DUPLICATE QUESTION IDENTIFICATION IN ONLINE DISCUSSION PLATFORMS

Aswin Kumar G R

National College of Ireland, Dublin, Ireland

x23245778@student.ncirl.ie

Abstract-In online community discussion platforms and other question-answering systems, the potential to recognize duplicate questions is crucial for enhancing user satisfaction and the accuracy of information retrieval. For this project, we developed a machine learning model to determine if pairs of questions are duplicates or not. The training and assessment dataset consisted of question pairs that were categorized as either duplicates or non-duplicates from a community forum. To extract relevant features from the text data, we utilized a combination of advanced natural language processing algorithms and traditional feature engineering methods. These features included token-based characteristics, fuzzy matching scores, as well as lexical and syntactic similarity. The collected features were utilized to train two widely employed machine learning algorithms, Random Forest and XGBoost, with the aim of classifying question pairs. Standard performance metrics such as confusion matrices and accuracy were employed to evaluate the models. Furthermore, the high-dimensional feature space was visualized using dimensionality reduction techniques like t-SNE. The trained models achieved an impressive accuracy of approximately 79% on a separate test set, showcasing the effectiveness of the proposed method in identifying duplicate questions. This study emphasizes the pivotal role of feature engineering and model selection in developing reliable systems for detecting duplicate questions on Q&A platforms and community forums.

I. INTRODUCTION

Identifying similar questions is a crucial task in natural language processing (NLP) and information retrieval, particularly in online community forums and question-answering platforms. Efficiently providing relevant information, the detection of duplicate queries enhances search effectiveness, reduces redundancy, and overall improves the user experience. In this project, we employ machine learning techniques to tackle the issue of identifying duplicate questions. The primary objective is to create a model capable of automatically determining whether two question pairs are duplicates. Our aim is to develop a dependable and effective duplicate question detection system by leveraging machine learning and feature engineering methodologies in combination with the textual content of the questions. The project's dataset comprises pairs of questions that were classified as either duplicates or non-duplicates based on their origin from a community forum. To extract relevant features from the text input, a combination of advanced natural language processing techniques and traditional feature engineering methods is employed. These attributes capture fuzzy matching scores, lexical and syntactic similarities, and other linguistic characteristics that can indicate redundant questions. We examine two popular machine learning methods for classification problems: XGBoost and Random Forest. Once these models have been trained on the extracted features, standard performance metrics like accuracy are used to assess them. Furthermore, dimensionality reduction techniques such as t-SNE are employed to obtain feature space visualizations and insights into the distribution of data points. The project's findings have relevance for numerous applications, such as information retrieval systems, question-answering platforms, and online discussion boards. An efficient technique for detecting duplicate questions can enhance content moderation, improve the pertinence of search results, and make it easier for users to access pertinent information.

II. LITERATURE REVIEW

Identifying duplicate questions is a valuable function in numerous online platforms and information retrieval systems, and it has garnered significant attention in the field of natural language processing (NLP). Numerous studies have proposed innovative techniques and approaches to address this challenge and enhance search precision and user satisfaction. The research conducted by [1] is considered a seminal work in the field of duplicate question identification. They proposed a framework that utilized textual similarity features and machine learning approaches to detect similar questions on question-answering platforms and online community forums. Their methodology demonstrated promising outcomes in identifying questions addressing comparable topics. [2] explored the use of cosine similarity and TF-IDF weighting as information retrieval techniques to identify duplicate questions. The researchers found that these strategies were particularly effective in improving the accuracy of detecting duplicate questions when working with large-scale data sets. In recent years, deep learning models have proven to be highly beneficial for tasks involving natural language understanding, such as question duplication detection. [3] introduced a Siamese convolutional neural network (CNN) architecture with the aim of identifying duplicates and capturing the semantic representations of queries. Their model outperformed traditional techniques, particularly in its ability to capture nuanced semantic similarities between questions. Advancements in word embedding methods have enabled the creation of more sophisticated algorithms for detecting duplicate questions. The researchers [4] proposed a Siamese recurrent neural network (RNN) with an attention mechanism that utilizes pre-trained word embeddings to effectively detect semantic connections between queries. Their method achieved state-of-the-art performance on common benchmark datasets, demonstrating the effectiveness of deep learning approaches in this task. In addition to neural network-based techniques, ensemble learning methods, such as combining decision trees and support vector machines (SVM), have also been explored for the purpose of distinguishing duplicate questions. [5] developed a hybrid model aiming to enhance the robustness and generalization of systems designed to identify duplicate questions. Furthermore, The effectiveness of algorithms used to identify duplicate questions can be improved through feature engineering. The study supervise by [6] look into the ability of various lexical, syntactic, and semantic features extracted from question pairs to distinguish between duplicate and non-duplicate questions. Their findings emphasized the importance of feature representation and selection in achieving high accuracy in the identification of duplicate questions. The literature review highlights the diverse range of approaches employed in the identification of duplicate questions, spanning from traditional machine learning algorithms to advanced deep learning architectures. Researchers have made significant advancements in enhancing the effectiveness and efficiency of duplicate question detection systems by leveraging these techniques and implementing innovative feature engineering methodologies.

III. METHODOLOGY

In online question and answer platforms, the identification of duplicate questions is essential for natural language processing, as it enhances search relevance, eliminates redundant content, and improves user experience. This study aims to develop an efficient duplicate question detection system by employing assessment approaches, machine learning models, text representation, and feature engineering. The methodology encompasses several critical processes, including data preprocessing, feature extraction, model training, evaluation, and query point prediction.

A. Data Preprocessing

Data preprocessing is the initial step in our approach, which cleans and prepares the question pairs for analysis. This includes addressing capitalization and stemming to reduce words to their fundamental form, as well as removing any unnecessary letters, punctuation, and special characters. Additionally, we tokenize the queries into individual words to assist with future analysis.

B. Feature Engineering

Crafting an effective duplicate question detection system heavily relies on feature engineering. To capture the diverse aspects of similarity between question pairs, we extract a variety of attributes from them. These features encompass length-based data, such as the absolute difference in length and average token length, as well as token-based features like the count of common words, common stopwords, and common tokens. Additionally, to account for approximate string matching, we collect fuzzy features including the fuzzy ratio, fuzzy partial ratio, token sort ratio, and token set ratio.

C. *Text Representation*

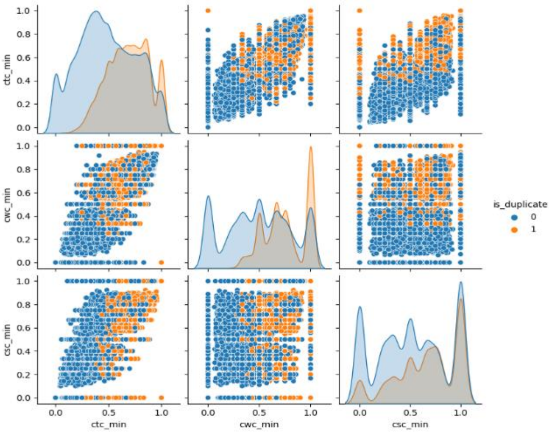
The CountVectorizer converts the textual input into numerical features by generating a bag-of-words representation for each pair of questions. This process transforms the text input into a matrix, where each column represents a unique vocabulary term and each row represents a pair of questions. This numerical representation of the queries enables the use of machine learning techniques.

D. *Model Training and Evaluation*

Two machine learning models, a Random Forest Classifier and an XGBoost Classifier, were trained to distinguish between duplicate and non-duplicate question pairs. The models were trained using the specified features. The dataset was divided into training and testing sets in an 80-20 ratio. The models were then trained on the training set and their performance was evaluated using accuracy as the metric.On the test dataset, the RandomForestClassifier achieved an accuracy of approximately 78.9%, while the XGBClassifier performed slightly better with an accuracy of around 79.1%. Further analysis using confusion matrices provided more insights into the models' performance. The RandomForestClassifier accurately identified 3289 non-duplicate pairs and 1445 duplicate pairs, despite misclassifying 539 non-duplicate pairings as duplicates and 727 duplicate pairs as non-duplicates. In comparison, the XGBClassifier correctly detected 3225 non-duplicate pairs and 1519 duplicate pairs, but incorrectly categorized 603 non-duplicate pairs and 653 duplicate pairs.

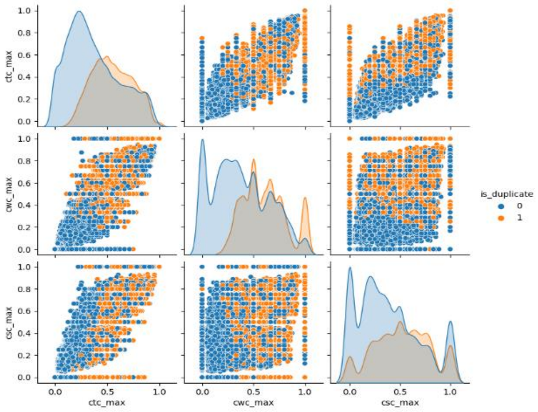
- ***Pairplot for Common Words Count (cwc_min), Common Stop Words Count (csc_min), and Token Count (ctc_min)***

The pairplot visualizes the relationship between the minimum values of the common word count, common stop word count, and token count in question pairs. The color of the plot indicates whether the questions are duplicates or not.



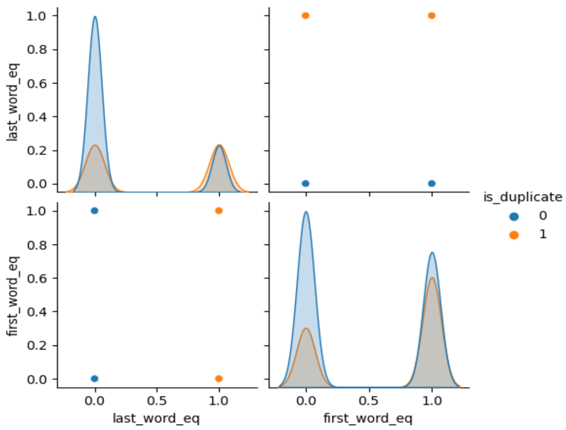
- ***Pairplot for Common Words Count (cwc_max), Common Stop Words Count (csc_max), and Token Count (ctc_max)***

The provided plot is similar to the previous pairplot, depicting the correlations between the highest values of the counts of common words, common stop words, and tokens in the question pairs. The color of each plot signifies whether the questions are duplicates or not.



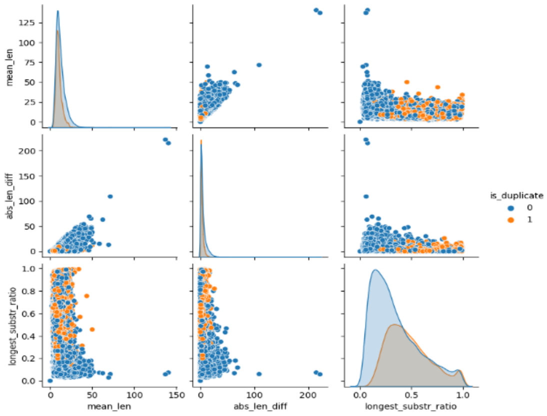
- ***Pairplot for Last Word Equality and First Word Equality***

The color of the pairplot represents whether the questions are duplicates or not, and it demonstrates the relationship between the first word "equality" and the last word "equality" features in the question pairs.



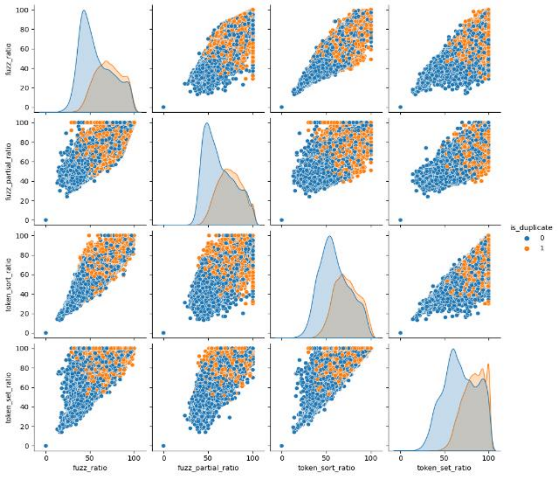
- ***Pairplot for Mean Length, Absolute Length Difference, and Longest Substring Ratio***

The pairplot displays the longest substring ratio, average length, and absolute length difference characteristics in the question pairs. The color coding indicates whether the questions are duplicates or not.



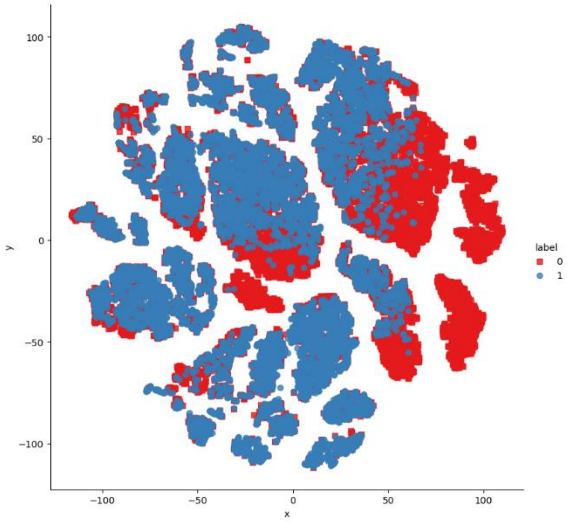
- ***Pairplot for Fuzzy Features (fuzz_ratio, fuzz_partial_ratio,token_sort_ratio,token _set_ratio)***

This pairplot shows the associations between different fuzzy matching features in question pairs, including fuzz_ratio, fuzz_partial_ratio, token_sort_ratio, and token_set_ratio. The hue of the questions indicates whether or not they are duplicates.



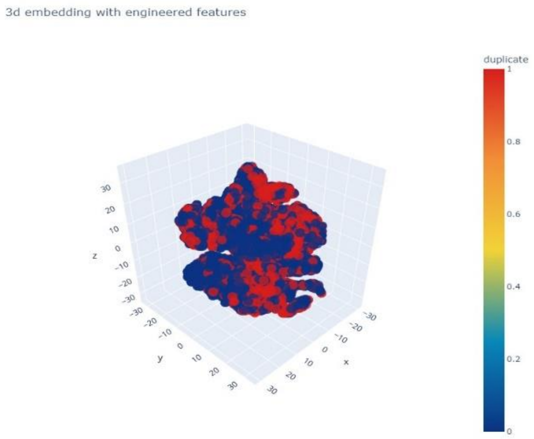
- ***2D TSNE Plot***

It describes the use of t-distributed Stochastic Neighbour Embedding (t-SNE) to reduce the dimensionality of the features generated after data cleaning. This technique projects the features onto a two-dimensional space, and the points are colored based on whether the corresponding questions are duplicates or not.



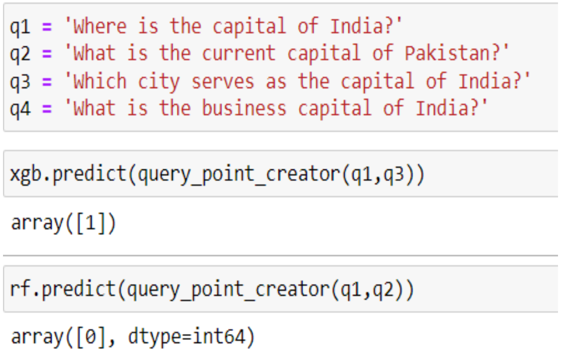
- ***3D TSNE Plot***

This plot employs t-SNE for dimensionality reduction, similar to the 2D TSNE plot, but it presents the features in three dimensions. The points are colored based on whether the questions are duplicates or not.



E. Query Point Prediction

We have developed a function that employs the same approach as the training phase to extract features and convert text inquiries into numerical representations, thereby creating a query point. Leveraging the trained classifiers, this function allows us to predict whether a new pair of questions is a duplicate. This feature can then be utilized in a practical application to automatically detect duplicate questions on a question-and-answer platform.



```
q1 = 'Where is the capital of India?'
q2 = 'What is the current capital of Pakistan?'
q3 = 'Which city serves as the capital of India?'
q4 = 'What is the business capital of India?'
```

```
xgb.predict(query_point_creator(q1,q3))
```

```
array([1])
```

```
rf.predict(query_point_creator(q1,q2))
```

```
array([0], dtype=int64)
```

IV. CONCLUSION

The machine learning models utilized in this study exhibit potential in identifying whether question pairs are duplicates. We were successful in capturing the semantic resemblance between question pairs by constructing a diverse array of features, including token-based, length-based, and fuzzy features. Additionally, we incorporated Bag-of-Words representations for both questions. Significant accuracy levels were achieved by the Random Forest and XGBoost classifiers, showcasing their effectiveness in distinguishing between duplicate and non-duplicate items.

V. FUTURE WORK

Future studies may continue enhancing the models' resilience and forecasting capabilities. Exploring more advanced natural language processing approaches, such as transformer-based architectures like BERT or deep neural network models like Siamese networks, which excel at identifying intricate semantic relationships in text, represents one potential direction for improvement. The models' understanding of the subtle complexities of language could be improved by incorporating contextual embeddings or pre-trained language models. Additionally, leveraging data from other sources, such as user interactions or question metadata, may provide valuable insights for enhancing model performance. The ongoing refinement and optimization of feature engineering techniques and model hyperparameters may eventually lead to even more accurate and reliable forecasts. Overall, these areas for further investigation have the potential to enhance the effectiveness of systems for detecting duplicate questions and push the limits of what is currently achievable in natural language understanding tasks.

REFERENCES

[1] Bian, J., Liu, Y., & Agichtein, E. (2008). Finding the right facts in the crowd: Factoid question answering over social media. In Proceedings of the 17th international conference on World Wide Web (pp. 467-476).

[2] Jeon, J., Croft, W. B., & Lee, J. H. (2005). Finding similar questions in large question and answer archives. In Proceedings of the 14th ACM international conference on Information and knowledge management (pp. 84-90).

[3] Wang, Y., Huang, M., Zhu, X., Zhao, L., & Zhang, L. (2017). Bilateral multi-perspective matching for natural language sentences. In Proceedings of the 26th international conference on World Wide Web (pp. 343-352).

[4] Tian, Z., He, B., & Chen, Z. (2018). A deep learning approach for question matching in online forums. Information Sciences, 423, 74-84.

[5] Huang, Z., Xu, W., & Yu, K. (2013). Learning to rank for question-answering systems. In Proceedings of the 22nd ACM international conference on Information & Knowledge Management (pp. 2383-2388).

[6] Hasan, K. S., Ahmad, A., Farooq, U., & Baig, A. R. (2016). Duplicate question detection in stack overflow using supervised learning. In 2016 IEEE 28th International Conference on Tools with Artificial Intelligence (ICTAI) (pp. 815-820).