Query-Based Summarization and Sentiment Analysis for Indian Financial Text by leveraging Dense Passage Retriever, RoBERTa, and FinBERT

Numair Shaikh, Jayesh Patil and **Sheetal Sonawane**
Department of Computer Engineering, SCTR's Pune Institute of Computer Technology

Abstract

With the ever-expanding pool of information accessible on the Internet, it has become increasingly challenging for readers to sift through voluminous data and derive meaningful insights. This is particularly noteworthy and critical in the context of documents such as financial reports and large-scale media reports. In the realm of finance, documents are typically lengthy and comprise numerical values. This research delves into the extraction of insights through text summaries from financial data, based on the user's interests, and the identification of clues from these insights.

This research presents a straightforward, all-encompassing framework for conducting query-based summarization of financial documents, as well as analyzing the sentiment of the summary. The system's performance is evaluated using benchmarked metrics, and it is compared to State-of-The-Art (SoTA) algorithms. Extensive experimentation indicates that the proposed system surpasses existing pre-trained language models.

1 Introduction

In recent years, the volume of data that we handle has increased exponentially, with vast quantities of textual data at our disposal. It can be a daunting task for a user to sift through such a significant amount of text in order to gain valuable insights. The process of reading through such extensive amounts of text places a considerable strain on short-term memory, ultimately leading to unfavorable outcomes. Text summarization is an example of a natural language processing (NLP) task that entails producing a condensed version of a text while retaining the essential information. There are two primary methods for summarizing text: Extractive Summarization and Abstractive Summarization. Extractive summarization entails selecting relevant sentences from the source text and combining them to form a summary. Conversely, abstractive summarization involves creating new sentences that capture the essential points of the original text. The financial services sector is expected to undergo a compound annual growth rate of 7.4% by 2026, which is twice the rate of growth that has been observed in the previous three years[1]. As a result, experts highly recommend utilizing artificial intelligence to improve various aspects of financial services. The finance industry generates and consumes vast amounts of information in the form of finance blogs/articles, reports, and other types of textual data. The major challenges in financial analytics include an extensive vocabulary, a large quantity of noisy unstructured data, and a robust model. Kumar and Ravi (2016) in their research, showed that Financial text analysis plays a critical role in making market decisions. Modaresi et al. (2017) revealed that automatic summarization systems considerably enhance employees' workflow in media monitoring and media response analysis through reduced processing time. This observation can be applied to the financial sector, where the proposed system caters to the needs of thousands of users dealing with a vast amount of textual data available on the internet. The system generates short summaries of text in 2–3 sentences, making it easier for users to read particular financial text documents. Furthermore, with an available sentiment score, users can make more informed decisions quickly, improving efficiency and yielding better results. This sentiment analysis approach has been demonstrated to improve the performance of recommender systems in previous studies (Dang et al., 2021).

In this research, our primary focus has been on the summarization and sentiment analysis of financial data. In order to achieve effective summarization of financial text documents, we have

[1] financial-services-global-market-opportunities

proposed a specialized system. Our model relies on data collected from diverse sources, including news and official reports. Given the voluminous and complex nature of financial documents, it is crucial to extract pertinent information by employing a query-based approach. This method allows for the retrieval of relevant documents based on user requirements and facilitates the generation of summaries. Sentiment analysis refers to the process of extracting subjective information, such as opinions, emotions, and beliefs, from text. It is an incredibly powerful tool that can provide valuable insights into people's thoughts and feelings about a particular topic. In the financial (Baker and Wurgler, 2006) and cryptocurrency (Kraaijeveld and De Smedt, 2020) trading sectors, sentiment analysis is used to track investor sentiment, identify emerging trends, and make informed investment decisions. Many researchers have found that sentiment analysis is a key feature in stock price prediction, which has proven to be highly successful. According to the research conducted by Nguyen et al. (2015), market sentiment plays a significant role in predicting stock prices, and this is why stock market prediction is so beneficial. It is widely acknowledged that carrying out effective and precise sentiment analysis for financial data can be an arduous undertaking. To address this challenge, we have put forward the implementation of a model that has undergone pre-training on vast quantities of financial data FinBERT (Araci, 2019). This has enabled us to generate precise sentiment scores for the summaries produced by our model.

The primary contribution of our research can be outlined as follows

1. Real-time financial data is acquired from the internet, and we construct an information retrieval system using Dense Passage Retriever (Karpukhin et al., 2020).

2. We investigate various extractive summarization algorithms and implement the RoBERTa model (Liu et al., 2020) for query-based extractive summarization.

3. We conduct a thorough analysis of the algorithm's performance and compare it to the SoTA summarization algorithms.

4. We analyze the sentiment of the generated summary, to offer a concise perspective on the utilization of sentiment in financial news.

To achieve efficient and prompt document retrieval, we have utilized the Dense Passage Retriever (DPR) technique. DPR is capable of representing queries and documents through dense vectors, which have proven effective in capturing the semantic meaning of the text. Moreover, DPR comes pre-trained on a vast corpus of textual data, simplifying the process of fine-tuning and enhancing the model's efficiency as opposed to creating a new one from scratch for every new task. Query and document indexing are employed by DPR to store the dense representations of the documents, leading to swift retrieval of relevant documents. For indexing these representations, we have opted for the FAISS (Facebook AI Similarity Search) (Johnson et al., 2017) document store. For document summarization, we have utilized the RoBERTa model, which is based on BERT. BERT based models are pre-trained on textual data, making them easy to fine-tune. Additionally, RoBERTa greatly enhances BERT's performance by utilizing a larger corpus of pre-training data and implementing other model improvements. For sentiment analysis, we have employed FinBERT, which has already undergone extensive training on a vast amount of financial text, specifically fine-tuned for sentiment analysis on our dataset.

The structure of the research paper is such that section 2 encompasses a discussion on related work pertaining to financial data summarization and sentiment analysis. Moving on, section 3 delves into the proposed work, which entails a methodology for the collection of data, document indexing, data query processing, search, and sentiment analysis. The outcomes of the research are expounded upon in section 4, while section 5 provides a summary of the work, along with the conclusion and future prospects.

2 Literature Review

In this section, we have briefly outlined other approaches that have been used for extractive text summarization and sentiment analysis.

One of the simplest techniques used for generating a summary for a given text is the Term Frequency—Inverse Document Frequency (TF-IDF) technique. Ramos (2003) in their research, showed that, TF-IDF can be used to determine the relevance of words in documents. The words can then be ranked according to their TF-IDF score and used to generate a summary for the given text. Qaiser and Ali (2018) have further explored the strengths and weaknesses of TF-IDF. The work of Ahuja and Anand (2017) was extended by Afsharizadeh et al. (2018) through the addition of further features to their paper. The initial work involved the extraction of various features from a text document, combining them to generate a sentence score. The resulting sentences were then ranked according to their score, allowing for the creation of a summary specific to the given document. To create a multi-document summary, individual text summaries were merged based on cosine similarity. Subsequently, the top N sentences are combined to form a summary. Du and Huo (2020) extracted word and sentence features in their system and employed the genetic algorithm to assign weights to each feature. The evaluation of each generation is carried out using ROUGE parameters, and fuzzy logic is then used to generate summaries. Wang et al. (2021) proposed a graph-clustering-based framework for generating financial news summaries. Their model incorporates a graph structure that considers cross-sentence relations and text embeddings to enhance representation and emphasize significant sentences. Li et al. (2020) presented a system that comprises an encoder, an attention level decoder, and a copy mechanism, with the encoder employing a GRU model. Kharde. and Sonawane (2016) presented an extensive survey on sentiment analysis and opinion mining utilizing conventional Machine Learning techniques as well as cross-domain and cross-lingual methods. They discovered that approaches like SVM and Naïve Bayes exhibit the highest level of accuracy. Additionally, they noted that combining ML models with the opinion lexicon method notably enhances the accuracy of sentiment classification. Vanetik et al. (2022) presented two approaches to summarize financial text: Abstractive and Extractive. The initial phase of the model identifies the commencement of the continuous narrative section in the document. Subsequently, the model extracts the most significant sentences from the continuous narrative section. To identify the beginning of the continuous narrative section, the first stage of the model employs a similarity search. In contrast, the second stage of the model employs a range of features to extract the most important sentences. These features encompass the sentence's position in the document, length, topic, and sentiment. Despite this, the model achieved a ROUGE score of below 0.5 on all ROUGE-N parameters. Abdaljalil and Bouamor (2021) presented two approaches for summarization, namely sentence-based and section-based summarization. The first approach involves the use of BERT for embedding and classification, which resulted in superior performance compared to the baseline models. As for the second approach, BERT was utilized for vector embeddings and Weighted-Section-Clustering for classification, achieving a ROUGE-L F1 score metric of 0.31.

Kraaijeveld and De Smedt (2020) in their research, explored the predictive power of public Twitter sentiment in forecasting cryptocurrency prices. The authors utilize a dataset of more than 100 million tweets discussing Bitcoin and Ethereum, and they apply a range of sentiment analysis techniques to extract sentiment from the tweets. These techniques include lexicon-based sentiment analysis, machine learning-based sentiment analysis, and a hybrid approach that combines both methods. The authors' findings indicate that public Twitter sentiment can serve as a valuable resource for investors seeking to make informed investment decisions. The study conducted by Leung et al. (2023) employs sentiment analysis in forecasting cryptocurrency prices via market tweets sentiment. A dataset of historical tweets was utilized to evaluate the proposed system, which was able to accurately predict the price of cryptocurrencies. The models examined were Text Blob and CNN-LSTM, with the latter yielding a higher degree of accuracy.

This literature survey highlights different techniques for extractive text summarization, including TF-IDF, feature extraction with ranking, graph-based approaches, and BERT-based models. It also emphasizes the value of sentiment analysis in forecasting cryptocurrency prices using Twitter data. BERT-based models have shown superior performance compared to other methods. However, feature extraction and sentence ranking continue to be widely employed for extractive summarization, with efforts to address ambiguity and redundancy in sentences to enhance performance.

3 Proposed Work

Our model, as depicted in Figure 1, offers financial text summarization and sentiment analysis through a system of four distinct modules.

1. Data Collection Module
2. Document Indexing Module

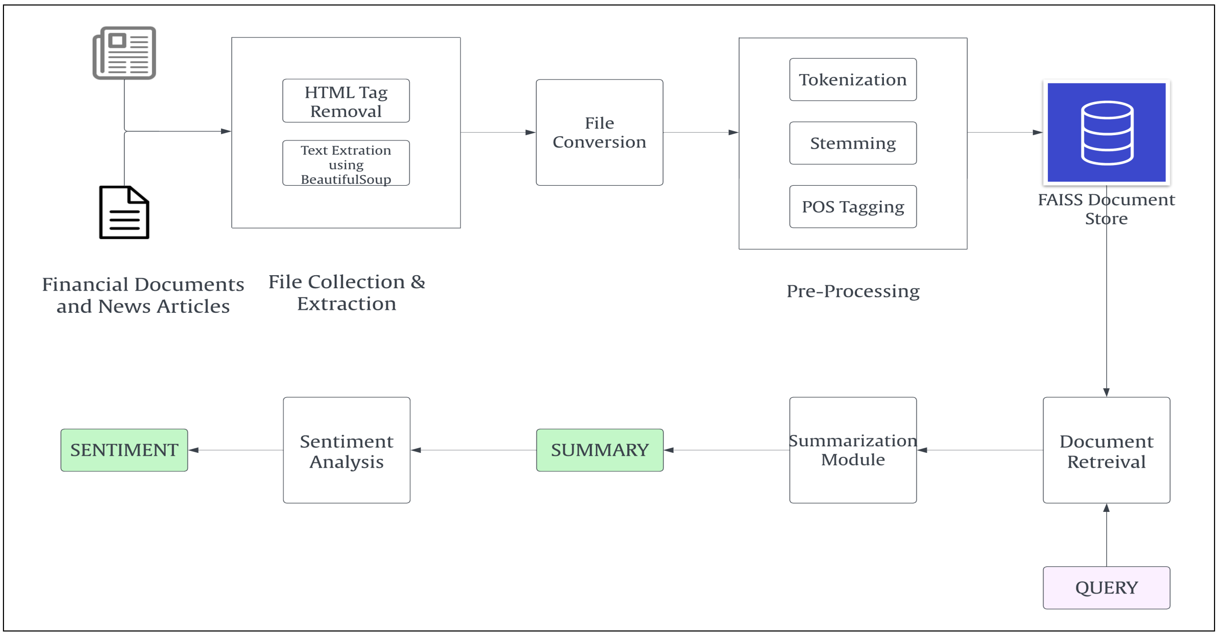

Figure 1: Architecture Diagram

3. Data Query Processing and Retrieval (Search) Module
4. Sentiment Analysis Module

3.1 File Conversion and Pre-Processing Module

Before the data can be indexed in the indexing module, it is crucial to standardize the input documents. In order to ensure uniform processing, all incoming documents are converted into text format in our system. To achieve this, we have incorporated support for a variety of file types such as PDFs, CSV files, and other formats, ensuring comprehensive coverage for conversion needs. The subsequent step involves pre-processing the documents by eliminating headers and footers, white spaces, as well as empty lines within the document. In order to extract relevant data from online news articles, we must acquaint ourselves with the various HTML tags, classes, and IDs in use to extract and convert the data into text files, which can then be pre-processed.

3.2 Document Indexing

The files that have been converted and pre-processed are subsequently stored within the document store. This repository is a database that houses both the textual data and any associated meta-data. The document store comprises an SQL database as well as an FAISS (Facebook AI Similarity Search) (Johnson et al., 2017) index. The FAISS index is a vector index that aids in narrowing down the search space for k-nearest neighbors of a vector, thereby enabling faster similarity searches between vectors. This index vector also assists in reducing query times during retrieval, given that an index is already being maintained for a document.

3.3 Document Retrieval and Summarization Module

The information retrieval and summarization module consists of two components, namely the document retrieval module and the summarization module.

3.3.1 Document Retrieval Module

The Retrieval Module operates by accepting a user query and subsequently retrieving pertinent documents. It functions as an efficient filter that promptly peruses through the FAISS document store and sieves out potential documents that possess relevance to the query. We propose the use of Dense Passage Retriever (DPR) for document retrieval. DPR utilizes dense representations in the process of retrieval. In order to achieve efficient retrieval of passages, DPR employs the concept of indexing, wherein all passages are indexed in a low-dimensional and continuous space. To build indexes for each of the input documents/text, DPR utilizes BERT-based encoders (Devlin et al., 2019a) to convert text passages into real-valued dense vectors. In order to facilitate real-time processing, we have implemented a FAISS index-based document storage module, which generates indices for the documents prior to this module. DPR utilizes separate encoders and decoders for the search query and the underlying documents, both of which are BERT-based. This is primarily due to the fact that queries are considerably shorter than the documents. The ranking of documents is determined by the dot product similarity between query and document embeddings. DPR returns the top K vectors (documents) based on this ranking.

It has been demonstrated by Karpukhin et al. (2020) that DPR surpasses BM25's (Amati, 2009) performance by a remarkable 9-19%. Therefore, we have confidently opted for DPR as our retrieval model for the document retrieval module.

DPR can be mathematically represented as follows.

$$\begin{aligned} &D_N = \{D_1, D_2, \ldots, D_n\} \\ &S_N = \{S_1, S_2, \ldots, S_n\} \ni \{S_1 > S_2 > \ldots > S_n\} \\ &S_i = \text{sim}(Q, D_i), where\ D_i \in D_N \end{aligned} \tag{1}$$

In Equation 1, let D_N be the set of N documents and S_N be the set of similarity scores calculate for each document D_i in D_N. S_i is the Similarity Score of the ith document. The function to calculate the cosine similarity score is given by,

$$\text{sim}(q, p) = E_Q(q)^\top E_P(p) \tag{2}$$

where, $q =$ Query Text, $p =$ Document Text, $E_P = BERT$ Embedding for underlying document, and $E_Q = BERT$ Embedding for input query. The documents are then ranked according to similarity score and the top K documents are selected to form a set of K documents D_K.

3.3.2 Summarization Module

The summarization module generates extractive summaries for the documents selected by the Retrieval Module. It utilizes the SoTA transformer model RoBERTa (Vaswani et al., 2017). The primary reason for selecting a BERT-based model like RoBERTa is that BERT has already been pre-trained on a vast amount of data, making models based on BERT easier to fine-tune, ultimately producing highly accurate and precise results. Research conducted by (Liu and Lapata, 2019) and (Zhang et al., 2019) both provide ample evidence that BERT-based embeddings result in significantly higher ROUGE-N scores and textual summaries with unparalleled accuracy.

In their paper showcasing RoBERTa, Liu et al. (2020) improved upon BERT by utilizing dynamic masking, eliminating next sentence prediction (NSP), and pre-training the model on more extensive data in larger batches over longer sequences. RoBERTa's query-based text summarization method works by taking in a text document or passage, and producing a summary based on a query.

To generate a summary, the model utilizes BERT Devlin et al. (2019b) to encode the query and passage as matrices with dimensions of $q \times d$ and $p \times d$, respectively, where d is the encoding's dimensionality. It calculates a similarity score by computing a dot product of the embeddings/matrices. These similarity scores are utilized to compute a probability distribution for each sentence. The top K sentences with the highest probability are selected to create the summary. This process can be represented mathematically as follows: Let Q be the input query, represented as a sequence of q words. Let P be the input passage, represented as a sequence of p words. RoBERTa generates embeddings for both the query and the input document/passage, which are represented by E_Q and E_P, respectively. Both of these embeddings are matrices with dimensions of $q \times d$ and $p \times d$. Let us consider S as the set of probable sentences in document P. We proceed to compute the similarity score for each sentence belonging to set S. Suppose s_i denotes a candidate sentence or phrase in set S. RoBERTa encodes this sentence and generates an encoding represented by E_{s_i}, which is a vector of dimension d.

$$S_{s_i} = E_{s_i} \cdot (\alpha \times E_Q + \beta \times E_P)^\top \tag{3}$$

In Equation 3, the hyperparameters α and β serve to regulate the significance of the query and text, respectively, in the computation of the similarity score. As a means of evading partiality towards lengthier sentences or phrases, the similarity scores are subsequently normalized by dividing them by the length of each sentence S_i. The Softmax normalization function, represented in Equation 6, is utilized for the purpose of normalizing the similarity scores. This results in the derivation of a probability distribution which is applied over the set of sentences present in S.

$$P_{s_i} = \frac{e^{S'_{s_i}}}{\sum_{j=1}^{n} e^{S'_{s_j}}} \tag{4}$$

The model proceeds to choose the foremost K sentences from the given set S by arranging them in descending order based on their probability scores P_{S_i}, and then selecting the top K sentences.

3.4 Sentiment Analysis Module

For the purpose of analyzing sentiment in the summaries generated, we have utilized FinBERT, a pre-trained BERT model that has undergone extensive training on financial data such as news articles, financial reports, and social media posts pertaining to finance. FinBERT's primary function is to enhance the accuracy of NLP tasks within the financial domain, and in this case, we have employed it to precisely forecast the sentiment of each summary produced by the data query and retrieval module. FinBERT achieves this by fine-tuning the pre-trained BERT model with a vast corpus of financial documents, enabling it to comprehend the domain-specific language and terminology used in the financial industry. To clearly demonstrate the working of FinBERT, let us take an example of the textual input, denoted by T, which is a sequence of n tokens and is generated in the above section as summaries. The FinBERT representation of this text is presented by a matrix of nxd dimensions, which is denoted by E_T. To proceed further, we need to consider a weight matrix W of dxk dimensions, where k corresponds to the number of sentiment labels. Weighted sum S is determined by the summation of the encoded text weighted by the weight matrix, W as depicted in Equation 5.

$$\begin{aligned} S &= E_T \cdot W \\ P &= softmax(S) \end{aligned} \tag{5}$$

In the above equation, P denotes the probability distribution across K sentiment labels. The sentiment label with the highest probability is predicted as the overall sentiment of the input text. The weight matrix W can be learned from a training dataset by leveraging methods like maximum likelihood or cross-entropy loss to reduce the error between the expected sentiment and actual sentiment labels. By fine-tuning the model using domain-specific training data and appropriately modifying the weight matrix W, the model can be enhanced. In their research, Araci (2019) observed that FinBERT demonstrated improved accuracy of 15-20% when compared to models like LM dictionary, NB, SVM, RF, CNN, and LSTM.

4 Results and Analysis

4.1 Dataset Statistics

For the purpose of our analysis, we have utilized a news dataset that has been collected for 49 conglomerates. This dataset comprises 2773 news articles that have been sourced from over 900+ verified online news sources. The data is dynamically collected from news aggregators such as Google News, with a refresh rate set at 3 hours. Once collected, the data is added to our database and an index is generated. Upon executing queries, summaries, and sentiments are generated. Figure 2 visualizes the various sources of training data and real-time analysis. The dataset employed is proprietary, characterized by ground truth summaries that have been meticulously generated and validated through human intervention. Each data point within the dataset is accompanied by an associated sentiment score.

Figure 2: Financial News Sources

Additional statistics about our data, such as the mean word count per article, mean word length, and mean sentence length, are displayed in Table 1.

Table 1: Additional Dataset Statistics

Statistic	Value
Mean word count per news article	387.29
Mean word length (in characters)	5.34
Mean sentence length (in words)	16.74
Duration for data collection (in days)	30

4.2 Evaluation of Retrieved Documents

Given that unlabeled data has been utilized for document retrieval, the module for document retrieval has been subject to evaluation through assessing the average similarity score of the documents retrieved in a specific instance. This system has undergone examination in the context of five diverse queries, with the five most relevant documents being taken into account for the computation of the similarity score. The mean similarity score of the retrieved documents was determined to be **0.64**.

4.3 Analysis of Text Summaries

To assess the summaries produced by our model, we have concentrated on examining documents acquired from a select number of Banks and IT companies. In order to make comparisons with an expert summary, we have utilized the ROUGE (Recall-Oriented Understudy for Gisting Evaluation) (Lin, 2004) parameters. Our model's performance has been evaluated in relation to summaries generated by SoTA models such as BART (Lewis et al., 2020) and BERT combined with K-Means (Miller, 2019).

Table 2: The evaluation of our proposed model, conducted against an expert summary.

Parameter	ROUGE-1	ROUGE-2	ROUGE-L
Recall	0.652	0.393	0.526
Precision	0.408	0.229	0.298
F1 Score	0.501	0.289	0.380

Table 3: Comparison of our proposed model's recall score, calculated for ROUGE-N scores, when evaluated against several SoTA models.

Model	ROUGE-1	ROUGE-2	ROUGE-L
Our Model	0.652	0.393	0.526
BART	0.485	0.296	0.409
BERT	0.266	0.154	0.219

Table 2 displays the specific ROUGE parameters that were utilized to compare the summary generated by our model with the summaries created by experts, which serve as the ground truth. The comparison between the summaries produced by our model and those generated by SoTA pre-trained transformer-based models is presented in Table 3, utilizing the ROUGE-N metric.

The ROUGE-L score being higher in comparison to the ROUGE-2 score indicates a more robust conformity in the arrangement of word sequences between the summary generated and the reference summary. This signifies that the model proficiently grasps the overall structure and progression of the summary, closely aligning with the reference summary at the sentence level.

Table 4 portrays the performance predicated on the BLEU metric. The table explicitly exhibits the BLEU-1, BLEU-2, BLEU-3, and BLEU-4 scores, with each score reflecting the precision of 1-gram, 2-grams, 3-grams, and 4-grams, respectively. The BLEU scores provide valuable insights into the degree of similarity between the text generated by our model and the reference sentences. However, we noted that the BLEU scores were considerably lower than the ROUGE precision values. This is primarily due to the imposition of the brevity penalty during the BLEU calculation. The brevity penalty is critical in ensuring equitable evaluation and mitigating any bias towards shorter translations or summaries over longer ones. The penalty is more severe if the length of the summary generated by our model strays significantly from the reference summary.

Table 4: The performance of the model we have proposed, assessed by evaluating it using BLEU parameters

BLEU Parameter	Our Model	BART	BERT
BLEU - 1	0.1319	0.0294	0.1140
BLEU - 2	0.0758	0.0168	0.0585
BLEU - 3	0.0731	0.0144	0.0472
BLEU - 4	0.0292	0.0093	0.0224

4.4 Sentiment Analysis Results

To assess the efficacy of FinBERT, we conducted a comparative analysis against other established sentiment analysis models, including VADER (Valence Aware Dictionary and sEntiment Reasoner) (Hutto and Gilbert, 2014), FLAIR (Akbik et al., 2019), GPT3.5[2], SiBERT (Hartmann et al., 2023), and XLNet (Yang et al., 2019). We fine-tunes the models for sentiment analysis using the Adaptive Moment Estimation (Adam) (Kingma and Ba, 2014) optimiser. The loss function used is the Binary Cross-Entropy with the number of Epochs set to 5 and a batch size of 32. To calculate the class probabilities, we apply the softmax activation, as desscribed in Equation 6.

$$P_{y_i} = \frac{e^{z_i}}{\sum_{j=1}^{n} e^{z_j}} \tag{6}$$

where, z_i score for class i, the scores represent the model's confidence in each class for the given input. Subsequently, the class with the highest calculated probability is selected as the predicted class for the input sequence.

$$y_{predicted} = \arg\max_i P(y_i) \tag{7}$$

[2] OpenAI GPT 3.5

Table 5: Comparison of results obtained on the financial phrasebank dataset

Metric	FinBERT	VADER	FLAIR	GPT3.5	SiEBERT	XLNet
Accuracy	0.91	0.57	0.51	0.71	0.375	0.79
Precision	0.91	0.52	0.67	0.68	0.290	0.77
F1 Score	0.89	0.48	0.50	0.69	0.40	0.81
Recall	0.88	0.51	0.56	0.76	0.648	0.79

Table 6: Comparison of results, evaluated on our labeled dataset

Metric	FinBERT	VADER	FLAIR	GPT3.5	SiEBERT	XLNet
Accuracy	0.86	0.43	0.55	0.589	0.85	0.76
Precision	0.88	0.51	0.59	0.585	0.76	0.75
F1 Score	0.86	0.43	0.55	0.582	0.79	0.76
Recall	0.84	0.59	0.59	0.627	0.83	0.79

Table 5 presents a thorough and detailed comparison between FinBERT and SoTA sentiment analysis models, which were specifically applied to the financial phrasebank dataset. Based on the data presented in the tables, it is evident that FinBERT outperforms the SoTA models when it comes to sentiment analysis on this particular dataset. Moreover, Table 6 presents a comparable analysis that underscores the superior performance of FinBERT as compared to SoTA sentiment analysis models. The outcomes depicted in these tables reaffirm the conclusion that FinBERT consistently delivers better results than the SoTA models when applied to a real-time news dataset.

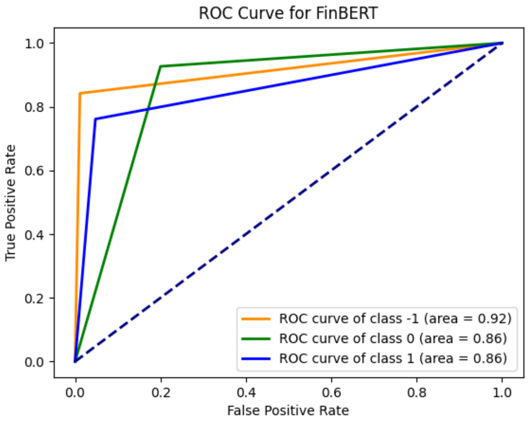

Figure 3: ROC Curve for FinBERT

The ROC Curve for Sentiment Analysis is depicted in Figure 3. The model's performance at all potential classification thresholds is quantified by the area under the ROC curve (AUC). In this instance, the AUC values for labels *-1*, *0*, and *1* are 0.92, 0.86, and 0.86, respectively. This data suggests that the model excels at distinguishing between true negatives and false positives for all three labels. The high AUC value indicates that the model can accurately classify the majority of the observations.

5 Conclusions

In this research, we have presented a workflow for summarizing financial news and documents. We got the financial news and texts from the internet, specifically from sources/aggregators. We used DPR for document retrieval and RoBERTa to create concise and accurate summaries based on user queries. Furthermore, we also added sentiment analysis using FinBERT to provide sentiment scores, analyzing financial text sentiment allows for identification of overall market sentiment, positive/negative trends, and impact of specific events on sentiment, aiding investors and financial professionals in making informed decisions.

We utilized various metrics, such as ROUGE and BLEU scores, to gauge the caliber of the generated summaries in contrast to the ground truth summaries created by experts. The findings exhibited that our approach consistently surpassed the baseline methods, indicating its proficiency in capturing pertinent information and creating concise and precise summaries. To assess the sentiment analysis module, we juxtaposed the accuracy, precision, and F1 scores of FinBERT with those of other SoTA sentiment analysis models, thus illustrating that our model outperforms them.

In conclusion, we have conducted research on query-based summarization for financial news and documents that demonstrates a comprehensive workflow utilizing cutting-edge techniques such as DPR, RoBERTa, and FinBERT. Through the integration of these approaches, we have created a robust system capable of extracting pertinent summaries that are personalized to user queries, while additionally conducting sentiment analysis to capture the overall market mood.

Declarations

The authors have no relevant financial or non-financial interests to disclose. The authors have no competing interests to declare that are relevant to the content of this article.

References

Samir Abdaljalil and Houda Bouamor. 2021. An exploration of automatic text summarization of financial reports. In *Proceedings of the Third Workshop on Financial Technology and Natural Language Processing*, pages 1–7, Online. -.

Mahsa Afsharizadeh, Hossein Ebrahimpour-Komleh, and Ayoub Bagheri. 2018. Query-oriented text summarization using sentence extraction technique. In *2018 4th International Conference on Web Research (ICWR)*, pages 128–132.

Ravinder Ahuja and Willson Anand. 2017. Multi-document text summarization using sentence extraction. In *Artificial Intelligence and Evolutionary Computations in Engineering Systems*, pages 235–242, Singapore. Springer Singapore.

Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter, and Roland Vollgraf. 2019. FLAIR: An easy-to-use framework for state-of-the-art NLP. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 54–59, Minneapolis, Minnesota. Association for Computational Linguistics.

Giambattista Amati. 2009. *BM25*. Springer US, Boston, MA.

Dogu Araci. 2019. Finbert: Financial sentiment analysis with pre-trained language models. *CoRR*, abs/1908.10063.

Malcolm Baker and Jeffrey Wurgler. 2006. Investor sentiment and the cross-section of stock returns. *The journal of Finance*, 61(4):1645–1680.

Cach N. Dang, María N. Moreno-García, and Fernando De la Prieta. 2021. An approach to integrating sentiment analysis into recommender systems. *Sensors*, 21(16).

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019a. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019b. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Yan Du and Hua Huo. 2020. News text summarization based on multi-feature and fuzzy logic. *IEEE Access*, 8:140261–140272.

Jochen Hartmann, Mark Heitmann, Christian Siebert, and Christina Schamp. 2023. More than a feeling: Accuracy and application of sentiment analysis. *International Journal of Research in Marketing*, 40(1):75–87.

C. Hutto and Eric Gilbert. 2014. Vader: A parsimonious rule-based model for sentiment analysis of social media text. *Proceedings of the International AAAI Conference on Web and Social Media*, 8(1):216–225.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2017. Billion-scale similarity search with gpus. *CoRR*, abs/1702.08734.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Vishal Kharde. and S.S. Sonawane. 2016. Sentiment analysis of twitter data: A survey of techniques. *International Journal of Computer Applications*, 139(11):5–15.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *arXiv e-prints*, page arXiv:1412.6980.

Olivier Kraaijeveld and Johannes De Smedt. 2020. The predictive power of public twitter sentiment for forecasting cryptocurrency prices. *Journal of International Financial Markets, Institutions and Money*, 65:101188.

B. Shravan Kumar and Vadlamani Ravi. 2016. A survey of the applications of text mining in financial domain. *Knowledge-Based Systems*, 114:128–147.

Man-Fai Leung, Lewis Chan, Wai-Chak Hung, Siu-Fung Tsoi, Chun-Hin Lam, and Yiu-Hang Cheng. 2023. An intelligent system for trading signal of cryptocurrency based on market tweets sentiments. *FinTech*, 2(1):153–169.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Haozhou Li, Ying Wang, Xu Mou, and Qinke Peng. 2020. Sentiment classification of financial microblogs through automatic text summarization. In *2020 Chinese Automation Congress (CAC)*, pages 5579–5584.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3730–3740, Hong Kong, China. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Ro{bert}a: A robustly optimized {bert} pretraining approach.

Derek Miller. 2019. Leveraging bert for extractive text summarization on lectures.

Pashutan Modaresi, Philipp Gross, Siavash Sefidrodi, Mirja Eckholf, and Stefan Conrad. 2017. On (commercial) benefits of automatic text summarization systems in the news domain: A case of media monitoring and media response analysis. *ArXiv*, abs/1701.00728.

Thien Hai Nguyen, Kiyoaki Shirai, and Julien Velcin. 2015. Sentiment analysis on social media for stock movement prediction. *Expert Systems with Applications*, 42(24):9603–9611.

Shahzad Qaiser and Ramsha Ali. 2018. Text mining: Use of tf-idf to examine the relevance of words to documents. *International Journal of Computer Applications*, 181(1):25–29.

Juan Enrique Ramos. 2003. Using tf-idf to determine word relevance in document queries.

Natalia Vanetik, Marina Litvak, and Sophie Krimberg. 2022. Summarization of financial reports with tiber. *Machine Learning with Applications*, 9:100324.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Jun Wang, Jinghua Tan, Hanlei Jin, and Shuo Qi. 2021. Unsupervised graph-clustering learning framework for financial news summarization. In *2021 International Conference on Data Mining Workshops (ICDMW)*, pages 719–726.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Haoyu Zhang, Jingjing Cai, Jianjun Xu, and Ji Wang. 2019. Pretraining-based natural language generation for text summarization. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 789–797, Hong Kong, China. Association for Computational Linguistics.