Phrase Matching Similarity for U.S. Patents

Michael Woo & Veena Chaudhari

Statistical Methods in Data Science

Real Data Analysis: Report

Background

The project performs phrase to phrase matching from different patent documents to determine the similarity between patents. A patent is a form of intellectual property that grants the patent holder the exclusive right to exclude others from making, using, importing, and selling the patented innovation for a limited period. The process of granting a patent is time-consuming and can take anywhere from 2 to 3 years. The most time-consuming step is the examination of a patent which requires the patent officer to search through the existing patents for claimed inventions. The number of patents granted per year has increased by two folds in the last two decades, with around 11 million in the repository (Figure 1). Therefore, Fig 1. estimates the amount of workload the patent offer is presented with to conduct a thorough search through prior art.

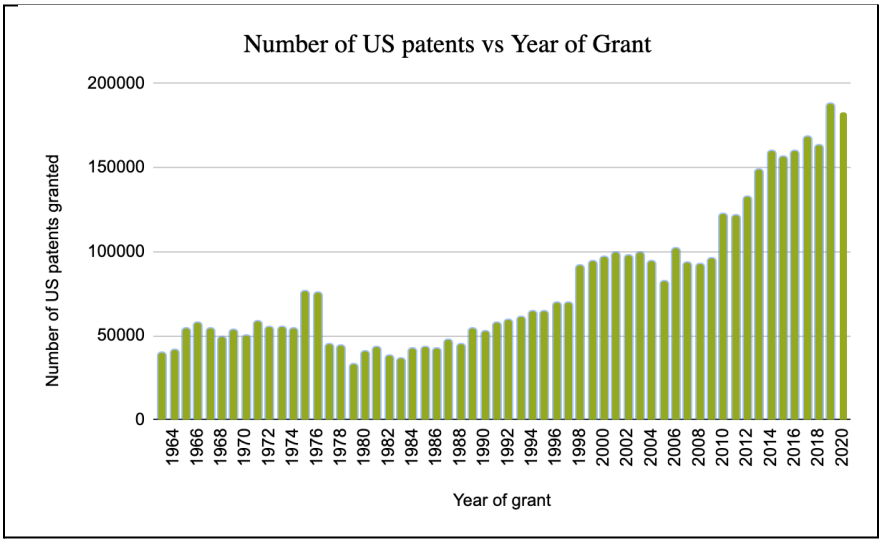

Figure 1: Graph showing the number of patents vs the year they were granted

As a result, the patent officers are backlogged and overworked, making the already time-consuming process even longer. Therefore this project aims to reduce the time for the PTO to search for existing patents by developing a model that extracts relevant information by matching key phrases in patent documents. The model determines how two phrases are similar and scores them based on their similarity.

Data Introduction

The U.S. Patent and Trademark Office (USPTO) portal has promoted a contest on the website Kaggle, where we have gotten our dataset. The USPTO offers one of the largest repositories of scientific, technical, and commercial information globally through its Open Data Portal.

The dataset consists of 4 attributes: id, anchor, target, and context. Id is the unique identifier for each pair of phrases, while anchor and target are the pairs of phrases that are the

predictors for the score response. Each couple of phases is labeled by the context, giving some background information about the domain to that phrases belong. The CPC (Cooperative Patent Classification) provides the context, which indicates the subject to which the patent relates. The CPC classification consists of a section and class. The section contains letters from A to H, and the class is a two-digit number. The sections represent different domains under which a patent is filed and granted. The sections are the following:

A: Human Necessities
B: Operations and Transport
C: Chemistry and Metallurgy
D: Textiles
E: Fixed Constructions
F: Mechanical Engineering
G: Physics
H: Electricity

The response variable is the similarity between the pairs of phrases on a scale of 0 to 1, with 1 being the closest match and 0 being unrelated. The table below (Table 1) explains the different scores and their meaning in scoring similarity.

Table 1: Score values and their meaning for scoring similarity between phrases

Score Value	**Meaning**
1.0	Very close match (Exact match except possibly for differences in conjugation, quantity)
0.75	Close synonym
0.5	Synonyms which don't have the same meaning
0.25	Somewhat related
0	Unrelated

The dataset consists of 36473 rows and 5 columns.

Methods

Exploratory data analysis

Exploratory data analysis is a crucial step in any machine learning project, which involves performing initial investigations on data to discover patterns, spot anomalies, test hypotheses, and check assumptions to help understand the dataset better and provide graphical representations.

Feature engineering

The second method used is feature engineering, which consists of tokenization, removing stop words, stemming, semantic similarity using BERT score, Spacy score, fuzzy wuzzy. The correlation matrix is calculated to determine highly correlated features.

Tokenization

Tokenization splits the raw data into small chunks of words or sentences called tokens. Then, the tokens can be used directly as a vector representing that document—the first step to any natural language processing pipeline. Converting sentences into vectors is crucial as it ensures that machine learning algorithms receive numerical values of words. The figure below shows an example of how the sentence "Introduction to statistical methods in Data science" is split into 7 tokens (words).

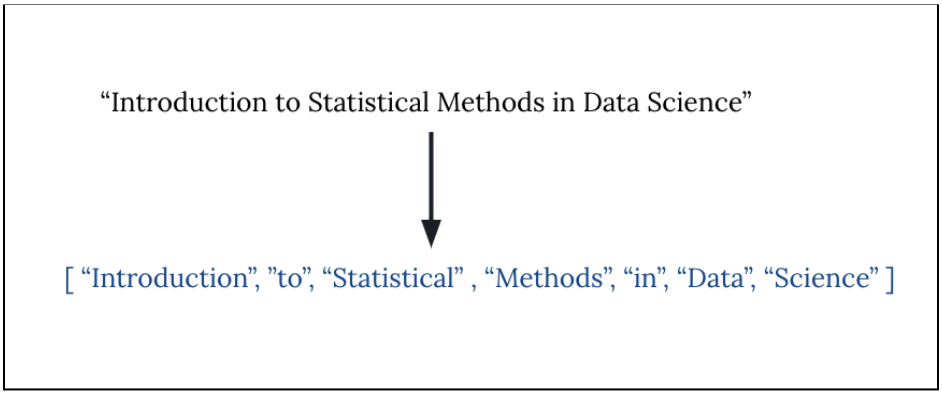

Figure 2: Example of Tokenization

Stopwords Removal

Common words are called stop-words (Example: its, an, the, for, and that) which carry low-level information from the phrases. Therefore, these stop words are removed from the tokens to focus on the crucial information.

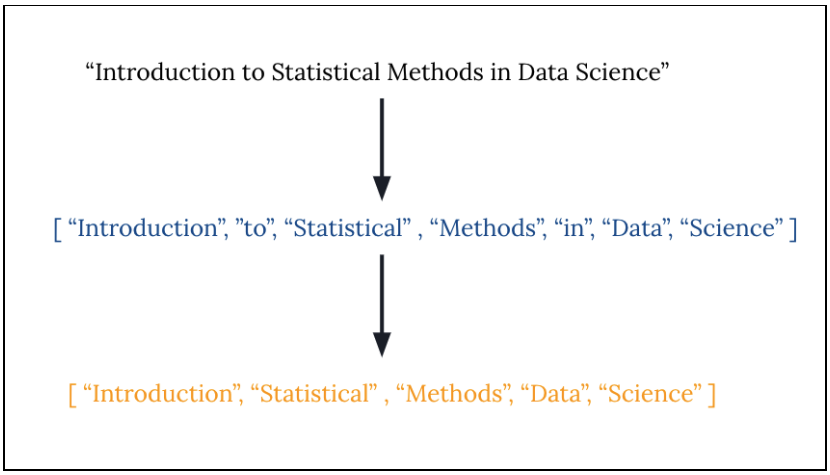

Figure 3: Example of removing stop words

Stemming

Stemming is another feature engineering process that involves reducing a word to its root word by removing suffixes or prefixes from its variants. This part in the pipeline ensures data normalization by eliminating repetition and transforming words, thereby reducing the redundancy in the data.

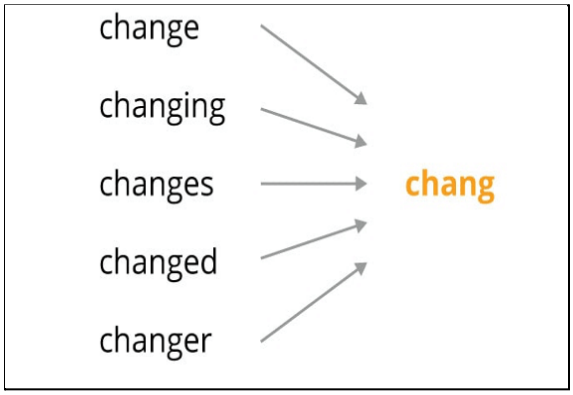

Figure 4: Example of Stemming

BERTScore

The next step in the pipeline is using BERT to generate a BERT score for the parts of phrases highlighting the similarity between the phrases. BERT is Bidirectional Encoder Representations from Transformers, which uses Recurrent Neural Network to score phrases depending on the distance. The BERTScore ranges from -1 to 1, with -1 being no similarity between them and 1 being highly similar. The BERT score provides the semantic similarity between anchor and target, which allows the model to perform better.

SpaCy Score

Another similarity scoring method used is the SpaCy score, which uses a Convolutional Neural Network to compare two vectors (words, text spans, and documents) and predict how similar using the Cosine Similarity. The SpaCy score ranges from 0 (no similarity) to 1 (high similarity).

Fuzzy Wuzzy Score

A different method is fuzzy wuzzy which uses a library to compare text. It calculates the differences between sequences using Levenshtein Distance (Levenshtein distance between two words is the minimum number of single-character edits (insertions, deletions, or substitutions) required to change one word into the other). The Levenshtein ratio is calculated by dividing the Levenshtein distance by the maximum length of strings 1 and 2.

Correlation Matrix Spearman

Finally, the correlation between features is calculated using Spearman correlation which measures the degree of monotonic association between two variables. The correlation values determine which highly correlated features and provide the model's same effect. Any two highly correlated features will have the same impact on the prediction. Therefore, when two highly correlated features are present, one of the two features can be removed to reduce redundancy and model complexity.

Feature Engineering: Combination of scores

The scores from BERT, Spacy and fuzzy wuzzy were calculated between two different attributes:

1) Anchor and target (given by 'at_score')
2) Anchor and context (given by 'ac_score')
3) Context and target (given by 'tc_score')

Along with these few more combinations of scores were calculated,

4) Sum of ac and tc scores (given by 'c_score')
5) Anchor and target score with respect to the total score(given by 'avg_at')
6) Sum of ac and tc scores with respect to the total score (given by 'avg_c')
7) Total score, sum of at,ac and tc (given by 'total')

<u>Machine Learning Models</u>

Two machine learning models were used to predict the similarity between the phrases from US patents.

1) *Random Forest Regressor:*

   The random forest regressor is an ensemble technique that performs regression tasks using multiple regression trees. It combines numerous decision trees in determining the final output rather than relying on individual decision trees. Each tree's output value (score) is averaged to give a final score.

2) *CatBoost Regressor:*

   The CatBoost regressor builds upon the theory of decision trees and gradient boosting. The main idea of boosting is to sequentially combine many weak models (a model

performing slightly better than random chance) and thus, through greedy search, create a solid competitive predictive model. Since the gradient boosting fits the trees sequentially, the next model will improve on the shortcoming of the previous model.

Model Evaluation

The two machine learning models are evaluated based on

1) *Mean Square Error (MSE):*
   The mean or average of the squared differences between predicted and expected target values in the dataset. A perfect mean squared error value is 0.0, which means that all predictions matched the expected values exactly.
2) *Root Mean Squared Error (RMSE):*
   The root mean square is the square root value of MSE. A perfect RMSE value is 0.0.
3) *Mean Absolute Error(MAE):*
   It is the average of the absolute error values. The MSE or RMSE punishes significant errors more than minor ones, but MAE does not give more or less weight to different types of errors; instead, the scores increase linearly with errors. A perfect mean absolute error value is 0.0
4) *R-Squared value:*
   It is a squared correlation between the observed values and the predicted values. The higher the R-squared, the better the model.

SHAP Model

The SHAP model is used to determine the feature importance based on each model. The SHAP model is built on the SHAP (SHapley Additive exPlanations) values that interpret the impact of having a particular value for a given feature compared to the prediction we'd make if that feature took some baseline value. Using these SHAP values for feature importance, we can determine how much each feature has influenced the model's prediction. In addition, a summary plot of SHAP values gives the general sense of features' directionality impact based on the distribution of the red and blue dots. These models and summary plots help to ensure that the model is intuitive and makes the right decisions.

Results

Exploratory Data Analysis

The id column was removed since it is a unique identifier and did not contribute to the prediction.

Anchor column has 733 unique values with phrases consisting of less than six words. Most of the phrases have two words. The word cloud figure below visualizes the anchor column wherein each word is picturized with its importance or frequency. The larger words are representative of higher frequency, whereas the smaller-sized words have a lower frequency.

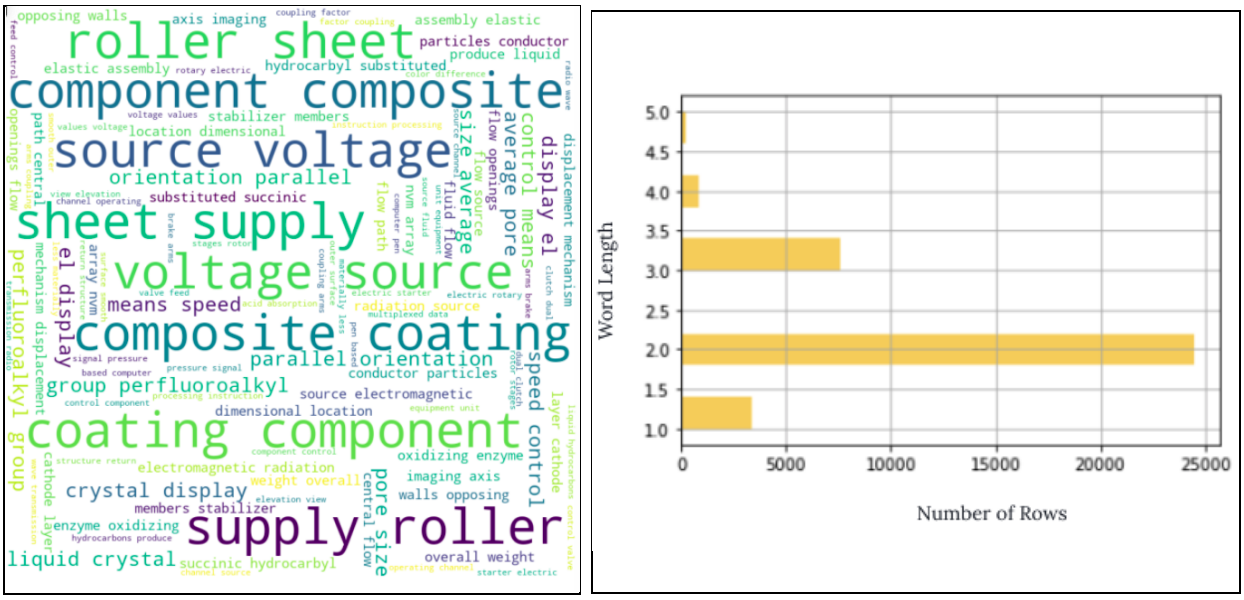

Figure 5: Exploratory data analysis for Anchor columns (a) (*Left side image)* Word Cloud (b) (*Right side image*) Number of rows having phrases with of less than 6 words

Target column has 29340 unique values with phrases consisting of less than 16 words. Most of the phrases had two words. The word cloud figure below visualizes the anchor column wherein each word is picturized with its importance or frequency. The more significant words represent a higher frequency, whereas the smaller-sized words have a lower frequency.

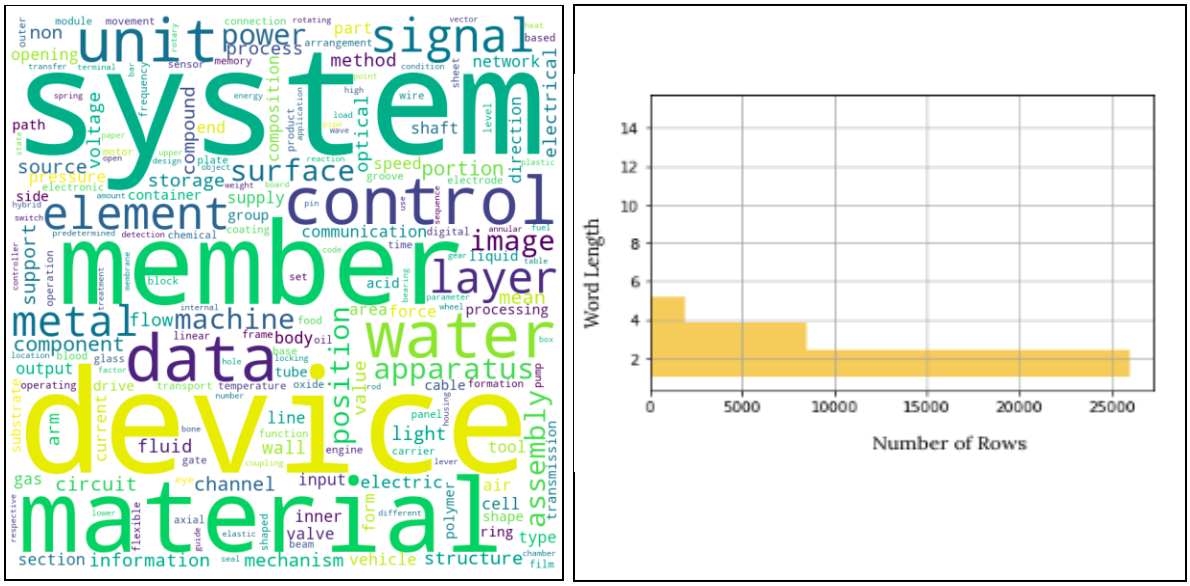

Figure 6: Exploratory data analysis for Target columns (a) (*Left side image)*Word Cloud, (b) (*Right side image*) Number of rows having phrases with of less than 16 words

The context column gives the sections to which the phrases belong. The sections, classes, and meanings are extracted from an additional dataset of CPC code titles. This column's analysis shows that the most common sections are B: operations and transport, followed by H: Electricity, and then C: Chemistry and Metallurgy.

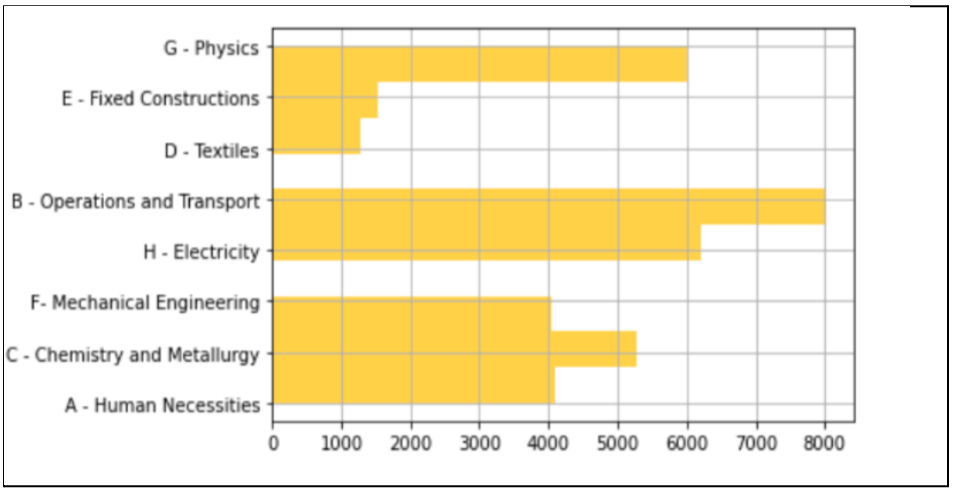

Figure 7: Frequency of sections from CPC classification from context column

Feature Engineering: Correlation Matrix Spearman

Feature engineering resulted in 24 attributes which were analyzed for correlations. 19 features out of the 24 predictors were used for prediction based on the correlation values. The red boxes in the figure show that the features are positively correlated, whereas the blue values indicate that the features are negatively correlated. Features that were positively correlated with the score were selected as the predictors.

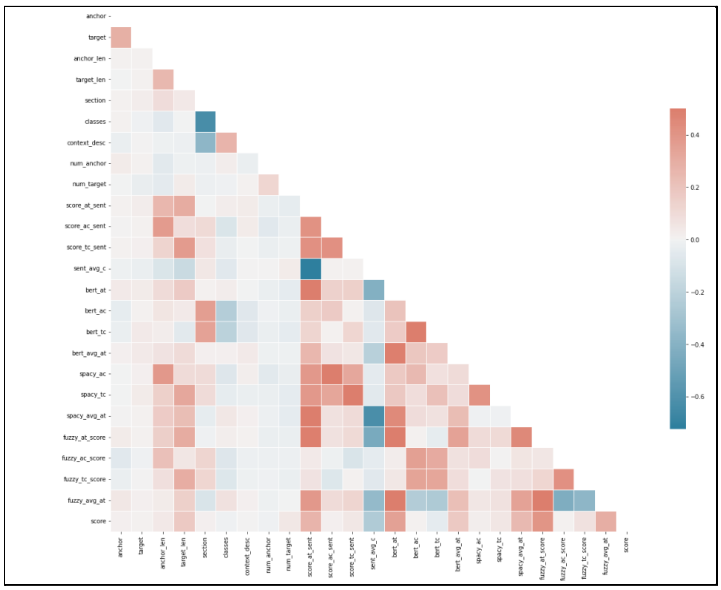

Figure 8: Correlation matrix using spearman correlation

The 19 features used for predicting are the following:

anchor num_target score_at_sent bert_at bert_avg_at spacy_avg_at

target target_len score_ac_sent bert_ac spacy_ac fuzzy_at_score

anchor_len section score_tc_sent bert_avg_at spacy_tc fuzzy_ac_score

fuzzy_tc_score fuzzy_avg_at

Model Evaluation

The metrics for both models is summarized in the table below.

Table 2: Metrics and values for both the models

Metrics	**Random Forest Regressor**	**CatBoost Regressor**
Mean Square Error	0.044	0.041
Root Mean Squared Error	0.209	0.203
Mean Absolute Error	0.160	0.158
R-Squared	**0.344**	**0.382**

Graphs:

One of the many decision trees that are evaluated in Random Forest is shown below.

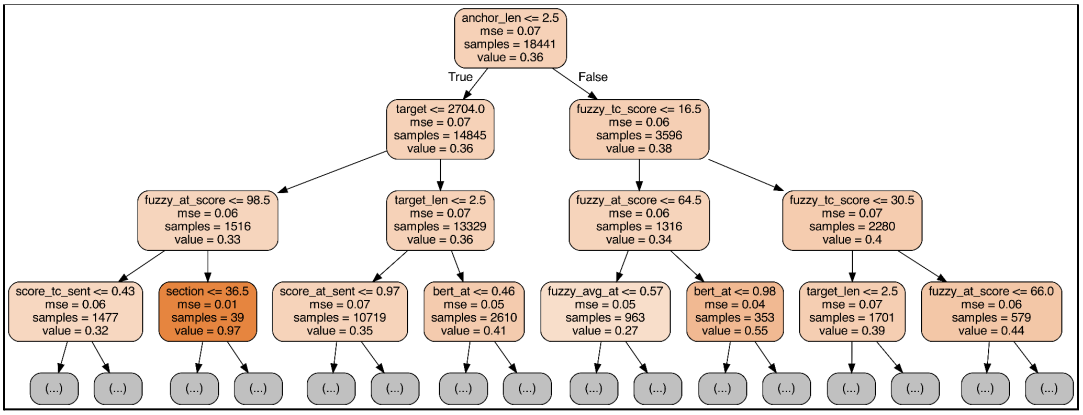

Figure 9: Random Forest graph showing one of the decision trees (Note that for the sake of simplicity only 3 levels are shown for both trees)

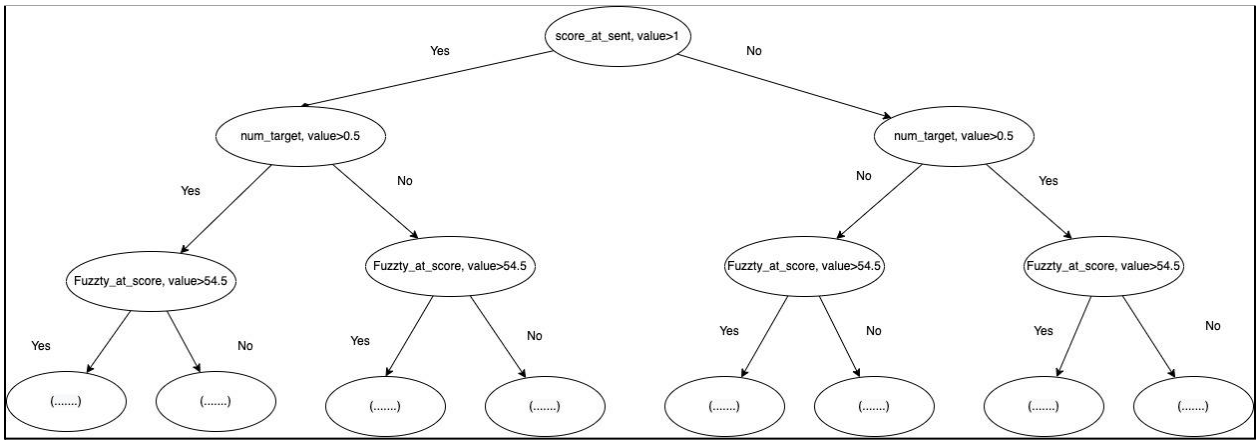

Figure 10: CatBoost graph showing one of the decision trees (Note that for the sake of simplicity only 3 levels are shown for both trees)

SHAP models

a) Random Forest Regressor

The SHAP model shows that the most important predictors for random forest are fuzzy scores between anchor and target, followed by BERT similarity score between anchor and target and spacy score between anchor and target. From the plot, it is also evident that similarity scoring between anchor and target columns is the most important when it comes to prediction.

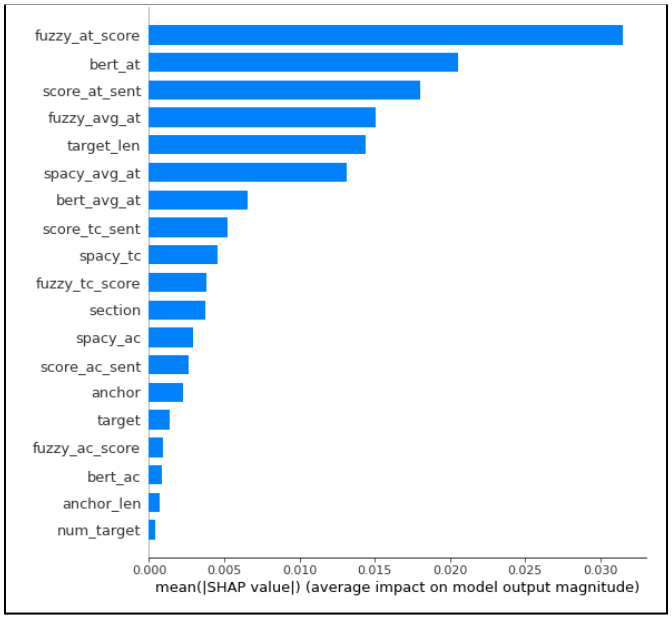

Figure 11: Bar plot for summary plot of SHAP values for Random Forest Regressor

In the dot plot, Red color means a higher value of a feature, and Blue means a lower value. The dot plot suggests that the fuzzy score followed by the BERT similarity score and spacy score, all between anchor and target, are the essential features for the model's points of view.

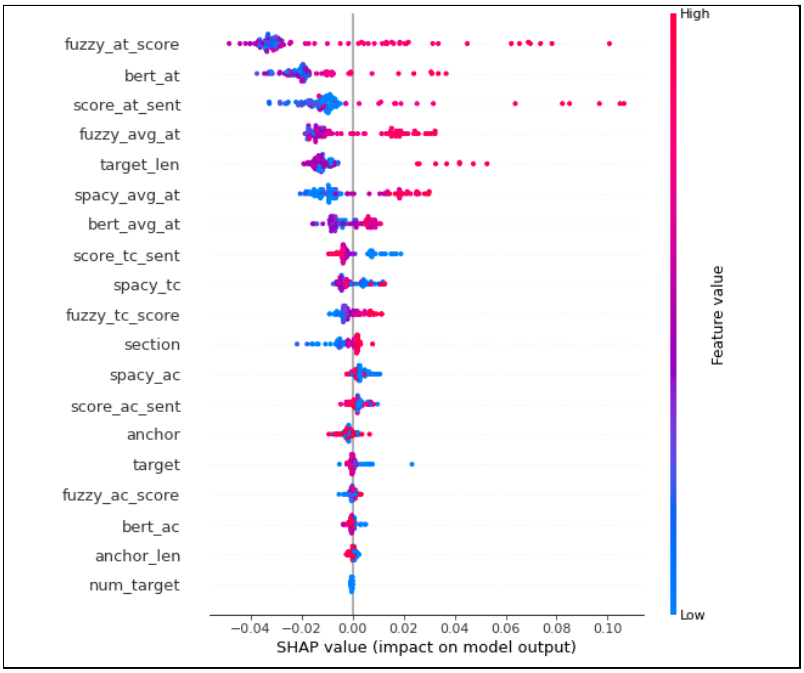

Figure 12: Dot plot for summary plot of shap values for Random Forest Regressor

b) CatBoost Regressor

The SHAP model results for CatBoost regressor are similar to the Random forest regression, concluding that fuzzy score followed by BERT similarity score and spacy score, all between anchor and target, is the most critical features from the models' points of view.

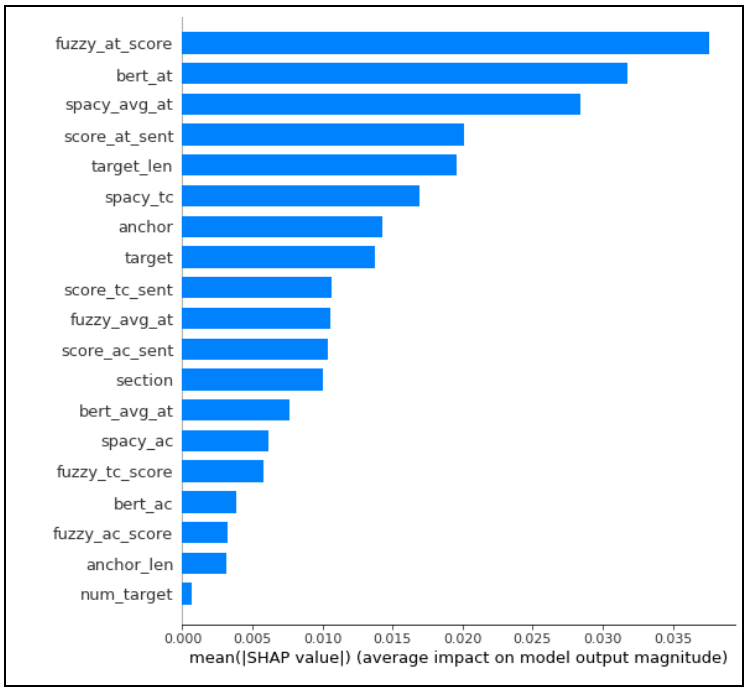

Figure 13: Bar plot for summary plot of SHAP values for Catboost regressor

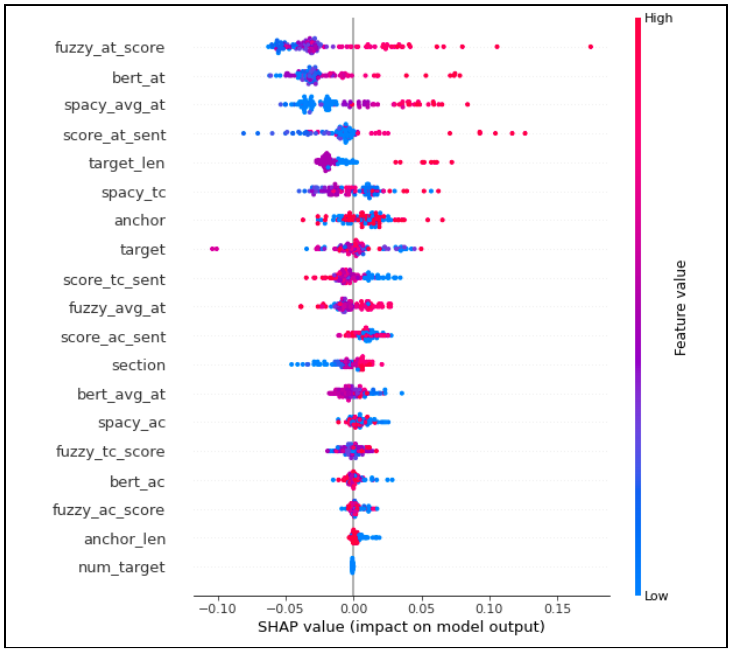

Figure 14: Dot plot for summary plot of SHAP values for Catboost regressor

Discussion

The models do give consistent results. Based on the metrics of the models, these metrics suggested that the CatBoost Regression performed better than the Random Forest Regression slightly. The slight performance could be due to the main characteristic of the CatBoost, which is its gradient boosting feature. Further improvements to the models, such as different feature engineering methods or different libraries for scoring phase similarities. Another approach is to increase the sample size of the phases. Instead of being given phrases to a patent document, the entire patent document containing all the words would be more suitable for our data processing method.