Finding Entities based on Financial News

Capstone Project Summary for CEU Business Analytics MSc

2022 June

Andras Bognar

Contents

1. Project Description

The client of the project was a large financial investment firm whose financial team had access to sizeable financial text data (articles, statements etc.) that they wanted to utilize to identify trending investment topics and specific entities interested in said topics.

An existing solution was developed by the financial team to filter article headlines based on a predefined dictionary of relevant search words. However, this solution was limited to only finding exact matches in the headlines of articles and the dictionary had to be manually updated for each new topic of interest.

Therefore, the data science team was contacted to develop a more flexible solution which can find articles and the entities discussed in them based on user-provided search terms without the use of predefined dictionaries.

2. Project Summary

The final solution had two main parts: the background process training and loading the model matching search words to articles and the front-end application where users can easily access and use the tool in real time.

2.1 Tools used

Most of the project was completed using Python notebooks with the exception of the data loading which was done with SQL queries to a cloud database.

An internal repository with version control was used to store the notebooks used in the project.

2.2 Semantic search model

The basis of the solution was a natural language processing model designed to predict position of words in context based on previous documents. The specific model used in the project was Word2Vec that belongs to the family of semantic search models.

To train the model historical articles were used from the same internal source used in the previous existing solution.

The publisher of the articles also provides meta-data for each document including the ID of the entities discussed in them which is the main interest of the project. Articles were filtered based on the meta-data for English language, geographical location of the entities and publishing date.
The article texts were cleared of irrelevant information (publishing date, authors contact information etc.), then tokenized which is the process of identifying the unique words in the articles.

The processed text was used to train the Word2Vec model of the public library Gensim which assigns a mathematical value to each word in the documents. Based on word numerical values, articles themselves can be assigned a value by simply taking the average of the words included in them.
Based on this process we get two set of numbers one for the user search words and one for each article which can be compared to measure similarity of their meaning.

2.3 Streamlit application

The Streamlit Python library allows easy implementation of Python based machine learning applications. By using basic functions of the library users can input search words whose numeric values are calculated in the live application and compared to the already processed articles.

Based on this calculation the most similar articles are found for the search word and are grouped by which entity they were written about. The entities with the most relevant articles are shown to the user.

Other features of the application include showing the historical trend of number of relevant articles over time and showing the text of the most relevant articles to read about the topic.

A difficult challenge in designing the application is ensuring high application performance so users don't have to wait for long to get the results of the underlying calculations. This was achieved by limiting the results to 3 months, storing the results of calculation, so they don't have to be repeated and using efficient file formats.

3. Limitations, Further Recommendations

The main limitation of the process is the same as its main strength: it relies on historical articles. While the historical articles use very similar language to newly published ones, they don't contain any new potential topics that may come up in the future.

If the user searches for a word that rarely or never came up in the past, chances are the results will not be relevant as the context might have significantly changed e.g.: if a new technology like blockchain emerges it will not be included in the historical text and even if the technology is not entirely new but it's usage heavily shifted over a short time the historical context will not be relevant to the new use-case.

Accordingly, I recommended that to further enhance the tool the model training process should be made continuous. If a way can be found to further train the model with new articles, the model output can keep up with changing environment. In the current version of the application number of results is clearly shown to the user, thus they can be aware of the potential quality of recommendations. In addition, they can check individual articles to evaluate the output.

Another potential issue is that the model predicts words solely based on their position in the text and doesn't explicitly consider the sentiment behind the documents. That means that it can only distinguish two texts about the same topic with opposite sentiments if they used different words or the order of words changes. However, in practice these documents often only vary in a few words e.g.: one document may explain how a company follows sustainable standards while another might say the opposite, but they may use very similar words to describe sustainability. In that case the model will consider the two texts very similar in terms of meaning.

To account for this effect sentiment-based scores may be assigned to the articles to further filter only positive or negative articles on a given topic.

While during my investigation I did not find many examples of this effect, I also suggested working on this improvement as the similarities in positive-negative texts varies from topic to topic.

4. Summary

In conclusion, the project reached its stated goals, the tool developed is able to provide quick suggestions for potential entities based on user search words and the results can be validated by the user. As described in the above section further improvements are required for a final solution and the data science team is looking into continuing the project based on the user's feedback.

For me personally this was a great opportunity to learn about semantic models and Streamlit applications and I got a chance to work with knowledgeable experts in the field.