Capstone Report - Crowdflower Search Results Relevance

Problem Statement:

Create an open-source model that can be used to measure the relevance of search results. Model performance is evaluated on quadratic weighted kappa, which measures the agreement between two ratings - scores assigned by the human rater and the predicted scores.

Benefits

Small Business Owners

Allow to have relevant information to compete with more resource-rich competitors on user experience on search. Save them cost on gathering the needed data to train and test different model algorithms

Dataset Description

Obtained from Kaggle, data are in the form of 2 dataset

- train:
    - id: Product id
    - query: Search term used
    - product_description: The full product description along with HTML formatting tags
    - median_relevance: Median relevance score by 3 raters. This value is an integer between 1 and 4.
        - 4 indicating the item completely satisfies the search query, and 1 indicating the item doesn't match the search term
    - relevance_variance: Variance of the relevance scores given by raters.
    - Size (10158, 6)
- test: same as train set, without ‘median_relevance’ and ‘relevance_variance’. Size (22513, 4)

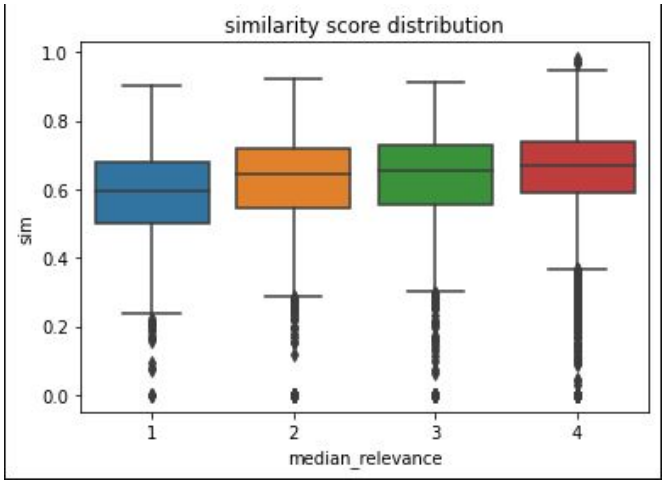
Cleaning and Wrangling

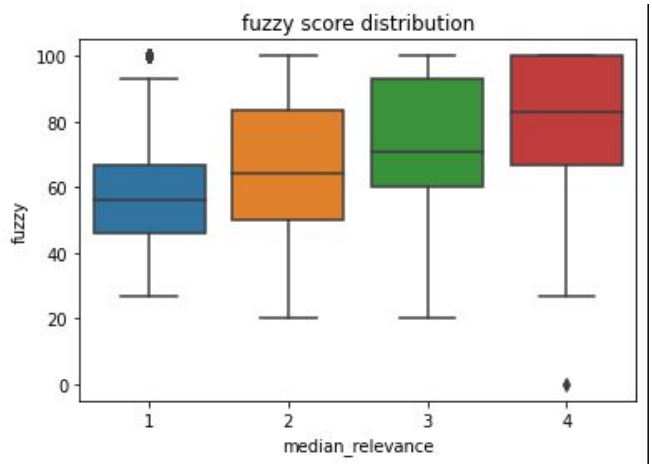
- Missing data - identified 2444 missing records under ‘product_descrpition’. Assign ‘none’ for those
- Text Preprocessing Steps on product title and product_description
    - Remove stop words, numbers, and punctuation
    - Lemmatize, then tokenize
    - Join the words tokens from both product title and product_description → text_fin feature created
- Obtain similarity scores from NLP packages
- Obtain fuzzywuzzy scores
- Create a new feature that counts the length of the query
- Generate word features using Tfdif to obtain keywords by median relevance rating

Exploratory Analysis

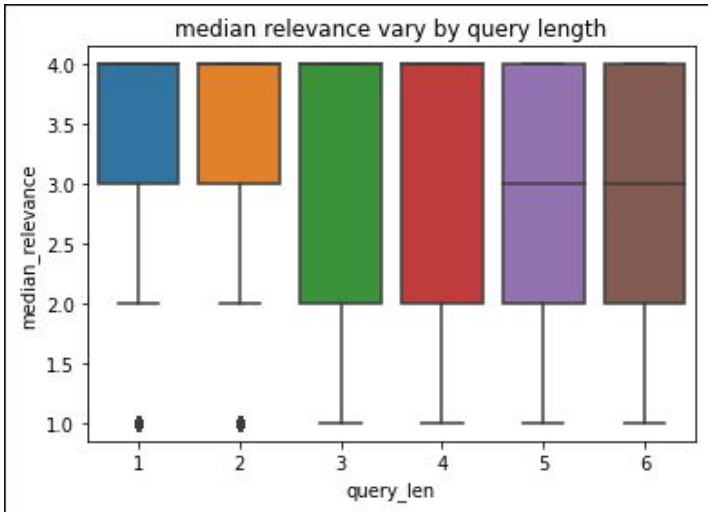
The goal is to discover features that may influence the median_relevance.

Findings:





- **Observation** : positive correlation between fuzzywuzzy/similarity scores with median_relevance.



- **Observations:** Longer search queries have lower median relevance score on average

Machine Learning

Applied dataset into Logistics Regression, Support Vector Machine, Random Forecast, and Extreme Gradient Boosting. Used Bayesian parameter optimization for models' tunning.

Model Performance (quadratic kappa on Kaggle)

Model	quadratic kappa	Time
dummy	0.0	1s
Logistic Regression	0.497	4s
Support Vector Machine	0.483	4m 20s
Extra Trees	0.554	3m 40s
Random Forest	0.539	3m 14s
XGBoost	0.532	9m 3s

Recommendation/Next Steps

Tree-based ensemble models such as Extra Trees and Random Forecast performed the best in this analysis. From the absolute quadratic kappa score standpoint, Extra Trees model performs the best and will be the winning model.

Note that the dataset target attribute is imbalanced, with 60% of rating being 4. We may form an assumption that users tend to give positive when they find the query result useful, but may not even give any rating when the search results are bad, which explained why only 8% of ratings were 1. Clients can leverage the result in two ways:

- Study the query results of 4, identify key drivers on why those results had good ratings, and apply that search algorithm into more product categories
- Study the query results of 1, plus queries that didn't have a rating, to draw key drivers on why the search results were not useful, and improve search algorithms on those

One more powerful information to obtain is whether the product recommended results in purchases. This information would help narrow down the most effective query/product combination, which will help clients to improve conversion on their site.