


Portfolio of Evidence (PoE)

Programming for Data Analytics 1

Riko Wolhüter

Submission UUID: 3ff2d8b3-68c5-c5a5-c637-14812f6f71bd

Total Score: 7 %  Low risk

Total Number of Reports 1	Highest Match 7 % safe assign.docx	Average Match 7 %	Submitted on 02/07/23 19:25 GMT+2	Average Word Count 3,186 Highest: safe assign.docx
------------------------------	--	----------------------	---	--

Attachment 17 %

Word Count 3,186
Source: safe assign.docx

Global database (5)4 %

- 6

Student paper
- 4

Student paper
- 5

Student paper
- 7

Student paper
- 1

Student paper

Internet (2)3 %

- 2

globalilluminators
- 3

gatech

Top sources (3)

- 2

globalilluminators2 %
- 6

Student paper2 %
- 4

Student paper1 %

Contents

1.

1

Table of contents.1

2. Background. 2

3. Text Processing. 2

4. Pipelines. 2

5.

2

frequency-inverse document frequency. 3

6. Tokenization... 3

7. stemming vs lemmatisation.3

8.

3

Latent Dirichlet Allocation.3

9. Dataset chosen.4

10. Analysis to be conducted.4

11. Process used for the analysis.5

12. Evaluating and improving model.6

13. References. 6

Background

The topic of financial analysis research has paid significant attention to sentiment analysis, which is a potent natural language processing (NLP) tool. Businesses and people who work in the financial industry value the capacity to effectively comprehend emotions, remarks, and reviews that are communicated in text data, whether they are good, negative, or neutral. In this report, we examine a sizable dataset that integrates data from two reputable sources, FiQA and Financial PhraseBank, into a single, comprehensible CSV file. This condensed dataset provides a wide range of financial sentences, all of which have sentiment labels. We use a variety of ways to address the difficulty of sentiment analysis, employing both rule-based sentiment analysis and machine learning-based approaches. Our goal is to improve comprehension and implementation of sentiment analysis in financial situations by exploration of diverse approaches and new developments in text classification. Text Processing

Data analysis techniques and methods used to alter, transform, and extract useful information from textual data are referred to as text processing. In order to evaluate and comprehend the content of text documents, including articles, reviews, emails, social media posts, and more, various computer techniques and tools must be applied. (towardsdatascience. 2018) Text cleaning entails eliminating any extraneous or distracting text components such punctuation, special characters, HTML tags, and non-alphanumeric characters. Additionally, it could entail changing the text's case to lowercase, eliminating stop words (frequently used words like "and," "the," "is," etc.), and stemming or lemmatizing the text in order to break down words into their simplest forms. (towardsdatascience. 2018) Tokenization is performed on the text. The process of tokenization entails dividing the text into smaller units called tokens, which can be words, sentences, or even n-grams (sequences of related words). Tokenization is an important step since it creates the fundamental units for additional analysis. (analyticsvidhya. 2023) Text parsing is performed on the text. Text parsing entails examining the grammar and structure of the text. It aids in the recognition of word relationships, syntactic patterns, and parts of speech, all of which are helpful in the extraction of relevant information. (pluralsight. 2019) We will identify and categorize the named entities, such as names of people, groups, places, dates, and other particular phrases, this process is known as named entity recognition (NER). It aids in comprehending the context and locating important textual components. (towardsdatascience. 2018) We will determine the sentiment or opinion expressed in the text using a procedure called sentiment analysis. It entails categorizing content as good, negative, or neutral, which can be useful for deciphering user reviews, emotion on social media, or public opinion. (towardsdatascience. 2018) We will be classifying the text documents into predetermined groups or categories which is the process of text classification. It is frequently employed for applications including sentiment analysis, topic modeling, spam detection, and document classification. Text clustering does not use predefined categories; instead, it clusters comparable papers together according to their content. (towardsdatascience. 2018) We will extract and locate particular pieces of information from text documents. It could entail activities like extracting named entities, entity relationships, keyphrases, or particular patterns. (towardsdatascience. 2018) For text summarizing, we will seek out to produce a succinct and cohesive summary of a lengthy piece of text. It can be done using abstractive approaches (creating new sentences to express the key ideas) or extraction-based procedures (picking out significant sentences or phrases). (towardsdatascience. 2018) Pipelines

A pipeline is a grouping of various data processing operations or transformations that come together to create a single workflow. To organize and speed up the data pretreatment and modeling process, pipelines are frequently employed in data science and machine learning jobs. (towardsdatascience. 2020) You may combine all of these stages into a single object and apply them to the data in a logical and consistent way by using a pipeline. This keeps the code readable, lowers the chance of errors, and makes the deployment of the entire workflow simpler. (towardsdatascience. 2020) A single interface provided by pipelines makes it simpler to do cross-validation and hyperparameter tweaking on the entire workflow rather than on individual components independently. (towardsdatascience. 2020)

② frequency-inverse document frequency

Frequency-Inverse Document Frequency (TF-IDF) measures how frequently a term appears in a corpus of documents. It is frequently used in information retrieval and natural language processing to gauge how relevant a term is to a particular text. (towardsdatascience. 2019) A term's frequency in a document (TF) and the inverse document frequency (IDF) throughout the full corpus are both considered in TF-IDF. While IDF gauges a term's rarity across texts, TF gauges how frequently a term appears in a document. The TF and IDF product offers phrases that are common in a document but uncommon in the corpus a higher weight, suggesting their significance in describing the content of that document. (towardsdatascience. 2019) Tokenization

A text sequence is tokenized when it is divided up into smaller pieces known as tokens. Tokenization is the process of breaking up a text document or text string into individual words, sentences, or other meaningful units in Jupyter Notebook and Python. (analyticsvidhya. 2023) Natural language processing (NLP) applications including text categorization, information retrieval, and text production frequently involve the preprocessing phase of tokenization. You can more efficiently examine and process text by dividing it up into tokens. Depending on the particular purpose and requirements, tokens can also be phrases, characters, or other types of units in addition to words. (analyticsvidhya. 2023) Stemming vs Lemmatization

Stemming and lemmatization are methods for breaking down words into their base or root forms in the context of natural language processing (NLP) and text analysis. By removing word variations including plurals, tenses, and other forms of a word so that they can be viewed as the same term, these techniques aid in normalizing text data. (towardsdatascience. 2020) Stemming is the process of stripping suffixes from words to return them to their fundamental or root form. After stemming, the final term might not always be a legitimate word in the language. Stemming algorithms use a set of predetermined criteria to remove frequent suffixes from words in order to get to the word's base form. When the words "running," "runs," and "ran" are stemmed, the root form is "run." (towardsdatascience. 2020) Lemmatization is a more sophisticated method than stemming because it considers the morphological examination of words to identify their root forms, or lemmas. In order to accurately reduce words to their basic forms by lemmatization, considerations such the word's part of speech (noun, verb, adjective, etc.) and context are taken into account. The resulting lemma is always a grammatically correct word. For instance, lemmatizing "running," "runs," and "ran" would result in "run," the verb's base form. (towardsdatascience. 2020)

③ Latent Dirichlet Allocation

A probabilistic topic modeling method called Latent Dirichlet Allocation (LDA) is used to find hidden topics in a group of documents. It is based on the

supposition that each piece of writing in the collection is a mashup of different subjects, and that each word belongs to one of the subjects. (towardsdatascience. 2019)

Dataset chosen

The dataset's emphasis on financial terms suggests that it is designed with sentiment analysis in the financial context in mind. Applications like the analysis of investor mood, market trends, and financial news sentiment can all benefit from this context. (towardsdatascience. 2018) The Financial PhraseBank and FiQA datasets are combined in this dataset. The diversity and coverage of financial sentences are projected to grow as a result of this combination, improving the generalizability of sentiment analysis models developed using the dataset. (towardsdatascience. 2018) For data processing and analysis operations, the data is delivered in the CSV (Comma-Separated Values) file format, which is popular and simple to use. The ability to load and edit the data using a variety of programming languages and data analysis tools is now convenient for data scientists. (towardsdatascience. 2018) The dataset contains financial sentences as well as the sentiment labels that go with them. Sentiment analysis techniques requiring supervised learning are made possible by the fact that each sentence is assigned to one of three sentiment categories (positive, negative, or neutral). The availability of sentiment labels makes it possible to train and assess sentiment categorization models. (towardsdatascience. 2018) Real-World Data: The dataset contains real-world financial sentences that are probably representative of the actual content found in news articles, market reports, debates on social media, and other financial sources. Training sentiment analysis models that can manage the complexity, variability, and noise contained in genuine text data is facilitated by real-world data. This dataset gives researchers and practitioners the chance to work with actual financial text, allowing them to create algorithms that can precisely analyze sentiment in comparable real-world settings. (towardsdatascience. 2018)

Analysis to be conducted

The ability to extract important insights from textual data makes text processing and sentiment analysis useful for analyzing financial sentiment. (towardsdatascience. 2020) Understanding Investor Sentiment: Investor sentiment, which can be challenging to measure and interpret, affects financial markets. We can determine the sentiment included in financial texts, such as news stories, social media posts, and corporate reports, by using text processing and sentiment analysis algorithms. We may learn more about the general mood of market players by evaluating sentiment, and this information can help us make financial decisions and strategies. (towardsdatascience. 2020) Extracting Useful Data: Financial texts are replete with useful data, such as market movements, corporate performance, and economic indicators. The most pertinent information from these texts can be extracted with the aid of text processing techniques like tokenization, lemmatization, and the elimination of stop words. Using sentiment analysis, we can categorize the extracted data into positive, negative, or neutral sentiments, giving us a succinct assessment of the sentiment the text was trying to communicate. (towardsdatascience. 2020) The ability to recognize market trends and the effects of news on the financial markets is made possible by text processing and sentiment analysis. We can spot patterns and trends that could affect market movements by examining sentiment across a variety of news sources. For traders, investors, and financial analysts, this data can be utilized to forecast how the market will react to particular events and news releases. (towardsdatascience. 2020) Sentiment research can help with risk assessment and portfolio management by highlighting potential threats and opportunities related to particular businesses or industries. Analysts can spot changes in market sentiment and modify their investment strategy by keeping an eye on sentiment indicators. The identification of sentiment-driven abnormalities, such as irrational optimism or pessimism, which may indicate market overvaluation or undervaluation, is another benefit of sentiment analysis. (towardsdatascience. 2020) Enhancing Trading tactics: By adding sentiment indicators into quantitative models and algorithms, text processing and sentiment analysis can improve trading tactics. Traders can take advantage of emotion-driven market inefficiencies and produce alpha by incorporating sentiment data into their trading techniques. Sentiment analysis can be especially helpful in algorithmic and high-frequency trading, where real-time sentiment data can be used to make swift and informed trading decisions. (towardsdatascience. 2020) Financial sentiment analysis requires both text processing and sentiment analysis in order to extract useful insights from textual data. Financial professionals can make better decisions, manage risks, and possibly gain a competitive edge in the financial markets by comprehending investor sentiment, extracting pertinent information, spotting market trends, and incorporating sentiment indicators into investment strategies. The use of text processing and sentiment analysis techniques is still developing, providing exciting new directions for future study and advancement in the area of financial sentiment analysis. (towardsdatascience. 2020)

Process used for the analysis




Data Loading and Exploration: The "data.csv" file is used to load the dataset using the pandas package. The code then conducts exploratory data analysis (EDA) by showing the data's shape, verifying its data types, looking for null values, and investigating distinct values and their counts in the "Sentiment" column. (towardsdatascience. ④ 2020) Data preprocessing: The code converts the sentiment labels from negative to positive, neutral to positive, and zero to neutral to negative. Additionally, it defines a text preprocessing function that entails lowercase text conversion, lemmatization, stop word removal, and tokenization. The dataset's "Sentence" column receives the function's application. (towardsdatascience. 2020) The code designates a function called "model" that accepts the dataset, a boolean flag for bag-of-words (BoW) or TF-IDF vectorization, and a model (the default is logistic regression). Within the function, stratified k-fold cross-validation is used to divide the data into folds. Either CountVectorizer (BoW) or TfidfVectorizer (TF-IDF) is used to vectorize the text data. Precision scores are printed once the model has been trained and assessed on every fold. (towardsdatascience. 2020) Evaluation of the Baseline Model: The "model" function is called twice, the first time with BoW vectorization and Multinomial Naive Bayes (MNB) as the model, and the second time with TF-IDF vectorization and MNB as the model. To assess how well the baseline models are performing, classification reports are printed. (towardsdatascience. 2020) Text visualization: To depict the most popular words in the preprocessed text data, word clouds are created. (towardsdatascience. 2020) Latent Dirichlet Allocation (LDA) is used to find topics in a dataset by applying it to vectorized text data. Each topic's top words are listed. (towardsdatascience. 2020) Word2Vec Vectorization: The code creates a class called "Word2VecVectorizer" that turns text into numerical vectors using word vectors that have already been trained. The class is created using word vectors that have been loaded from a file. The Word2VecVectorizer class is used to convert the text in the dataset into word vectors, and the dataset is divided into train and test sets. (towardsdatascience. 2020) Model Training and Evaluation: ⑤ To determine the optimum hyperparameters, a support vector classifier (SVC) model is trained using grid search and cross-validation. The model is developed using the classification report after being trained on the training set. (towardsdatascience.


2020) The code accepts user input, preprocesses it, vectorizes it using Word2VecVectorizer, and then forecasts the sentiment using the trained SVC model. The output includes the anticipated mood. (towardsdatascience. 2020) Model tuning: The code first builds a function to compute the F1 score before tuning the hyperparameters for a Multinomial Naive Bayes (MNB) model using grid search with cross-validation. The best parameters are printed after the code computes the F1 score. (towardsdatascience. 2020) Evaluating and improving model

④ **Preprocessing of the Data:** The first stage entails preprocessing the data to make sure it is in a format that is appropriate for model training. This covers operations like tokenization, stopword removal, stemming or lemmatization, handling special characters, and handling numerical values. Data preparation aids in text data standardization and enhances the caliber of model input. (towardsdatascience. 2018) Training and Test Sets: A training set and a test set are created from the dataset. The test set is used to assess the models' performance after they have been trained on the training set. The split makes sure that the model is tested on hypothetical data in order to gauge its capacity for generalization. (towardsdatascience. 2018) Model Selection and Training: Several models, including Naive Bayes, Support Vector Machines (SVM), Random Forest, or deep learning models like Recurrent Neural Networks (RNN) or Convolutional Neural Networks (CNN), are taken into consideration for sentiment analysis. ⑥ **Using the training data, several models are trained, and their performance is assessed using suitable evaluation measures including accuracy, precision, recall, and F1-score.** (towardsdatascience. 2018)

Cross-Validation: K-fold cross-validation is used to ensure the models' robustness. With this method, the training data is divided into k subsets (folds), k-1 folds are used for training, and the last fold is used for validation. Each fold serves as the validation set once during the course of the next k iterations of this operation. A more accurate estimation of model performance is provided by the average performance over all folds. (towardsdatascience. 2018) Hyperparameter tuning: Before the model training process starts, parameters that are not learned from the data are set. They have a considerable effect on model performance. To investigate various combinations of hyperparameters and find the ideal configuration that produces the highest performance, methods like grid search or random search are used. Learning rate, regularization strength, batch size, and the number of layers in a deep learning model are examples of common hyperparameters. (towardsdatascience. ⑦ **2018) Evaluation Metrics:** Depending on the particular issue, the model performance is assessed using the appropriate evaluation metrics. ④ **Metrics like accuracy, precision, recall, and F1-score are frequently employed in sentiment analysis.** These metrics offer information on the model's accuracy in classifying positive, negative, and neutral moods. (towardsdatascience. 2018) Iterative Improvement: Models can be improved iteratively based on the findings of the evaluation. This could entail changing the model's architecture, expanding the size of the training set, or adding new features. Hyperparameters may also need to be adjusted. Up till a sufficient performance is obtained, the process is repeated. (towardsdatascience. 2018) Test Set Validation: After the model has passed muster according to the assessment metrics, the test set is used to validate it. This last assessment offers a fair assessment of the model's performance and its capacity to generalize to fresh, untested data. (towardsdatascience. 2018) The method makes sure that the sentiment analysis models attain improved accuracy and resilience in capturing sentiment patterns in textual data by applying relevant metrics and iteratively refining them. (towardsdatascience. 2018)

Source Matches (11)

① Student paper	100 %
Submitted paper	Original source
Table of contents.	Table of Contents
② http://globalilluminators.org/wp-content/uploads/2015/12/MISG-15-198.pdf 	83 %
Submitted paper	Original source
frequency-inverse document frequency.	Inverse document frequency
③ https://smartechn.gatech.edu/bitstream/handle/1853/62189/HUTTO-DISSERTATION-2018.pdf 	100 %
Submitted paper	Original source
Latent Dirichlet Allocation.	Latent dirichlet allocation
② http://globalilluminators.org/wp-content/uploads/2015/12/MISG-15-198.pdf 	74 %
Submitted paper	Original source
frequency-inverse document frequency Frequency-Inverse Document Frequency (TF-IDF) measures how frequently a term appears in a corpus of documents.	Inverse document frequency Tf-idf is the measure of product of term frequency and inverse document frequency

<p>③ https://smartech.gatech.edu/bitstream/handle/1853/62189/HUTTO-DISSERTATION-2018.pdf </p> <p>Submitted paper</p> <p>Latent Dirichlet Allocation</p>	<p>Original source</p> <p>Latent dirichlet allocation</p>	<p>100 %</p>
<p>④ Student paper</p> <p>Submitted paper</p> <p>2020) Data preprocessing:</p>	<p>Original source</p> <p>preprocessing the received data</p>	<p>63 %</p>
<p>⑤ Student paper</p> <p>Submitted paper</p> <p>To determine the optimum hyperparameters, a support vector classifier (SVC) model is trained using grid search and cross-validation.</p>	<p>Original source</p> <p>This model is trained using a grid search cross validation</p>	<p>66 %</p>
<p>④ Student paper</p> <p>Submitted paper</p> <p>Preprocessing of the Data:</p>	<p>Original source</p> <p>preprocessing the received data</p>	<p>75 %</p>
<p>⑥ Student paper</p> <p>Submitted paper</p> <p>Using the training data, several models are trained, and their performance is assessed using suitable evaluation measures including accuracy, precision, recall, and F1-score.</p>	<p>Original source</p> <p>For the evaluation of the trained models, we used different evaluation measures including accuracy, precision, recall and f1-score</p>	<p>66 %</p>
<p>⑦ Student paper</p> <p>Submitted paper</p> <p>2018) Evaluation Metrics:</p>	<p>Original source</p> <p>The evaluation metrics chosen</p>	<p>65 %</p>
<p>④ Student paper</p> <p>Submitted paper</p> <p>Metrics like accuracy, precision, recall, and F1-score are frequently employed in sentiment analysis.</p>	<p>Original source</p> <p>accuracy, precision, recall, and F1-score</p>	<p>64 %</p>