Scraping & Summarizing Financial News for Investors - Project Milestone

SMITKUMAR B. SARAIYA, University of Calgary, Canada

1 Introduction

This project explores the application of natural language processing using NLTK and SpaCy for financial news summarization and sentiment analysis. The goal is to develop an automated pipeline that scrapes financial news from platforms, like MoneyControl and MarketBeat, processes the data, and delivers actionable condensed information to investors. The system will incorporate sentiment analysis using tools such as FinBERT[1] or VADER to provide additional market insight, as sentiment has been shown to significantly impact stock prices.

2 Proposed Solution

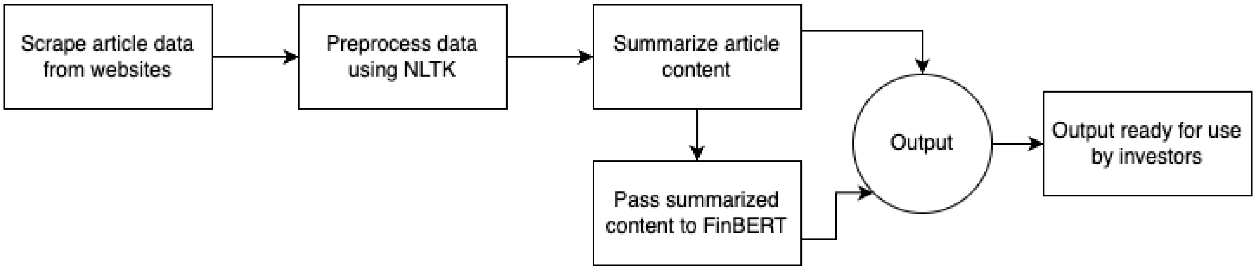
Scrape article data from websites
Preprocess data using NLTK
Summarize article content
Pass summarized content to FinBERT
Output
Output ready for use by investors

Fig. 1. System architecture showing the proposed complete pipeline for financial news text summarization and sentiment analysis with FinBERT for investor use.

2.1 Summary of choices

The proposed solution includes the usage of libraries in Python including, but not limited to, natural language processing language libraries such as NLTK and SpaCy, web scraping libraries, such as requests and BeautifulSoup4, and finally libraries to help handle and preprocess data such as numpy and pandas.

Data are gathered by scraping financial news websites. The data are collected and stored in csv files under columns labeled as title, link, content no matter where the data is sourced from. This is to allow the same text summarization techniques to be applied on data collected from different websites, in turn collecting all the data together in a single format that can then be finally used

Author's Contact Information: Smitkumar B. Saraiya, smitkumar.saraiya@ucalgary.ca, University of Calgary, Canada.

by investors to make financial decisions and finally for the FinBERT model. In addition to these libraries, the FinBERT model will be used for sentiment analysis after text summarization.

2.2 Using FinBERT

The FinBERT model is perfect for this application as it is specially developed to use with financial data. This FinBERT model is based on BERT-base. BERT's model architecture is a multi-layer bidirectional Transformer encoder [2], the number of layers (i.e., Transformer blocks) as L, the hidden size as H, and the number of self-attention heads as A [2]. BERT-base (L=12, H=768, A=12, Total Parameters=110M) [2]. We use the original BERT code 3 to train FinBERT on our financial corpora with the same configuration as BERT-Base. Following the original BERT training, we set a maximum sentence length of 128 tokens, and train the model until the training loss starts to converge. Wethen continue training the model allowing sentence lengths up to 512 tokens [3].

Sentiment classification is conducted by adding a dense layer after the last hidden state of the [CLS] token [1]. A CLS token being a Classify token. Then, the classifier network is trained on the labeled sentiment dataset [1]. Since the model is already very accurate, with 97% accuracy on the subset of Financial PhraseBank with 100% annotator agreement [1], there is not a need to modify the model architecture in order to increase accuracy for my specific application.

3 Current progress & Going forward

Currently, the web scraping module along with a specialized pipe to use the FinBERT model has been setup and tested. The implementation can be run to get news articles periodically. Investment decisions are complex, with consumers plausibly considering a fund's overall investment strategy, risk level, recent returns, domestic versus international exposure, and underlying composition [4]. A widely cited example demonstrates that consumers are more likely to choose fuel-efficient vehicles, consistent with their preferences, when viewing "gas consumption per 100 miles" versus the standard "miles per gallon" metric and label [5]. What this means is that investment decisions can be driven by visual aids which can be later implemented in for example a web application using the output from the processed data.

4 Challenges

One challenge encountered was that the initially chosen sources of data, like Bloomberg, CBS, and investopedia among others, had a lot of obstructions to data collection such as paywalls, restricted access to web scrapers which came as a major roadblock. After testing several websites, moneycontrol.com and marketbeat.com were selected as they did not have any of the above mentioned hindrances. More sources can be considered, but that comes with the drawback of having more scraped data to go through as all these data sources will have different HTML structures, which require tailored processing to extract data.

References

[1] Dogu Tan Araci. 2019. *FinBERT: Financial sentiment analysis with pre-trained language models*. Master's thesis. University of Amsterdam. arXiv:1908.10063 https://doi.org/10.48550/arXiv.1908.10063

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, 2019. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. arXiv:1810.04805. https://doi.org/10.48550/arXiv.1810.04805

[3] Yi Yang, Mark Christopher, Siy UY Allen Huang, 2020. *FinBERT: A Pretrained Language Model for Financial Communications*. School of Business and Management, Hong Kong University of Science and Technology. arXiv:2006.08097. https://doi.org/10.48550/arXiv.2006.08097

[4] Brian Scholl, et al. 2023. *A Picture Is Worth a Thousand Dollars: Visual Aids Promote Investor Decisions. Journal of the Association for Consumer Research*, vol. 8, no. 4, pp. 416–428. https://doi.org/10.1086/726428

[5] Richard P. Larrick and Jack B. Soll. 2008. *The MPG Illusion. Science*, vol. 320, pp. 1593–1594. https://doi.org/10.1126/science.1154983