and attitudes [1]. In this section, we review the literature on the application of Graph Neural Networks (GNNs) in sentiment analysis.

Graph Neural Networks (GNNs) offer several advantages that make them particularly useful in economic data and predicting recession. Economic data often involves complex networks of relationships among various entities such as industries, sectors, and economic indicators [2]. GNNs are well-suited for capturing these interconnected structures and extracting meaningful patterns and insights from them. Also, economic data often includes diverse types of information [3], such as time-series data, textual data from sources like social media, and relational data representing connections between entities.

The Graph Convolutional Network (GCN) is a well-liked GNN architecture for sentiment analysis [4]. The Graph Attention Network (GAT), proposed by [5], is another GNN architecture for sentiment analysis. It has the ability to selectively attend to various nodes in a graph, allowing it to understand the significance of various words or phrases in a sentence for predicting sentiment.

TABLE I
OVERVIEW OF EXISTING METHODS

Reference	Method	Main Contribution
(Socher et al., 2013)	Recursive Neural Tensor Network (RNTN)	Utilizes a tree structure to represent the relationships between words in a sentence.
(Li et al., 2017)	Graph Convolutional Network (GCN)	Learns low-dimensional embeddings to represent nodes in a graph and propagates information from neighbouring nodes to predict sentiment.
(Veličković et al., 2018)	Graph Attention Network (GAT)	Selectively attends to different nodes in a graph to capture the importance of different words or phrases in a sentence for predicting sentiment.
(Li et al., 2020)	Heterogeneous graph-based approach	Incorporates both word-level and aspect-level information for sentiment analysis using a heterogeneous graph structure.

The Recursive Neural Tensor Network (RNTN) developed by [6] is one of the most used methods in graph-based sentiment analysis. A tree structure is used by RNTN, a deep learning methodology, to describe the connections between words. The Graph Convolutional Network (GCN), a sort of graph neural network that has been used in sentiment analysis, is an alternative method [4].

Overall, by creating a unique framework for forecasting recessions in the UK economy using real-time Twitter data and GNN, this research focuses on recession prediction and social media analysis [7]. The research will also aid in the creation of a data pipeline for processing social media data in real time. The study could offer insightful information on the elements influencing the UK's economic health as well as suggest reliable techniques for recession prediction which might be useful to policymakers, economists, and companies.

III. METHODOLOGY

The project's general strategy and methods are described in the methodology section. A real-time data pipeline's design and implementation, feature extraction and normalisation, data collecting, and data pre-processing are all included. The project adopts a methodical approach to guarantee the accuracy and dependability of the data.

A. Data Collection

Real-time Twitter feeds [8] were used to anticipate a recession in the UK economy as part of the data-gathering procedure for this study. I used the Twitter API to extract tweets containing terms associated with the UK and recession in order to gather pertinent Twitter data. The study's key terms were ”UK” and ”recession.” Azure Databricks and Azure Event Hubs were used to organise and process the data. Data scientists, engineers, and analysts can work together in a collaborative setting with Azure Databricks, an analytics platform powered by Apache Spark. A big data streaming technology called Azure Event Hubs makes it possible to gather enormous amounts of data from several sources, including social media sites like Twitter. The data was continually gathered and analysed in real-time using real-time Twitter data, which was downloaded in CSV and gave insights into the elements impacting the UK's economic health. For the purpose of gathering, processing, and analysing the data, Azure Databricks and Azure Event Hubs were used as a reliable data pipeline. Overall, real-time Twitter data and Azure technologies were used in the data-collecting procedure.

Identifying the data sources is the initial stage in constructing a real-time data pipeline, as shown in Fig. 1. In this instance, the Twitter API is used to get the data. The data is restricted to those tweets that had the terms ”UK” and ”recession.” A cloud-based solution called Azure Event Hubs can manage massive streams of streaming data in real time.

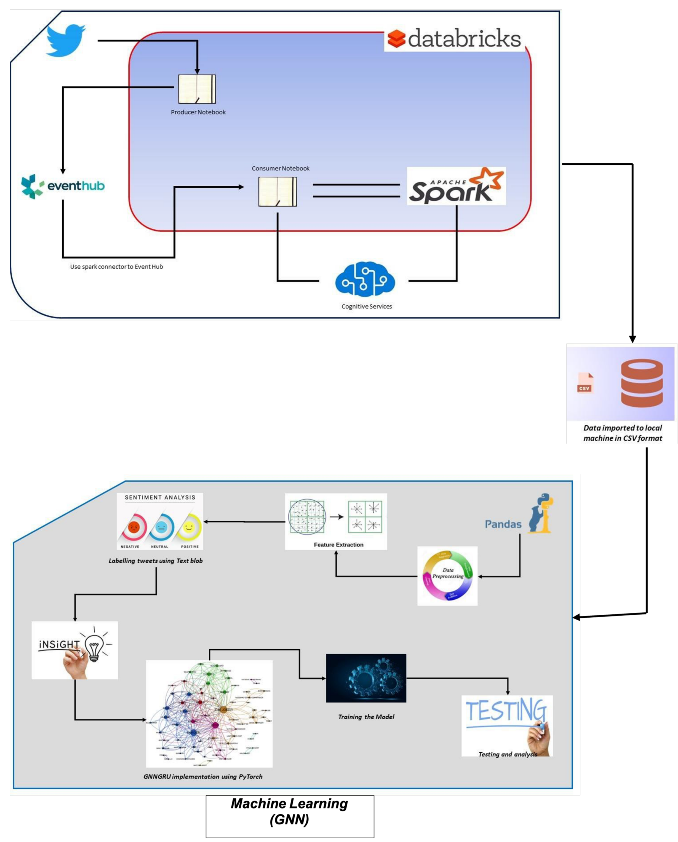

Fig. 1. The methodology adopted in the proposed research

The important information from the tweets, including the individual who posted it, their location, the time, and date it was sent, and the emotion score is collected. After that, the data is standardised to make sure it is uniform and comprehensible across various characteristics.

B. Data Pre-processing

The potential biases in data collection process are eliminated by data pre-processing techniques. The first stage in data pre-processing is to eliminate any extra columns or data points that are not needed for the study. Starting with the ”Content” column, which holds the raw tweet text in this project, all other columns are removed from the code. The data is easier to handle and deal with as a result. The ”Content” column will now, for clarity's sake, be renamed to ”Tweets raw.” This improves the code's readability and makes it simpler to grasp what the column contains. Any mentions (usernames beginning with ”@”) are then taken out of each tweet by the code. Additionally, the code eliminates any tweets that contain NaN values or no text. Any duplicate tweets are then deleted by the code.

Pre-processing the text comes after the data has been cleaned. Pre-process text, a function defined in the code, performs the following on each tweet - Removes mentions, hashtags, retweets, hyperlinks, and emojis; Removes numbers and special characters; Converts to lowercase; Tokenize; remove stop words; join words into single string. This process standardizes the text and removes any noise that may affect the analysis. It also prepares the text for feature extraction and normalization.

Sentiment labelling is the last stage in the data pre-processing process. The code designates a function named label sentiment that calculates a sentiment score for each tweet using the Text Blob library. A high score reflects a cheerful attitude, whereas a low score reflects a pessimistic outlook. This phase is crucial since it gives the sentiment analysis model's target variable. In general, pre-processing the data is an essential step in every NLP activity. It entails preparing the data for analysis by cleaning and preparing it to be accurate and consistent.

C. Feature Extraction and Normalization

Feature extraction and normalization are important steps in machine learning that help to pre-process data and make it suitable for training models. In the context of natural language processing (NLP), these steps involve converting text data into numerical representations that can be fed into machine learning algorithms.

1) Feature Extraction: Utilizing pre-trained word embeddings is one well-liked method of feature extraction in NLP. The vector representations of words known as word embeddings are discovered by learning from a huge quantity of textual data. They may be used to represent the meaning of phrases and texts because they capture the semantic links between words. Pre-trained word embeddings from the Google News dataset are used in the project's code. These 300-dimensional embeddings were developed using a sizable corpus of news items.

A vocabulary of the terms that appear most frequently in the collected tweets is built once the embeddings have loaded. Each word in the vocabulary is mapped to an index in the embedding matrix using a dictionary. The code tokenizes the text and turns it into collections of word indices in order to translate the text data into numerical representations. After that, zeros are added to the sequences to make sure they are of the same length. The generated numerical representations can be utilised as training data for neural network-based machine learning techniques.

2) Normalisation: The process of re-scaling data to have a mean of zero and a standard deviation of one is called normalisation. Because it can increase the performance of models by making the data more consistent and simpler to learn, normalisation is a crucial step in machine learning. To guarantee that word embeddings in NLP have a constant scale across many dimensions, normalisation is done. Figure 2 shows the length of Tweets after normalization.

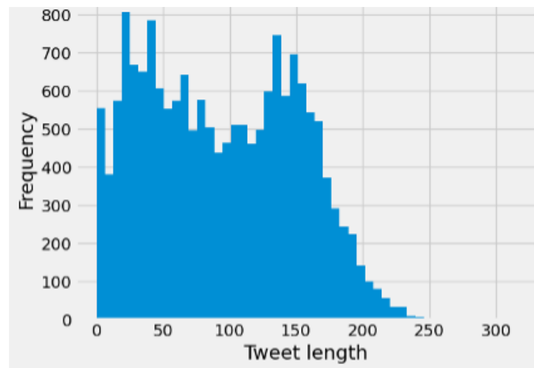

Fig. 2. Tweets length distribution

IV. Method Description

Graph recurrent units (GRUs) are used by the GNNGRU model to analyse sentiments in graphs. The GNNGRU model extracts features by pre-processing input tokens into dense vector representations using pre-trained embeddings, encoding the series of embeddings into a fixed-length sentence representation using a bidirectional GRU layer and an attention mechanism, and then processing the sentence representation through a fully connected layer to produce a set of logits. The model is trained using a loop over the specified number of epochs, with a training set and validation set used for each epoch. For each epoch, the model's accuracy is calculated and printed.

A. Graph-based sentiment analysis using GNNGRU

In the GNNGRU model, the feature extraction is mainly done through the following steps. The initial stage in the embedding layer is to create a dense vector representation for each character in the input phrase. The Embedding layer, which accepts a pre-trained embedding matrix as input and returns the associated embedding vectors for each word in the input phrase, is used to do this. The next stage is to encode the succession of embedding vectors into a fixed-length sentence representation using the bidirectional GRU layer. A bidirectional GRU layer is used for this, which provides the hidden states for each token in both the forward and backward directions after receiving the embedding vectors as input. The bidirectional GRU layer generates a series of hidden states, one for each token in the input phrase, to the attention layer. An attention mechanism is used on the hidden states to produce a fixed-length sentence representation. This entails assigning each concealed state a weight that indicates its significance in the sentence representation. Using a linear layer and a softmax activation, the weights are calculated. Sentence Representation: After computing the attention weights, a weighted sum of the hidden states is computed. As a result, the input phrase is represented by a single vector. Finally, a fully connected layer is applied to the sentence representation to create a collection of logits that may be used to forecast the emotion of the input sentence. Let the input sequence of length L be denoted as $x = (x1, x2, ..., xL)$, where each x_i is a one-hot encoded vector of size V (vocabulary size).

- Embedding Layer: Let the pre-trained word embeddings be denoted as E, which is a matrix of size $V \times E$. Initialize an embedding matrix W with these pre-trained embeddings:

$$W = X \tag{1}$$

The embedding layer maps each one-hot encoded vector x_i to its corresponding embedding vector x_i:

$$E(x_i) = W x_i \tag{2}$$

The embedded sequences tensor is of shape (N, L, E).

- Bidirectional GRU Layer: Let the hidden size of the GRU layer be denoted as H. Initialize a bidirectional GRU layer with the input size as E, hidden size as H, number of layers as 2(for bidirectionality), and a dropout probability as p:

$$\overrightarrow{h_t} = GRU(e(x_t), \overrightarrow{h_{t-1}} \tag{3}$$

$$\overleftarrow{h_t} = GRU(e(x_t), \overleftarrow{h_{t-1}} \tag{4}$$

$$h_t = [\overrightarrow{h_t}; \overleftarrow{h_t}] \tag{5}$$

where $[\overrightarrow{h_t}; \overleftarrow{h_t}]$ denotes concatenation, and h_t is the hidden state at time t. The GRU layer takes the embedded sequences tensor as input and produces a GRU output tensor of shape $(N, L, 2H)$.

- Attention Layer: A linear layer is used to compute the attention weights for each element in the GRU output tensor. Let W be the weight matrix of the linear layer, which is a matrix of size $2H \times 1$. The attention layer takes the GRU output tensor as input and outputs a tensor of shape $(N, L, 1)$:

$$\alpha_t = softmax(W^T h_t) \tag{6}$$

where α_t is the attention weight for time t.

- Sentence Representation: The weighted sum of the GRU output tensor is computed using the attention weights to obtain a sentence representation tensor of shape $(N, 2H)$:

$$v = \sum_{t=1}^{L} \alpha_t h_t \tag{7}$$

- Fully Connected Layer: Let the number of classes be denoted as C. A linear layer is initialised with input size as $2H$ and output size as C:

$$y = W_2 v + b_2 \tag{8}$$

where W_2 is a matrix of size $C * 2H$ and $b_2$is a vector of size C.

- Dropout Layer: A dropout is applied to the embedded sequence tensor, the GRU output tensor, and the sentence representation tensor to prevent overfitting:

$$x_i Bernou\tilde{l}i(1-p) \tag{9}$$

$$e(x_i) \leftarrow x_i \bigodot e(x_i) \tag{10}$$

$$h_t \leftarrow x_i \bigodot h_t \tag{11}$$

$$v \leftarrow x_i \bigodot v \tag{12}$$

$$\tag{13}$$

where $\bigodot$ denotes element-wise multiplication.

The training portion of the code constructs Data Loader objects for the training and validation sets, initialises lists to hold the training and validation losses and accuracies, and initialises the number of epochs and batch size for training. A loop over the specified number of epochs is then used to train the model. The model is switched to training mode for each epoch, and the total loss is initialised. Once the gradients have been zeroed out and a forward pass of the model has been run on the batch, the loop repeats this process for all the batches in the training set. After computing the loss using the loss function, updating the parameters is done via a backward pass. The batch loss is included in the epoch's overall loss calculation.

V. Results

The GNN Gated Recurrent Unit sentiment analysis model was tested on test data to categorise tweets as positive or negative. The model was validated at 94.12% and tested at 94.19%. The model's predictions were close to the actual labels, showing good accuracy. The GNNGRU model outperformed LSTM model and earlier sentiment analysis approaches on this dataset.

A. Evaluation of sentiment analysis model on test data

A test set of tweets was used to evaluate the model, which was trained using the GNN Gated Recurrent Unit (GRU) architecture, a form of Recurrent Neural Network (RNN). The evaluation's goal was to evaluate the model's performance and accuracy in identifying tweets' sentiment as positive or negative in graph format, as illustrated in figure 3. The model was trained using a dataset of more than 20,000 tweets during the training phase, with labels indicating whether each tweet was favourable or negative. The accuracy gained during training was 90.14% and in validation was 94.12%. To avoid the model being overtrained, validation accuracy was employed as a criterion for ending the training.

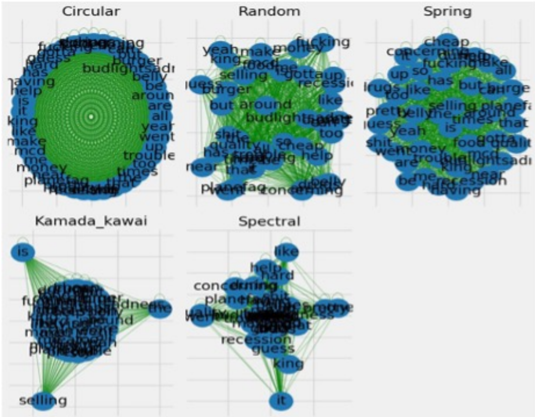

Fig. 3. Graph representation of Graph tweet data

A test dataset of tweets with labels indicating whether each tweet was favourable or negative is utilised to assess how well the model performed on the test data. On the test dataset, the model's accuracy was 94.19 percent, which is in line with the validation accuracy attained during training. The model accurately identified 2032 negative tweets and 1193 positive tweets, with 110 incorrect negative predictions and 89 false positive predictions, according to the confusion matrix generated for the test data.

The model's high degree of accuracy in the test data suggests that it is a trustworthy tool for sentiment analysis of tweets. The evaluation of the sentiment analysis model described in this research shows, in general, the potential of deep learning approaches for the analysis of textual data and the extraction of insightful information. Due to the model's excellent accuracy on the test data, it is possible to utilise it for a variety of purposes, such as political sentiment analysis, brand reputation management, and social media monitoring.

B. Performance evaluation and comparative analysis

My GNN GRU model was compared with LSTM created using the same hyperparameters and the comparative analysis is presented in the table II. This suggests that the GRU model performs better than the LSTM model for sentiment analysis on this dataset. The GRU model has a greater capacity to generalise to unknown data as seen by the higher accuracy it achieved on the test data as well as the lower loss and higher accuracy it achieved on the validation set. One of the reasons for the high performance of GRU model is that the GRU model contains fewer parameters than other models, which makes it easier to train and more effective to employ. The GRU model also has a gating mechanism that enables it to selectively update and forget information. This feature may be especially helpful for sentiment analysis, where specific words or phrases may be more significant than others. Overall, the findings demonstrate that the GRU model outperforms the LSTM model as a model for sentiment analysis. However, more study is required to assess how well these models perform on different datasets and to investigate different deep learning architectures for sentiment analysis which is a future scope of the work. Future research may also concentrate on methods like data augmentation, fine-tuning pre-trained language models, or using outside information sources to enhance the performance of the models.

TABLE II
GNN COMPARISON WITH EXISTING METHOD LSTM

Model	Training acc.	Validation acc.	Loss	Testing Acc.
GNNGRU	90.14%	94.12%	0.0616	94.19%
LSTM	88.66%	85.76%	0.2878	62.56%.

TABLE III
STATE-OF-THE-ART COMPARATIVE ANALYSIS OF THE PROPOSED METHOD

Method	Dataset Size	Accuracy
Recursive Neural Tensor Network (RNTN)	11K	85.4%
Graph Convolutional Network (GCN)	23K	85.7%
Graph Attention Network (GAT)	39K	87.4%
Heterogeneous graph-based approach	52K	87.9%
GNNGRU Approach	21k	90.14%

Deep learning techniques including recurrent neural networks (RNNs), convolutional neural networks (CNNs), and general neural networks (GNNs) have lately been used to analyse sentiment with promising results III. For sentiment analysis tasks where the relationships between words and their contexts may be represented as a graph, GNNs have demonstrated tremendous promise in modelling structured data, such as graphs.

C. Analysis of factors affecting the likelihood of a recession in the UK

The insights have been analysed in this section that give a thorough overview of the variables influencing the risk of a UK recession. Figure 4 shows the public's perception of the possibility of a recession as a bar graph. According to the graph, almost 6,000 individuals predict there won't be a recession, whereas 11,000 people predict one. It shows that there is a substantial gap in popular opinion on the likelihood of a UK recession. A bar graph in Figure 5 displays the most popular phrases said by those who anticipate a recession. Recession, inflation, and economy are the top three terms, followed by market (around 900), think (about 800) and price (Around 650). The significance of these elements in shaping the public's view of the risk of a UK recession is further supported by this graph. Our study leads us to the conclusion that the economy, inflation, and recession are the main variables influencing the risk of a recession in the UK. The way the market, think, and price are affected by these elements, determines how the general public views them. The word cloud and bar graph also demonstrate the importance of banks and financial crises in determining the likelihood of a UK recession.

These variables interact with one another and can either increase or decrease the chance of a recession. For instance, inflation may lower the value of money, which may cause the

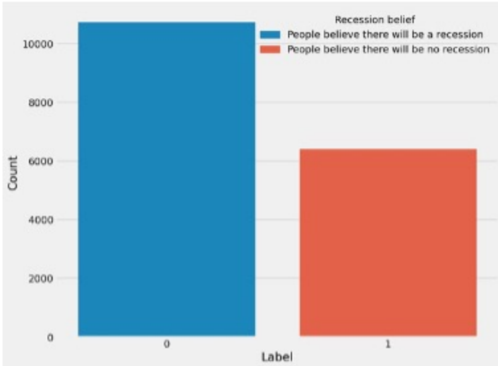

Fig. 4. Predicting recession based on sentiments of people

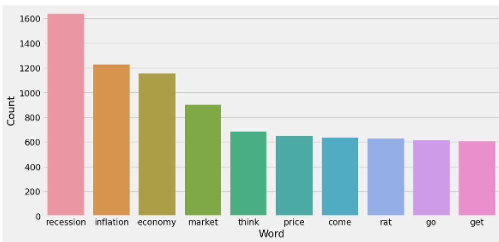

Fig. 5. Factors responsible for recession

stock market to decline. A decline in the stock market may trigger a financial crisis, which might worsen the chances of a recession and have severe effects on the economy. On the other hand, government action to tame inflation and boost the economy can lessen the chance of a recession. Overall, our study shows how critical it is to keep an eye on and address the major variables that affect the chance of a UK recession. Policymakers and market participants may proactively take action to lessen the effects of a recession on the UK economy by knowing these elements and how they are interconnected.

VI. Conclusion and Future Directions

This study helped to use sentiment analysis and real-time Twitter data to forecast a recession in the UK economy. The research developed a pipeline that takes in twitter data related to the recession and feeds it to Azure in real-time. A sentiment analysis model is built using GNNGRU architecture which achieves an accuracy of 94.19% which is much higher when compared with state-of-the-art models. The research also helped in visualising the results to generate insights on factors affecting the recession. Overall, this research helps to build deep learning methods for studying textual material and gleaning insightful information. The use of sentiment analysis models in a variety of disciplines, including social media monitoring, brand reputation management, and political sentiment analysis, can offer insightful information about the general public's viewpoint and facilitate decision-making.

The performance of sentiment analysis models might be enhanced in the future through the application of transfer learning strategies as well as data augmentation. The future work also includes incorporating outside knowledge sources like ontologies or knowledge graphs which might enhance the precision and readability of sentiment analysis models. In future, we also plan to consider other countries or regions for study by including different datasets.Additionally, incorporating outside knowledge sources like ontologies or knowledge graphs could enhance the precision and readability of sentiment analysis models.

References

[1] B. Pang, L. Lee *et al.*, "Opinion mining and sentiment analysis," *Foundations and Trends® in information retrieval*, vol. 2, no. 1–2, pp. 1–135, 2008.
[2] R. H. Day, "Complex economic dynamics: obvious in history, generic in theory, elusive in data," *Journal of Applied Econometrics*, vol. 7, no. S1, pp. S9–S23, 1992.
[3] L. Einav and J. Levin, "Economics in the age of big data," *Science*, vol. 346, no. 6210, p. 1243089, 2014.
[4] Z. Li, Z. Cui, S. Wu, X. Zhang, and L. Wang, "Fi-gnn: Modeling feature interactions via graph neural networks for ctr prediction," in *Proceedings of the 28th ACM international conference on information and knowledge management*, 2019, pp. 539–548.
[5] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio, Y. Bengio *et al.*, "Graph attention networks," *stat*, vol. 1050, no. 20, pp. 10–48 550, 2017.
[6] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proceedings of the 2013 conference on empirical methods in natural language processing*, 2013, pp. 1631–1642.
[7] R. Miao, Y. Huang, and Z. Zhang, "A social media knowledge retrieval method based on knowledge demands and knowledge supplies," *Mathematics*, vol. 11, no. 14, p. 3154, 2023.
[8] Y. Qi and Z. Shabrina, "Sentiment analysis using twitter data: a comparative application of lexicon-and machine-learning-based approach," *Social Network Analysis and Mining*, vol. 13, no. 1, p. 31, 2023.
[9] R. Nyman and P. Ormerod, "Predicting economic recessions using machine learning algorithms," *arXiv preprint arXiv:1701.01428*, 2017.
[10] V. Kumar and P. Nanda, "Social media marketing: practices, opportunities and challenges for msmes," *International Journal of Business Information Systems*, vol. 44, no. 2, pp. 268–284, 2023.
[11] L. Yue, W. Chen, X. Li, W. Zuo, and M. Yin, "A survey of sentiment analysis in social media," *Knowledge and Information Systems*, vol. 60, pp. 617–663, 2019.
[12] N. K. Singh, D. S. Tomar, and A. K. Sangaiah, "Sentiment analysis: a review and comparative analysis over social media," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, pp. 97–117, 2020.
[13] V. V. Vydiswaran, Y. Zhang, Y. Wang, and H. Xu, "Special issue of bmc medical informatics and decision making on health natural language processing," pp. 1–3, 2019.
[14] J. Bollen, H. Mao, and X. Zeng, "Twitter mood predicts the stock market," *Journal of computational science*, vol. 2, no. 1, pp. 1–8, 2011.
[15] B. Liu, *Sentiment analysis and opinion mining*. Springer Nature, 2022.
[16] A. Pak, P. Paroubek *et al.*, "Twitter as a corpus for sentiment analysis and opinion mining." in *LREc*, vol. 10, no. 2010, 2010, pp. 1320–1326.
[17] A. Severyn and A. Moschitti, "Twitter sentiment analysis with deep convolutional neural networks," in *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*, 2015, pp. 959–962.
[18] X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," *Advances in neural information processing systems*, vol. 28, 2015.
[19] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.
[20] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.
[21] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.
[22] R. v. d. Berg, T. N. Kipf, and M. Welling, "Graph convolutional matrix completion," *arXiv preprint arXiv:1706.02263*, 2017.