Technical Review

[**Introduction** 1](#_Toc24880541)

[**Project Goal** 1](#_Toc24880542)

[**Data Introduction:** 1](#_Toc24880543)

[**Data Set:** 1](#_Toc24880544)

[**Data Fields Overview** 1](#_Toc24880545)

[**Data Overview** 1](#_Toc24880546)

[**Software Libraries** 2](#_Toc24880547)

[**Method** 3](#_Toc24880548)

[**Reference:** 3](#_Toc24880549)

**Introduction**

Quora is a place to gain and share knowledge—about anything. It’s a platform to ask questions and connect with people who contribute unique insights and quality answers. This empowers people to learn from each other and to better understand the world.

Over 100 million people visit Quora every month, so it's no surprise that many people ask similarly worded questions. Multiple questions with the same intent can cause seekers to spend more time finding the best answer to their question, and make writers feel they need to answer multiple versions of the same question. Quora values canonical questions because they provide a better experience to active seekers and writers, and offer more value to both of these groups in the long term.

**Project Goal**

The goal is to use natural language and machine learning model to predict which of the provided pairs of questions contain two questions with the same meaning. We use this project to explore language models, compare machine learning models, and understand modeling trade-offs

**Data Introduction:**

The goal of this NLP project in Python is to predict which of the provided pairs of questions contain two questions with the same meaning. The ground truth is the set of labels that have been supplied by human experts. The ground truth labels are inherently subjective, as the true meaning of sentences can never be known with certainty. Human labeling is also a 'noisy' process, and reasonable people will disagree. As a result, the ground truth labels on this dataset should be taken to be 'informed' but not 100% accurate, and may include incorrect labeling. We believe the labels, on the whole, to represent a reasonable consensus, but this may often not be true on a case by case basis for individual items in the dataset.

**Data Set:**

The training set contains 404290 pair of labeled question pairs as "duplication" or not. The test set contains 2345796 unlabeled question pairs for prediction

**Data Fields Overview**

▸ **id** - the id of a training set question pair

▸ **qid1, qid2** - unique ids of each question (only available in train.csv)

▸ **question1, question2** - the full text of each question

▸ **is\_duplicate** - the target variable, set to 1 if question1 and question2 have essentially the same meaning, and 0 otherwise.

**Data Overview**

|  |  |  |
| --- | --- | --- |
|  | Question1 | Question 2 |
| 0 | What are some good movies to watch? | What are the best movies to watch? |
| 1 | Do dentist earn more than other doctors? | Do dentist earn more than other doctors? Why? |
| 2 | Should I wait for iPad Air 3 or Purchase the iPad Air 2? | Should I buy the iPad Air or wait for the next iPad Air(iPad Air 2) |
| 3 | What is the difference between Java and Android programming | Are there major difference between programming in Android vs plain Java? |
| 4 | Why do you yawn when you are tried? | Why do we yarn when we are sleepy? |

▸ Duplicates proportion**: 36.9% in train, 17.4% in test**

▸ Number of question pairs: **~400k in train, ~2,3M in test**

▸ **~80%** of test dataset contains fake question pairs, such that we can’t hand label test question pairs (avoid cheating)

▸ **~530k unique questions** in train dataset

▸ **~110k** questions appear multiple times in train and test datasets

**Software Libraries**

The project is implemented with the following software libraries.

1. **Pandas** is a software library written for the Python programming language it provide high performance for data manipulation and analysis
2. **Numpy** is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.
3. **Jellyfish** is a library of functions for approximate and phonetic matching of strings
4. **Gensim** is a software library for unsupervised topic modeling and natural language processing, using modern statistical machine learning.
5. **NLTK (Natural Language Toolkit)** is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries, and many more**.**
6. **Scikit-learn** (**sklearn** ) It’s a software Library, It features various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, k-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.
7. **TextBlob** is a library for processing textual data. It provides a simple API for diving into common natural language processing (NLP) tasks such as part-of-speech tagging, noun phrase extraction, sentiment analysis, classification, translation, and more**.**
8. **Matplotlib** is a Python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms. You can generate plots, histograms, power spectra, bar charts, errorcharts, scatterplots, etc

**Method**

The project is implemented with the following machine learning modeling steps.

* **Step 1.** Exploratory Data Analysis
* **Step 2.** Language Model Representation Analysis and Selection
  + Use gensim's word2vec text representation with context consideration, combine question pairs and calculate the mean vector representation as a feature
  + Use gensim's paragraph vector (Doc2vec) text representation with context consideration, combine question pairs and use paragraph vector as a feature
* **Step 3.** Feature Engineering with addition pair features
* ***Used:***
  + Average word set similarity using ochiai, jaccard, overlap similarity measures
  + Cosine distance between sentence vector representations
  + Shared words between 2 questions

***Considered not used:***

* + Sentiment score
  + Noun verb counts
  + Number of unique words
* **Step 4.** Classification Model Evaluation and Selection
* ***Methods Used:***
  + Logistic Regression
  + Naive Bayes
  + Support Vector Machine
  + Random Forrest
* **Step 5.** Hyperparameter Tuning
* **Step 6.** Test Set Evaluation and Reporting

**Reference:**

1. <https://en.wikipedia.org/>
2. <https://matplotlib.org/>
3. <https://pandas.pydata.org/>
4. <https://www.nltk.org/>
5. <https://textblob.readthedocs.io/en/dev/>
6. <https://scikit-learn.org/stable/>
7. <https://pypi.org/>
8. <https://numpy.org/>