Capstone Project Summary

Project Title: Transactional Text Matching Using Semantic Models

Xibei Chen
June 12, 2022

Project Background, Objectives, and Scope

My client is a B2B company providing IT services and consulting to external clients. One of their services is Procurement Analytics as a Service (PAaaS). This service offers data-driven insights regarding savings opportunities to external clients by focusing on the volume aggregation of certain products and services. At the end of the pipeline, they create a dashboard for their clients that they can analyze with the help of Business Intelligence officers. Their clients can then use the insights to negotiate better terms for their future procurement.

During the process of PAaaS, an essential part is when my client categorizes the spending of their external clients into the UNSPSC taxonomy. UNSPSC stands for the United Nations Standard Products and Services Code, a global classification system of products and services. The detailed processing of three types of client data: vendor, master, and transactional, achieved the final categorization of UNSPSC taxonomy. This project focuses on transactional data, as this type is usually the most comprehensive, detailed textual information available regarding a particular purchase. A step in this categorization process is transactional text matching, where a machine learning model predicts the probability of a given new transactional text coming from client data and a similar UNSPSC taxonomy level description. Currently, the transactional text matching model already has five features that focus on the syntactic aspects of texts (e.g., distance metrics). Therefore, my client is already familiar with some of the concepts and methodologies in NLP. The motivation for this project is that my client would like to potentially extend this set of features by exploring the effects of other NLP techniques, especially those that focus on the semantic aspects of texts, to achieve a better text matching model.

Therefore, my project aims to make this model predict similarity as accurately as possible. Natural Language Processing (NLP) is one of the fastest-growing sectors in the field of artificial intelligence (AI) and machine learning (ML). Research has progressed in the sector of textual similarity. My project aims to take advantage of new NLP techniques to improve our existing

model potentially. Meanwhile, computational performance is a crucial aspect that I will have to consider during development because most external clients have several thousands of new transaction texts each month. The scope of my project is to improve the current transactional text matching process by adding more features to the current model and trying different algorithms. The expected output is a machine learning model for the problem mentioned above. It is not in the project's scope to deploy the model into production.

Technical Approach and Methodology

I researched multiple NLP techniques for the feature engineering process, including BERT Semantic Text Similarity, Hugging Face Sentence Similarity, Fuzzy String Matching, Stemming, N-gram Similarity, Cosine Similarity, Soft Cosine Similarity, and some other distance metrics.

Moreover, I created a customized error function to help me evaluate models in a project-specific way as an add-on metric to traditional classification metrics such as accuracy and AUC.

I also compared various ML algorithms, including Logistic Regression, Random Forest, Gradient Boosting, K-Nearest Neighbors, Multi-layer Perceptron, and XGBoost with hyperparameter tuning. In addition, I explored the Stack Ensemble technique and H2O AutoML.

Furthermore, I also conducted analysis in model interpretation and feature selection, which helped me better understand the correlation between features, their mutual information with the target variable, and their feature importance. I tried testing model performance, excluding a few least important features compared to including all the new features.

Last but not least, my client and I worked on putting together an evaluation notebook to test the old and new models in terms of computing time and prediction accuracy with actual external client data. We also wanted to detect some unique prediction patterns of both models during manual validation.

Conclusions, Limitations, and Future Research

The model of my choice after the model selection process was Robust Scaler and Random Forest Classifier with specific hyperparameters, which increased accuracy and F1 score by 4%, and AUC by 4.5%, compared to the old existing model in the legacy process. During model interpretation and feature selection, I found that excluding some least important features resulted in worse prediction performance to some extent. Therefore, I kept all features eventually to move on to the final testing on the holdout sample.

We found that only BERT Semantic Similarity and Hugging Face Sentence Similarity took a significant amount of time to compute, while the computing time for other features was negligible. In addition, the computing time for BERT Semantic Similarity tripled the computing time for Hugging Face Sentence Similarity. However, according to my feature selection analysis, they shared similar feature importance. Therefore, I would not recommend my client to use BERT Semantic Similarity as the trade-off between prediction performance enhancement and computing time is not as promising as Hugging Face Sentence Similarity.

After conducting manual validation of the prediction of both models, I found that the new model had 6% more not null predictions than the old model. The new model also had 2% more accuracy than the old model. I also analyzed the class probability distributions of both models. The distribution of the new model made more sense than the old one, as the distribution of the old model had a rather strange rugged look. Moreover, I analyzed the accuracy of different probability intervals of both models. The new model outperformed the old one in general. I also found out that the accuracy of the probability interval of (0.7, 0.8] was 4% higher than for the interval of (0.8, 0.9] in the old model, which was a strange pattern.

My client and I also found that the new model had a better prediction for specific categories of products, but there were also unique patterns where it did not perform as well as the old model. One limitation of my project was the size of the training dataset, which contains 3235 observations, as my client would need to invest many resources into gathering more representative training data. The unique patterns mentioned above where the new model performed worse than the old model would be helpful for choosing more training data with specific patterns for future model training.

However, we only tested the model on three client rounds and analyzed 306 rows of sampled data. Therefore, our findings in this final evaluation step could be client-specific and not representative of all the external clients. If my client wants the findings to be more general, testing on more client rounds with more sampled data needs to be conducted in future work.

In conclusion, as proof of concept, I found that Hugging Face Sentence Similarity stood out regarding feature importance and computing time and that Random Forest is an ideal classifier with great prediction and computing performance. The new model outperformed the old model on both the test set in the training dataset and the holdout sample with actual external client data. Furthermore, if we scale with thousands of external client data each month using the new model, the amount of accurately predicted actual data would be huge, leading to a significant reduction in manual validation workload. Therefore, the improvement is not as trivial as the percentage seems. However, putting a new model into production would need a lot more future work to be done, such as testing on various external client data and taking care of package dependencies.