

Major Project Report

# **Duplicate Questions Pair Identification on Quora Using NLP**

*Submitted in partial fulfilment of requirements for the award of the*

*Degree of*

**Bachelor of Technology**

**In**

**Information Technology**

**Submitted By**

**Shwetank Chaudhary**

*(07811503120)*

**Under the guidance of**

**Dr. Surinder Kaur**

**(Associate Professor)**



**Department of Information Technology**  
**Bharati Vidyapeeth's College of Engineering, New Delhi – 110063, INDIA**

**May 2024**

## CANDIDATE'S DECLARATION

---

It is hereby certified that the work which is being presented in the B.Tech Major Project Report entitled "**Duplicate Questions Pair Identification on Quora Using NLP**" in partial fulfilment of the requirements for the award of the degree of **Bachelor of Technology** and submitted in the **Department of Information Technology** of **BHARATI VIDYAPEETH'S COLLEGE OF ENGINEERING, New Delhi** (Affiliated to **Guru Gobind Singh Indraprastha University, Delhi**) is an authentic record of my own work carried out during a period from **February 2024 – May 2024** under the guidance of , **Dr. Surinder Kaur**.

The matter presented in the B. Tech Major Project Report has not been submitted by me for the award of any other degree of this or any other Institute.

**Student Name: Shwetank Chaudhary**  
**Enrollment No: 07811503120**

This is to certify that the above statement made by the candidate is correct to the best of my knowledge. He/She/They are permitted to appear in the External Major Project Examination.

**Dr. Surinder Kaur**  
Associate professor

**Prof. (Dr) Prakhar Priyadarshi**  
HOD, IT

# ABSTRACT

---

Duplicate question identification is a critical task in natural language processing (NLP), particularly in online platforms like Quora, where ensuring content quality and enhancing user experience are paramount. This project aims to develop an efficient model for identifying duplicate question pairs on Quora, leveraging state-of-the-art NLP techniques and machine learning algorithms.

The motivation behind this project stems from the need to improve search relevance, content recommendation, and spam detection on Quora by accurately identifying duplicate questions. Through a comprehensive literature review, existing methodologies and challenges in duplicate question identification are explored, providing valuable insights for the development of novel approaches.

The project utilizes a dataset collected from Quora, which is pre-processed to handle noise, imbalance, and semantic ambiguity. Feature engineering techniques such as TF-IDF, word embeddings, and semantic similarity measures are employed to extract meaningful representations of the text data. Various machine learning models, including traditional algorithms and deep learning architectures, are considered and evaluated for their effectiveness in identifying duplicate question pairs.

Challenges such as dataset imbalance, semantic ambiguity, and computational resource limitations are addressed through careful experimentation and model optimization. Ensemble techniques are explored to further improve the robustness and performance of the developed model.

The proposed system has potential applications beyond Quora, including enhancing search relevance, content recommendation, and spam detection in other online platforms and information retrieval systems. The findings of this project contribute to the body of research in NLP and provide practical solutions for improving user-generated content moderation and management.

In conclusion, this project presents a comprehensive approach to duplicate question pair identification on Quora, offering valuable insights, methodologies, and applications for enhancing content quality and user experience in online platforms.

## ACKNOWLEDGEMENT

---

We express our deep gratitude to **Dr. Surinder Kaur**, Associate Professor, Department of Information Technology for her valuable guidance and suggestions throughout my project work.

We would like to extend sincere thanks to the Head of Department, **Prof. (Dr.) Prakhar Priyadarshi** for his time-to-time suggestions to complete my project work. We are also thankful to **Prof. (Dr.) Dharmendra Saini**, Principal for providing me the facilities to carry out my project work.

Shwetank Chaudhary

(07811503120)

# TABLE OF CONTENTS

---

CANDIDATE DECLARATION.....	ii
ABSTRACT.....	iii
ACKNOWLEDGEMENT.....	iv
TABLE OF CONTENTS.....	v
LIST OF FIGURES.....	viii

## Chapter 1: Introduction 1 – 11

1.1	Background	1
1.1.1	Significance of this project	1
1.1.2	Objective	2
1.1.3	Scope and Limitation	2
1.1.4	Structure of Report	2
1.2	Application of Duplicate Question pair identification	3
1.3	Future Trends	4
1.4	Challenges Faced	5
1.4.1	Contextual Understanding	5
1.4.2	Ambiguity and Polysemy	5
1.4.3	Sarcasm and Irony	5
1.4.4	Biases in Training data	5
1.4.5	Multilingual Data	6
1.4.6	Privacy Concerns	6
1.4.7	Emotional Subjectivity	6
1.4.8	Ethical Considerations	6
1.4.9	Addressing Challenges	6
1.4.10	Future Directions	7
1.5	System Requirements	8
1.5.1	Introduction	8
1.5.2	Hardware Requirements	8

	1.5.3	Software Requirements	8
	1.5.4	Development Requirement	9
	1.5.5	Development & Maintainence	9
	1.5.6	Scalability Considerations	9
	1.6	Technology Stack & Libraries Used	10
	1.6.1	Programming Language	10
	1.6.2	Libraries & Framework	11
	1.6.3	Conclusion	11
<b>Chapter 2:</b>	<b>Literature Review</b>		<b>12 – 13</b>
	2.1	Duplicate Question Identification: Definition and Challenges	12
	2.2	Traditional Approaches	12
	2.3	Machine Learning Approaches	12
	2.4	Deep Learning Approaches	12
	2.5	Hybrid Approaches	13
	2.5.1	Evaluation Metrics	13
	2.6	Challenges & Consideration	13
	2.6.1	Future Direction	13
<b>Chapter 3:</b>	<b>Data Collection &amp; Preprocessing</b>		<b>14 – 15</b>
	3.1	Introduction	
	3.1.2	Data Collection Sources & Importance	
	3.1.3	Our Dataset	
	3.1.4	Dataset & Preprocessing	
<b>Chapter 4:</b>	<b>Research Methodology</b>		<b>16 – 27</b>
	4.1	Our Approaches & Work Flow	16
	4.1.1	Basic Data Analysis	16
	4.1.2	Feature Engineering	17
	4.1.3	Exploratory Analysis of newly added Feature	17
	4.2	Machine Learning Model Creation	21
	4.2.1	Introduction	21
	4.2.2	Separate the independent & Dependent Features	21
	4.2.3	Vectorizing the Feature	21
	4.2.4	Train-Test Split	21
	4.2.5	Train the machine Learning model	22
	4.3	Baseline Model Performance	22
	4.4	Optimizing the Model	22
	4.4.1	Text Preprocessing	22
	4.5	Advance Feature Engineering	23
	4.5.1	EDA & New Features	24

4.6	Dimensionality Reduction	26
<b>Chapter 5: Results</b>		
5.1	Observation	28
5.2	Application Screenshots	29 - 30
<b>Conclusion &amp; Future Scope</b>		31
<b>References</b>		33

## **LIST OF FIGURES**

---

Fig 4.1 Question Distribution

Fig 4.2 Distribution of no of words

Fig 4.3 Analysis of Common Words

Fig 4.4 Analysis of Total Words

Fig 4.5 Analysis of Words Share

Fig 4.6 Analysis of newly Added Features

Fig 4.7 Analysis of first Word & Last Word

Fig 4.8 Analysis of Length Based Features

Fig 4.9 Analysis of Fuzzy Features

Fig 4.10 TSNE For Data Exploration

Fig 5.1 Application Screenshot 1 : Streamlit Front-End

Fig 5.2 Application Screenshot 2 : Not Duplicate Found Page



# Chapter 1 : Introduction

---

## 1.1 Background

In the digital age, online platforms serve as hubs for information exchange, social interaction, and knowledge sharing. Quora, one such platform, facilitates the dissemination of knowledge through user-generated questions and answers. However, the quality and relevance of content on Quora heavily depend on the accuracy of question identification, particularly in cases of duplicate or highly similar questions. Duplicate questions not only clutter the platform but also hinder the user experience by leading to redundant content and diluting valuable discussions.

### 1.1.1 Significance of this project

The motivation behind this project lies in the need to improve content quality and user experience on Quora by addressing the challenge of duplicate question identification. By accurately identifying and filtering out duplicate question pairs, Quora can enhance search relevance, content recommendation, and user engagement. Additionally, mitigating duplicate questions can streamline content moderation efforts and contribute to a more organized and informative platform.

The significance of this project is multifaceted, primarily aiming to enhance the quality and relevance of user-generated content on Quora, a popular online platform for knowledge sharing. By effectively identifying and managing duplicate question pairs, this project contributes to a cleaner and more informative user experience. Users benefit from reduced clutter and redundancy, enabling them to discover diverse and relevant content more efficiently. Additionally, the project addresses practical concerns such as improving search relevance, enhancing content recommendation algorithms, and bolstering spam detection mechanisms. Moreover, by optimizing resource utilization and content moderation efforts, Quora can maintain its integrity as a trusted source of information and foster a more engaging and productive community environment. Ultimately, the insights and methodologies developed in this project have broader implications for content management and quality control across various online platforms, highlighting its significance in the context of modern information sharing and online interaction.

### **1.1.2 Objectives of the Report**

The primary objective of this project is to develop an efficient and robust model for identifying duplicate question pairs on Quora. Specifically, the project aims to:

1. Investigate existing methodologies and challenges in duplicate question identification.
2. Collect and preprocess a dataset of question pairs from Quora.
3. Explore feature engineering techniques to extract meaningful representations of text data.
4. Evaluate various machine learning models and deep learning architectures for their effectiveness in duplicate question identification.
5. Address challenges such as dataset imbalance, semantic ambiguity, and computational resource limitations.
6. Develop an ensemble model to enhance the robustness and performance of the duplicate question identification system.
7. Demonstrate the practical applications of the developed system in improving search relevance, content recommendation, and spam detection on Quora.

### **1.1.3 Scope and Limitation**

This project focuses specifically on the task of identifying duplicate question pairs on Quora. While the developed methodologies and techniques may have broader applications in NLP and information retrieval, the scope of this project is limited to the Quora platform. Additionally, the effectiveness of the duplicate question identification system may be influenced by factors such as dataset quality, language diversity, and evolving user behaviour on Quora.

### **1.1.4 Structure of the Report**

The subsequent chapters of this report will delve into each of these objectives, offering a comprehensive and nuanced exploration of Natural language processing. Chapter 2 will provide an in-depth review of Literature Survey, Chapter 3 will focus on dataset collection sources and its importance and preprocessing, Chapter 4 will discuss the research methodologies done during the project development, and Chapter 5 will discuss the results observation and key findings. The final chapter will conclude our exploration, summarizing key findings and providing insights into the future of this project.

In conclusion, this chapter establishes the foundation for a thorough investigation into duplicate question identification—a field positioned at the intersection of natural language processing, machine learning, and online community dynamics. The subsequent chapters will navigate through the intricacies of duplicate question identification, unravelling its significance and impact on diverse facets of our interconnected online world.

## **1.2 Applications of Duplicate Question Pair Identification**

Duplicate question pair identification, with its ability to identify and classify similar questions within textual data, finds diverse applications across industries, influencing decision-making processes and enhancing user experiences. This chapter explores the manifold applications of duplicate question pair identification, showcasing its versatility and impact in domains ranging from content moderation and customer support to information retrieval and AI-driven chatbots.

These applications highlight the broad impact of question pair identification in streamlining information retrieval, enhancing AI capabilities, improving user experiences, and optimizing various business processes. The accuracy and efficiency of question pair identification methods directly contribute to the effectiveness of these applications across different domains.

### **1. Content Moderation and Quality Control**

Platforms like Quora, Stack Exchange, and community forums use question pair identification to detect and remove duplicate questions, ensuring content quality and reducing redundancy.

### **2. Information Retrieval and Search Engines**

Search engines leverage question pair identification to improve search relevance by grouping similar queries and providing more accurate and diverse search results.

### **3. Customer Support and FAQ Systems**

Companies use question pair identification to streamline customer support processes by automatically matching user queries with existing FAQs or support articles.

### **4. E-Learning and Education**

Educational platforms utilize question pair identification to recommend relevant study materials, identify similar questions for practice, and personalize learning experiences.

### **5. Chatbots and Virtual Assistants**

Chatbots and virtual assistants use question pair identification to understand user intents and respond appropriately, improving conversational AI interactions.

### **6. Text Summarization and Information Extraction**

NLP applications benefit from question pair identification to summarize text, extract key information, and generate concise responses based on similar questions.

## **7. Community Engagement and User Experience**

By identifying duplicate questions, platforms enhance community engagement by promoting original content and reducing clutter, leading to a better user experience.

## **8. Legal and Compliance Document Review**

Legal professionals use question pair identification for document review tasks, identifying duplicate or similar clauses, contracts, or legal queries efficiently.

### **1.3 Future Trends**

The chapter concludes with a glimpse into the future trends of Duplicate Question Pair Identification Using NLP, exploring potential advancements, such as Duplicate Question pair identification analysis in emerging technologies, personalized Duplicate Question pair identification models, and increased focus on industry-specific applications.

In summary, this chapter illuminates the myriad applications of Duplicate Question Pair Identification, showcasing its transformative impact across industries. As we delve deeper into the implications and challenges, the subsequent chapters will provide a holistic understanding of data analysis in diverse contexts.

## **1.4 Challenges in Duplicate Question Pair Identification**

As Natural language processing continues to evolve as a critical tool in understanding human emotions through textual data, it faces a spectrum of challenges that necessitate nuanced solutions. This chapter explores the inherent difficulties in Question Pair analysis, ranging from contextual intricacies to ethical considerations, and discusses ongoing efforts to address these challenges.

### **1.4.1 Contextual Understanding**

One of the primary challenges in Question Pair analysis lies in the nuanced nature of language. Words and expressions often derive their meaning from the context in which they are used. Sentences, phrases, or even individual words can convey different Duplicate Question pair identifications based on their surrounding context. This challenge requires Duplicate Question pair identification analysis models to develop a deeper contextual understanding for accurate interpretation.

### **1.4.2 Ambiguity and Polysemy**

The inherent ambiguity in language, characterized by words having multiple meanings, poses a significant hurdle for Duplicate Question pair identification analysis models. Polysemy, where a single word carries different meanings, can lead to misinterpretations. Resolving this challenge involves context disambiguation and refining models to discern the intended meaning within a specific context.

### **1.4.3 Sarcasm and Irony**

Expressions of sarcasm and irony present a formidable challenge for Duplicate Question pair identification analysis. The literal meaning of words may convey one Duplicate Question pair identification, while the intended meaning could be the opposite. Deciphering these subtle linguistic cues requires advanced models capable of recognizing patterns and understanding the speaker's intent.

### **1.4.4 Biases in Training Data**

Duplicate Question pair identification analysis models are trained on datasets that inherently reflect the biases present in the data sources. Biases stemming from cultural, social, or demographic factors can result in skewed predictions, impacting the model's accuracy and fairness. Addressing biases involves careful curation of training data and ongoing efforts to mitigate unfair outcomes.

### **1.4.5 Multilingual Challenges**

The global nature of textual data introduces multilingual challenges in Duplicate Question pair identification analysis. Differences in language structures, idioms, and Duplicate Question pair identification expressions necessitate models capable of handling diverse linguistic nuances. Cross-lingual Duplicate Question pair identification analysis is an evolving field that aims to address these challenges effectively.

### **1.4.6 Privacy Concerns**

The analysis of Duplicate Question pair identifications expressed in personal or private communications raises ethical concerns related to privacy. Duplicate Question pair identification analysis applications must navigate the delicate balance between extracting insights and respecting individual privacy rights. Striking this balance requires robust privacy policies and transparent communication about data usage.

### **1.4.7 Emotional Subjectivity**

Capturing the subjectivity inherent in human emotions presents a challenge for Duplicate Question pair identification analysis models. Individual experiences and perspectives contribute to the subjective nature of Duplicate Question pair identifications, making it difficult to create universally applicable models. Research efforts focus on developing models that can adapt to the emotional subjectivity present in diverse datasets.

### **1.4.8 Ethical Considerations**

The ethical use of Duplicate Question pair identification analysis is paramount. Issues such as inadvertent reinforcement of stereotypes, unintended consequences of decision-making based on Duplicate Question pair identification analysis outputs, and the potential for misuse require vigilant consideration. Ethical guidelines and frameworks are crucial in guiding the responsible deployment of Duplicate Question pair identification analysis.

### **1.4.9 Addressing Challenges: Ongoing Research**

Researchers actively engage in addressing these challenges through innovative approaches. Ongoing efforts involve the development of more sophisticated algorithms, the creation of diverse and unbiased datasets, and the implementation of explainable AI techniques to enhance the transparency and interpretability of Duplicate Question pair identification analysis models.

#### **1.4.10 Future Directions**

The chapter concludes by envisioning future directions in overcoming Duplicate Question pair identification analysis challenges. Embracing advancements in explainable AI, continual refinement of models through adversarial training, and increased collaboration between researchers and industry practitioners will shape the trajectory of Duplicate Question pair identification analysis in the years to come.

In summary, this chapter illuminates the multifaceted challenges inherent in Question Pair analysis. Acknowledging and addressing these challenges is integral to advancing the field and ensuring the responsible application of Language analysis across various domains. The subsequent chapters will build upon this foundation, exploring technological advancements and ethical considerations in Duplicate Question pair identification analysis.

## **1.5 System Requirements for Duplicate Pair Identification**

### **1.5.1 Introduction**

Implementing a system for duplicate question pair identification requires careful consideration of the underlying technology stack, computational resources, and software dependencies. This chapter outlines the system requirements essential for deploying and running duplicate question pair identification models effectively.

### **1.5.2 Hardware Requirements**

#### **i. CPU**

A modern multicore processor is recommended for handling the computational demands of Language analysis tasks. The choice of CPU depends on the scale of the application and the volume of incoming textual data.

#### **ii. Memory (RAM)**

A sufficient amount of RAM is crucial for storing and processing large datasets efficiently. The amount of RAM required depends on the size of the datasets and the complexity of the Language analysis models.

#### **iii. Storage**

Adequate storage space is necessary to store datasets, pre-trained models, and any additional resources. The choice of storage capacity depends on the size of the datasets and the frequency of model updates.

### **1.5.3 Software Requirements**

#### **i. Operating System**

The choice of the operating system depends on the preferences of the development and deployment team. Language analysis models can be implemented on Windows, Linux, or macOS.

#### **ii. Python**

Python is a widely used programming language for natural language processing and Language analysis. The system should have a compatible version of Python installed. Python libraries such as NLTK, seaborn, scikit-learn, and others may be required based on the chosen Duplicate Question pair identification analysis approach.



**iii. Machine Learning tools**

The specific Language analysis libraries chosen for implementation will dictate additional software requirements. For instance, NLTK and its dependencies should be installed.

#### **1.5.4 Development Environment**

**i. Integrated Development Environment (IDE)**

A suitable IDE such as Jupyter Notebook, Visual Studio Code, or PyCharm is recommended for developing and testing Language analysis models. These environments provide tools for efficient code development and debugging.

**ii. Version Control**

Utilizing version control systems like Git is advisable for tracking changes in the codebase, collaborating with team members, and maintaining a history of model updates.

#### **1.5.5 Deployment and Maintenance Requirements**

Discussion of deployment and maintenance considerations for deploying the duplicate question pair identification system in production.

Deployment Strategy: Description of deployment procedures (e.g., cloud deployment, containerization) and configuration management practices. Monitoring and Maintenance: Specification of monitoring tools (e.g., logging, performance metrics) and maintenance procedures to ensure system uptime and reliability.

#### **1.5.6 Scalability Considerations**

If the Duplicate Question Pair identification system is expected to handle increasing amounts of data over time, considerations for scalability should be integrated into the system architecture. This may involve deploying the system on cloud infrastructure that can be scaled dynamically.

## **1.6 Technology Stack and Libraries Used**

In the development of the Duplicate Question pair identification analysis project, a diverse set of technologies, libraries, and programming languages were employed to create a robust and feature-rich application. This chapter provides an overview of the key components that constitute the technology stack.

### **1.6.1 Programming Language**

#### **Python**

Python was the primary programming language for the Duplicate Question Pair identification project. Its simplicity, readability, and extensive ecosystem of libraries made it an ideal choice for implementing natural language processing tasks and developing the Streamlit-based web application.

### **1.6.2 Libraries and Frameworks**

#### **i. Streamlit**

Streamlit was instrumental in building the user interface for the Duplicate Question pair identification analysis application. Its intuitive syntax and ability to turn data scripts into shareable web apps with minimal code made it an excellent choice for creating an interactive and user-friendly platform.

#### **ii. Streamlit Option Menu**

The Streamlit Option Menu library enhanced the user interface by providing customizable option menus, contributing to a more polished and dynamic user experience.

#### **iii. Streamlit Lottie**

Streamlit Lottie facilitated the integration of Lottie animations into the application. This library added a visually engaging element to the user interface, enhancing the overall aesthetics.

#### **iv. Pandas**

Pandas, a powerful data manipulation library, was used for handling and processing datasets. Its DataFrame structure allowed for efficient data manipulation and analysis.

**v. NumPy**

NumPy, a fundamental library for numerical operations in Python, played a crucial role in handling numerical data and supporting various mathematical operations.

**vi. Word Cloud**

The WordCloud library was employed to generate word clouds visualizing the most frequent words in the analysed tweets. It contributed to a more comprehensive understanding of the prevalent Duplicate Question pair identifications.

**vii. Seaborn and Matplotlib**

Seaborn and Matplotlib were used for data visualization. These libraries provided a range of plotting options, including bar charts and heatmaps, to convey Duplicate Question pair identification insights effectively.

### **1.6.3 Conclusion**

The technology stack and libraries chosen for the Duplicate Question pair identification analysis project were carefully selected to ensure a balance between functionality, user experience, and data analysis capabilities. The combination of Streamlit, Pandas and other libraries empowered the development of a versatile and accessible Duplicate Question pair identification analysis application. This chapter highlights the diverse tools and technologies that came together to create a cohesive and impactful solution for Duplicate Question Pair Identification.

# Chapter 2: Literature Review

---

Duplicate question identification is a crucial task in natural language processing (NLP), with significant implications for improving the quality of user-generated content on online platforms. This chapter presents a thorough review of existing literature and methodologies relevant to duplicate question identification, focusing on techniques employed in NLP and machine learning.

## 2.1 Duplicate Question Identification: Definitions and Challenges

This section begins by defining the concept of duplicate question identification and highlighting the challenges associated with this task. Challenges include variations in language usage, semantic ambiguity, and the need to balance precision and recall in identifying duplicate pairs effectively.

## 2.2 Traditional Approaches

Traditional approaches to duplicate question identification typically rely on heuristic-based methods and similarity measures. Techniques such as cosine similarity, Jaccard similarity, and edit distance have been widely used to compare question pairs based on lexical, syntactic, and semantic similarities.

## 2.3 Machine Learning Approaches

Machine learning approaches have gained prominence in recent years for their ability to learn complex patterns and representations from data. Supervised learning techniques, including logistic regression, support vector machines (SVM), and decision trees, have been applied to classify question pairs as duplicate or non-duplicate based on labeled training data.

## 2.4 Deep Learning Approaches

Deep learning models, particularly neural networks, have revolutionized Duplicate Question pair identification analysis. Recurrent Neural Networks (RNNs) and Long Short-Term Memory networks (LSTMs) capture sequential dependencies in text, while Transformer-based models like BERT excel in contextual understanding. The hierarchical nature of deep learning architectures enables the extraction of intricate features, contributing to more nuanced Duplicate Question pair identification analysis.

## **2.5 Hybrid Approaches**

Hybrid methodologies combine elements of rule-based systems and machine learning to harness the strengths of both. These approaches often integrate linguistic rules with machine learning models, striking a balance between interpretability and predictive power. Hybrid models are adept at handling specific domain knowledge and idiosyncrasies in language expression.

## **2.6 Evaluation Metrics**

Evaluation metrics play a crucial role in assessing the performance of duplicate question identification systems. Common metrics include accuracy, precision, recall, F1-score, and area under the receiver operating characteristic curve (AUC-ROC). These metrics provide insights into the system's ability to correctly identify duplicate question pairs while minimizing false positives and false negatives.

## **2.7 Challenges and Considerations**

Despite significant advancements in duplicate question identification, several challenges and open issues remain. These include handling multi-language datasets, addressing domain-specific variations, mitigating the impact of imbalanced datasets, and scaling algorithms to handle large volumes of data efficiently.

While these methodologies offer diverse tools for identification, challenges persist. Contextual understanding, ambiguity, and cultural nuances pose hurdles for accurate interpretation. Additionally, biases in training data can result in skewed predictions, warranting continuous refinement of models and ethical considerations.

### **2.7.1 Future Directions**

The chapter concludes by highlighting potential avenues for future research and development in duplicate question identification. Areas of interest include exploring ensemble methods to combine multiple models, integrating domain knowledge into the modelling process, and investigating the impact of pre-trained language models on system performance. This chapter provides a comprehensive overview of existing literature and methodologies in duplicate question identification, laying the groundwork for the subsequent chapters' exploration of techniques and methodologies applied in the project.

# Chapter 3: Dataset Collection and preprocessing

---

## 3.1 Introduction

In the realm of duplicate question pair identification, the availability of high-quality data plays a critical role in the effectiveness and performance of machine learning models. This chapter delves into the intricacies of data collection and preprocessing, highlighting the significance of robust data preparation for accurate and reliable analysis.

### 3.1.2 Data Collection Sources and Importance

Data collection for duplicate question pair identification often involves sourcing datasets from various platforms and repositories where user-generated content is abundant. Popular sources for such datasets include:

1. **Quora:** As a leading platform for knowledge sharing and question answering, Quora provides a rich source of question pairs that can be used to train and evaluate models for duplicate question identification.
2. **Stack Exchange:** This network of Q&A websites covers a wide range of topics, making it another valuable resource for collecting question pairs for research purposes.
3. **Online Forums and Communities:** Diverse online communities and forums contribute to the pool of user-generated content, offering insights into common question patterns and similarities.

The importance of data collection in this context cannot be overstated. Access to a diverse and representative dataset enables researchers to develop models that generalize well to real-world scenarios. Moreover, comprehensive data collection ensures the inclusivity of various linguistic styles, topics, and contexts, enhancing the model's ability to accurately identify duplicate question pairs across different domains. The selection of an appropriate dataset is crucial for ensuring the validity and applicability of research findings. Researchers must consider factors such as dataset size, diversity, and annotation quality when choosing a dataset for their studies.

### 3.1.3 Our Dataset

For this project we make use of the publicly available Quora Question Pairs dataset available on Kaggle. For our experiments, we make use of the training set provided by the website. The training set consists of 4,04,290 labeled question pairs. The fields in the dataset are shown in TABLE 2. Of all the question pairs, 149302 are duplicates, or roughly 37% of the full dataset. We assume that questions marked as duplicates in the Quora dataset are semantically equivalent since Quora's duplicate question policy concurs with our definition of semantic equivalence above. The dataset has been labeled

manually by humans. Hence there is bound to be some noise in the labels. We have split our data into three sets of training, validation, and testing sets. The training set has 3,04,290 entries whereas the validation and test sets contain 50,000 entries each.

**TABLE 3.1: Field Description of our Dataset**

Fields	Description
id	unique identifier for the question pair
qid1	qid1 unique identifier for the first question
qid2	qid2 unique identifier for the second question
question1	question1 full unicode text of the first question
question2	question2 full unicode text of the second question
is duplicate	is duplicate 1 if questions are duplicates, 0 otherwise

### 3.1.4 Dataset and Preprocessing

Since the classifier is only concerned with “question1”, “question2” and “Is\_duplicate”, the rest of the attributes of the dataset are ignored.

The steps required for organizing the data in understandable format by handling the missing, inconsistent and redundant values is called preprocessing. Various pre-processing steps are performed on experimental dataset. Several NLP techniques are used such as conversion to lower letters of text, stop-words removal, stemming, and tokenization, with the help of freely available libraries such as NLTK and keras’s.

**TABLE 3.2: Sample Date in our Dataset**

id	qid	qid2	question1	question 2	is duplicate
234	256	257	How can I be a good geologist?	What should I do to be a great geologist?	1
345	356	357	How do I read and find my YouTube comments?	How can I see all my Youtube comments?	1
678	456	467	Which is the best digital marketing institution in banglore?	Which is the best digital marketing institute in Pune?	0
876	789	800	Do smart people brag?	Why do smart people think dumb?	0
890	921	922	What is f and o in a stock market?	What is F&O in stock market?	1

# Chapter 4: Research Methodology

---

## 4.1 Our Approaches and Work Flow

In this chapter, we delve into the methodologies employed to conduct a comprehensive study on Duplicate Question pair identification analysis. A well-structured research methodology is fundamental to the credibility and reliability of any scientific investigation. It serves as the blueprint for the entire research process, outlining the procedures, tools, and techniques used to gather, analyze, and interpret data.

### 4.1.1 Basic data Analysis

Basic Data Analysis, often referred to as Exploratory Data Analysis (EDA), is a fundamental process in data science that involves thoroughly examining and understanding a dataset before proceeding to more complex analyses or modelling tasks. The primary goal of EDA is to extract insights and uncover hidden patterns, relationships, and anomalies within the data.

The workflow of basic data analysis begins with data collection and understanding, where the dataset is obtained and its structure is comprehended, including the types and meanings of variables. Following this, data cleaning and preprocessing are performed to handle missing values, remove duplicates, and standardize formats, ensuring that the dataset is ready for analysis.

Descriptive statistics are then calculated to summarize the dataset, providing initial insights into the distribution of numeric variables and the frequency of categorical variables. Visualization plays a crucial role in EDA, allowing for the creation of graphical representations such as histograms, box plots, scatter plots, and heatmaps to explore relationships and identify trends, outliers, and potential patterns.

Feature engineering is conducted to derive new features or transform existing ones based on insights gained during analysis. Correlation analysis helps understand relationships between variables, identifying highly correlated features that may impact modelling. Additionally, exploratory techniques like clustering or anomaly detection can reveal meaningful patterns or anomalies within the data.



### 4.1.2 Feature Engineering

Feature engineering is a crucial technique in data science aimed at improving model accuracy by creating new, meaningful features from existing data. This process involves transforming, scaling, extracting, and encoding features to enhance predictive power.

In our next step, we will augment our dataset by adding seven new features using the bag of words model applied to pairs of questions. The bag of words [5] model converts textual data (questions in this case) into numerical feature vectors based on word frequencies. For each question pair, this model will generate unique features representing the similarity or overlap of words between the questions. These new features will undergo exploratory data analysis (EDA) to understand their impact and relationship with other variables. Subsequently, they will be incorporated into our machine learning model to improve its predictive capability, leveraging the insights gained from EDA.

This iterative process of feature engineering and analysis plays a pivotal role in refining our dataset and optimizing model performance for effective prediction and decision-making.

- i. **Question Length:** The size of the question is a critical feature because when we vectorize it, the question gets split by words, so having the length feature is good. The length we are having is the character-wise length. So, it will create 2 new features for the length of questions 1 and 2.
- ii. **Number of words:** The number of words in both questions is another feature that should impact the model performance. So, it will add 2 new features for questions 1 and 2. To add the feature, split the sentence with space and extract the length of the list.
- iii. **Common words:** Another feature is to know how many common words there are in both questions. It helps identify the similarity between both questions. Calculating where you only need to apply the intersection between both questions is simple. For this, we find the number of unique words in both questions and apply the set intersection to the set length.
- iv. **Total words:** The sum of the total number of unique words in each question. In simple terms, find the number of unique words in both questions and return their sum.
- v. **Word share:** It is one exciting feature and simple to add. To calculate, divide the common words by the total number of words

### 4.1.3 Exploratory Analysis of Newly Added Features

We have introduced some new features in the dataset, and it is an excellent time to analyze the relationship and their spread with the target variable.

#### i. Distribution of Question

We will analyze the length of the question, like the average length of each question in 1 and 2. Minimum and maximum number of characters. So, plot the distribution plot for questions 1 and 2.

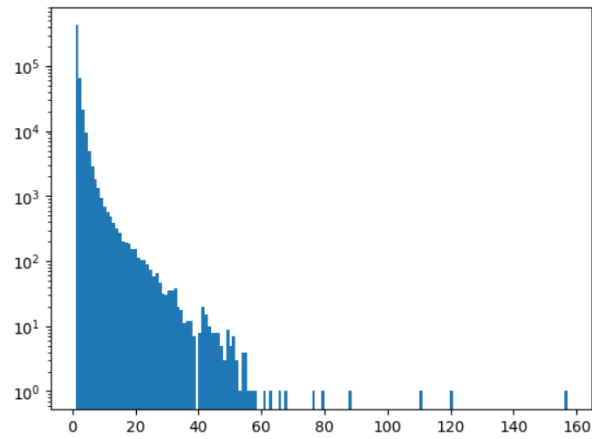


Fig 4.1: Distribution of Question

#### ii. Distribution of Number of words

Generate the same graphical analysis for the average, minimum, or maximum number of words in each question we have done.

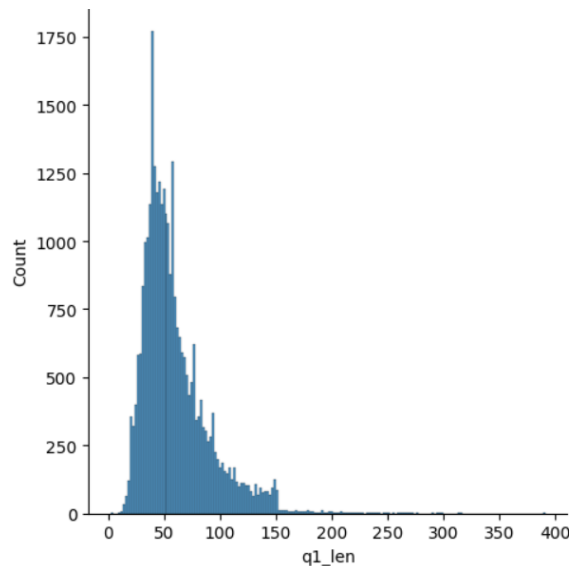


Fig 4.2: Distribution of Number of words

### iii. Common word Analysis

We will plot the distribution of common words in questions 1 and 2. In this distribution plot, we will have a separate curve of non-duplicate and vice-versa

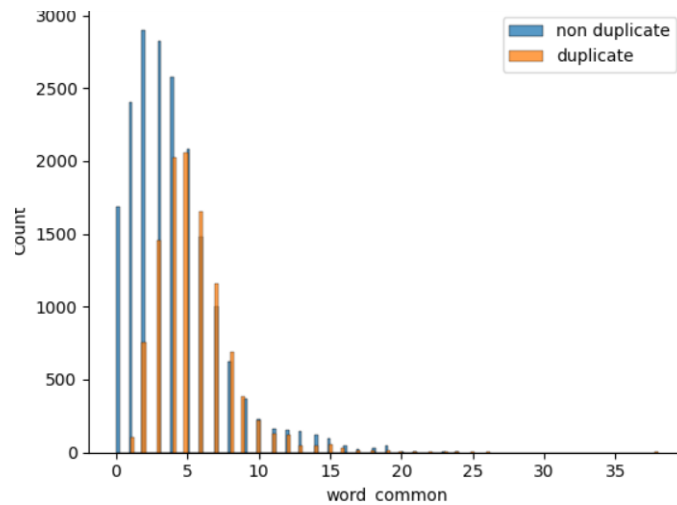


Fig 4.3: Analysis of Common words

### iv. Total word Analysis

The same analysis for total words against target variable unique entries.

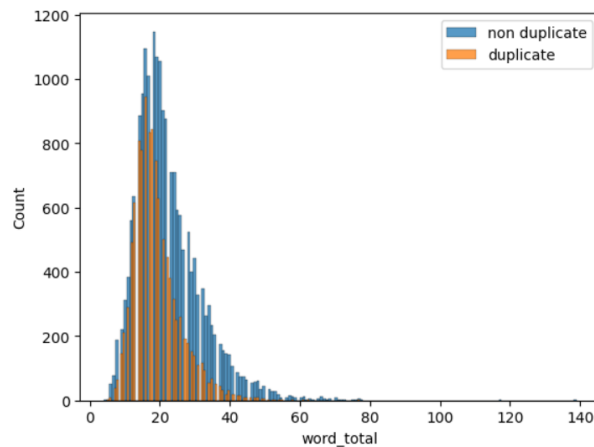
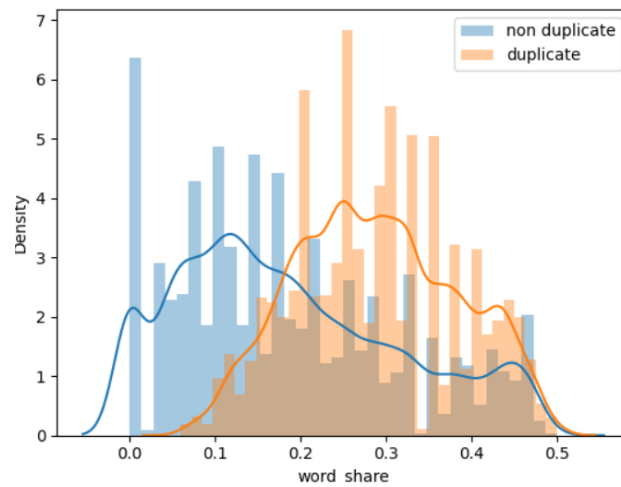


Fig 4.4: Analysis of Total words

Records will be duplicated if the total word count is between 0 and 20, but if it is more significant than 40, the model gives high weightage to non-duplicates.

v. **Word share Analysis**

Plot the distribution plot for duplicate and non-duplicate against the words share column.



**Fig 4.5: Analysis of Words Share**

A non-duplicate is likely to occur if the words share value is less than 0.2, but duplication occurs if the word's share value is more significant than 0.2.

## **4.2 Machine Learning Model Creation**

### **4.2.1 Introduction**

The selection and optimization of machine learning models are crucial steps in achieving accurate and efficient duplicate question pair identification. Choosing appropriate algorithms, such as Random Forest or XGBoost, depends on factors like dataset characteristics, task complexity, and desired performance metrics. Optimization involves fine-tuning model parameters, conducting hyperparameter tuning, and implementing feature engineering techniques to enhance model robustness and generalization capabilities. By optimizing machine learning models, researchers and practitioners can achieve higher accuracy and reliability in question pair similarity detection, leading to improved user experience and content quality in knowledge-sharing communities.

After performing the above EDA, we gain the confidence to keep the features in our dataset and move to the modelling part.

### **4.2.2 Separate the Independent and Dependent features**

First, we need to drop the unnecessary columns and pick the columns needed for training and one target feature (dependent). So, we will pick questions in different data frame for vectorizing and other features in another data frame with the target variable. And concat them after vectorizing.

### **4.2.3 Vectorizing the Feature**

We need to turn the questions into numerical ones because we can't provide the string to the model. To do this, we employ a variety of feature vectorizing techniques; for the moment, we'll use a bag of words (BOW). The bow is a technique for extracting characteristics from text input for machine learning algorithms. It displays the text that describes words' behavior in the corpus, which entails two things. The first is words' vocabulary (unique words in the corpus added as a new feature), and the second is a way to count the number of known words (represent the word's presence in that query using binary format).

### **4.2.4 Train – Test Split**

For calculating the performance of the model, we need some amount of data that the model has not seen as a test set, so we will split the final data frame into two parts, training and test set, where 80 percent of data in the train set and 20 percent for the test set.

#### 4.2.5 Train the Machine Learning Models

To determine which model works the best, we will train XGboost, and Random Forest both models.

### 4.3 Baseline Model Performance

TABLE 4.1: Initial Performance

Model Type	Mode Detail	Accuracy (%)	Recall (%)
Classifier	Random Forest	74	73
	XGBoost	73	72

### 4.4 Optimizing the Current Model Performance

The main work now is increasing the model performance and getting a well-generalized model. therefore far, we have not performed any text preprocessing, so from now we will perform text mining and analyze the text in different ways to generate some advanced features, which we can say is Advance feature engineering. The feature I have found through multiple notebooks available on Kaggle is that the features are represented as Magic. So, for this, we will again load the data in a new data frame because we will clean the text and calculate all features again.

#### 4.4.1 Text Preprocessing

The first step is to clean up the text and rectify the dataset by removing the irregularities in regular NLP Projects. As a result, we will do the following text-cleaning processes.

- i. **Lowercase:** If the text falls into one case, it is simple to vectorize and interpret because the vectorizer considers token and Token to be different words. So, we will convert the entire text into lowercase.
- ii. **String Equivalents:** The text contains multiple symbols, so we will replace them with corresponding string words.
- iii. **Expand Contraction:** Contraction is written communication in human language to write words in short form. For example, don't stand for do not so there are multiple contractions which we need to change to corresponding complete forms.
- iv. **Remove HTML tags:** The text contains some unnecessary HTML tags, so that we will remove them.

- v. **Remove Punctuation:** Punctuation is unnecessary and does not convey meaning, so it is better to remove them.

## 4.5 Advance Feature Engineering

In the domain of identifying duplicate question pairs on Quora, advanced feature engineering plays a pivotal role in augmenting the performance of machine learning models. By employing sophisticated techniques tailored to the characteristics of textual data, we can extract nuanced features that effectively capture the subtle nuances and distinctions between questions. Advanced feature engineering for duplicate question pair identification involves leveraging techniques such as text embeddings, where words are represented as dense vector representations using pre-trained models like Word2Vec [3] or GloVe [2]. These embeddings encode semantic relationships between words and can be aggregated to represent entire questions in a meaningful vector space.

### A. Token Based Feature

Tokens are chunks of words that unite to form a sentence. A single word is a Token so we will add some of the features at the token level or that will help to calculate some valuable columns.

- i. **Cwc min** – It represents the number of common words with a minimum number of words in more minor questions.
- ii. **Cwc max** - represents the number of common words to a maximum number of words in a larger question.
- iii. **Csc min** – It represents the number of common stop words to the smaller stop word count among two questions.
- iv. **Csc max** - represents the number of common stop words to the larger count among two questions.
- v. **Ctc min** – The number of standard tokens to count more minor tickets among two questions.
- vi. **Ctc max** – The number of standard tokens to count more significant tokens among two questions.
- vii. **Last word equal** – The binary feature includes the value 1 if the last word of both questions is the same, else 0.
- viii. **The First equal** - If the first word of both questions is equal then the value is 1. Else 0.

## B. Length Based Feature

- i. **Mean Length** – It is an essential but crucial feature where we will find the average of both questions' length, known as the mean length of both questions. For example, if the first question is 8 characters in length and the second is 6 characters in length, then the average becomes 14 divided by 2 equals 7, so the mean is 7.
- ii. **Absolute Length Difference** - The absolute difference between the length of the two questions (length of words)
- iii. **Longest Substring Ratio** - The ratio of the length of the longest substring between two questions is divided by the length of the smaller questions. The first thing is you need to find the substring in both questions and determine the longest one and then divide it with a length of a small token sentence.

## C. Fuzzy Features

These are some features that are generated using the fuzzy-wuzzy library. To get important about this feature, you can recommend this blog written by the founders of the library itself, where in detail, they explain how these features are calculated.

### 4.5.1 EDA and Newly Created features

We have added nearly 23 new features to our dataset, so before modeling, we want to be confident that these features would dominate the prediction of output variables, so we run some analysis to identify specific patterns.

#### i. Minimum Variables with Target Variable

The feature we have added as a minimum calculation against the target feature to see how they affect record duplication.

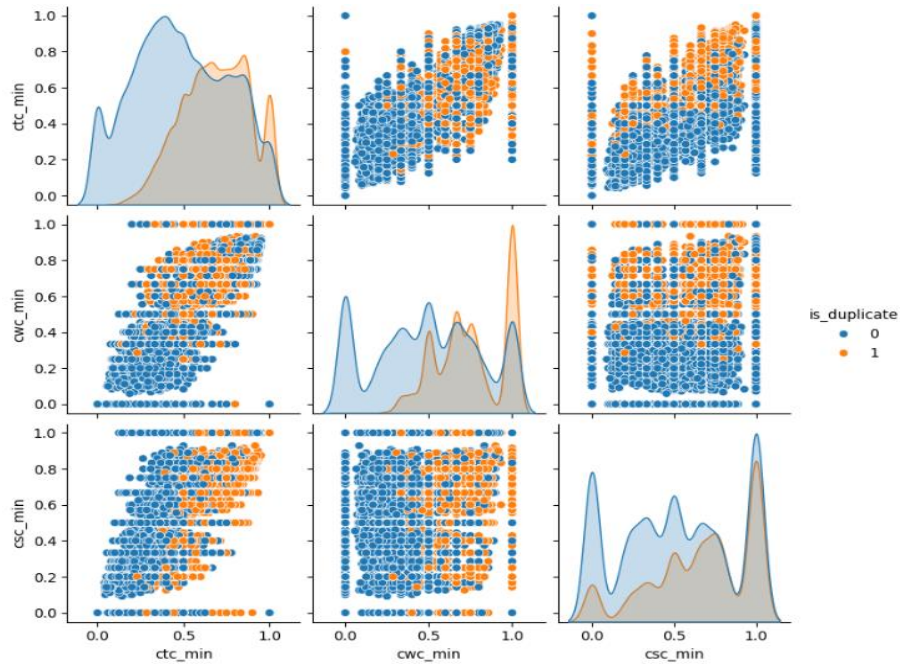


Fig 4.6: Analysis of Newly added features



ii. **Maximum Variable with Target Variable**

We have also appended the maximum calculation, so let us plot them against target variables.

iii. **Last Word and First Word Analysis**

We plot the first and last word match against the target variable.

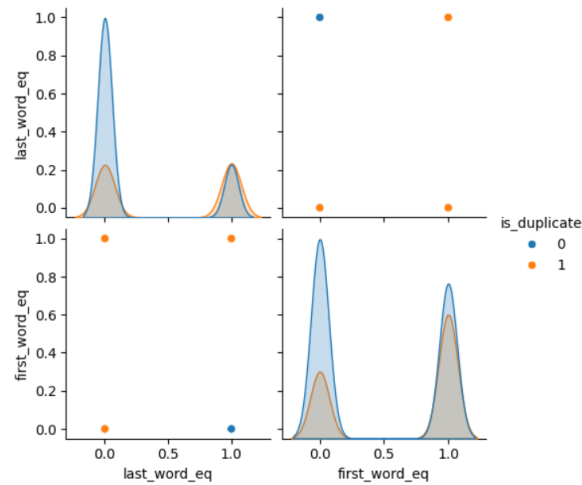


Fig 4.7: Analysis of first word last word

iv. **Length-based Feature Analysis**

The mean length and the absolute curve do not give much information because both the curves are moving almost together, but the most extended substring feature is beneficial where a blue curve is dominating.

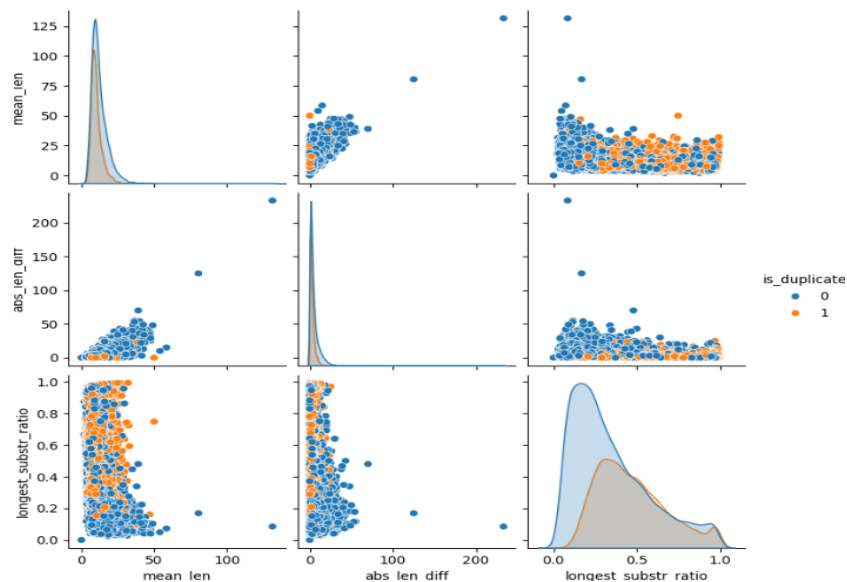
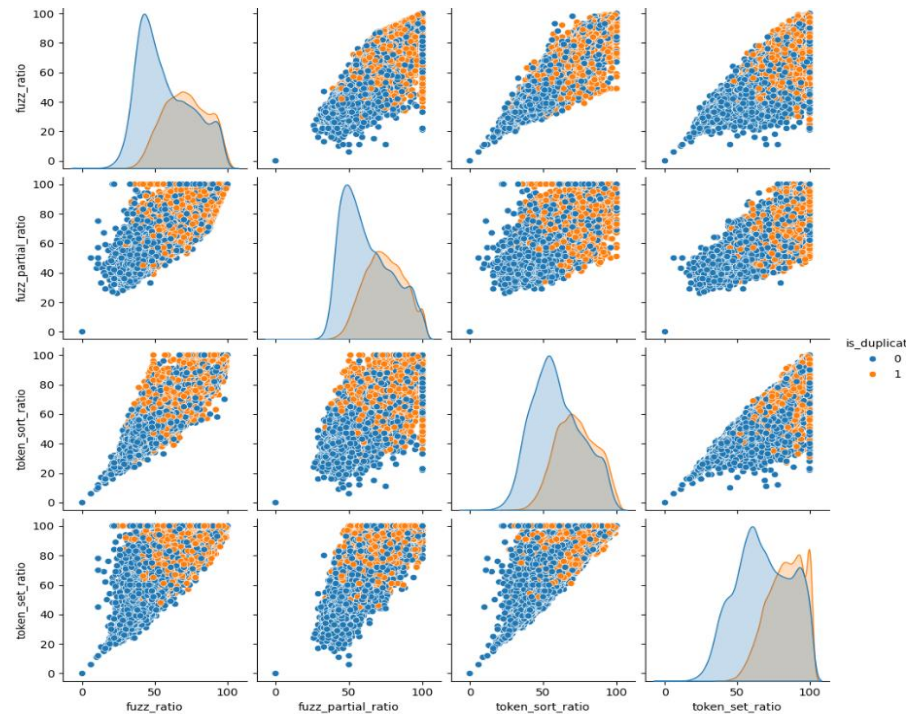


Fig 4.8: Analysis of Length Based features

**v. Fuzzy Feature Analysis**

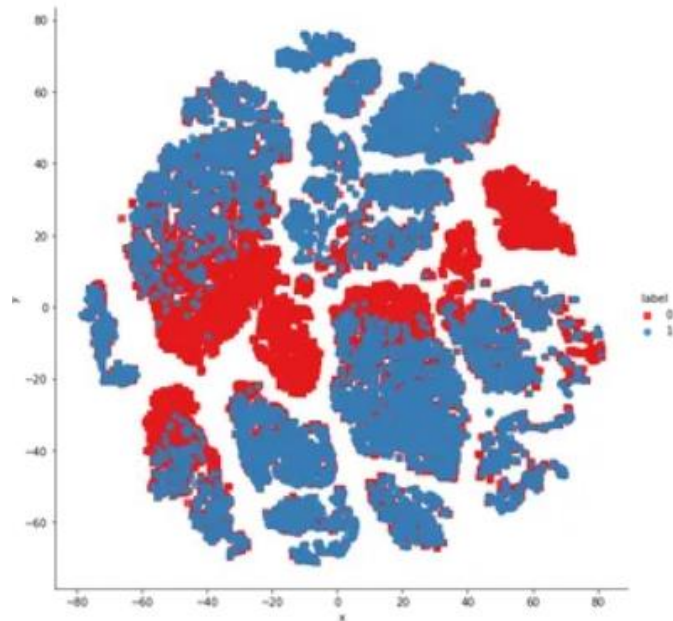
All 4 features give a good understanding of the output variables, which can be useful.



**Fig 4.9: Analysis of Fuzzy Features**

## 4.6 Dimensionality Reduction

We will employ TSNE (T-distributed stochastic neighbor Embedding) [23], a non-linear unsupervised dimensionality reduction technique for data exploration and visualization. First, we'll plot the data on the 2D graph. Then, we'll use it to visualize the data in 3-D, allowing you to see the impact of 15 features on the target variable. Visit the notebook to get the code for the plot 3-D view.



**Fig 4.10: TSNE for Data Exploration**

The data is now ready, and you must repeat the steps above to train the Random Forest and XGboost models for NLP Project. You can rerun the cell or copy and paste the code. Only the difference here is we have 15 more features which total become 6023 by adding 15 more features. The random forest accuracy is approximately 78.7, and XGboost gives 80 percent. So, by doing this much optimization, we could boost the performance by 2 to 2.5 percent. We believe that the performance will only increase slowly after the model gets the main explanation.

# Chapter 5: Results

## 5.1 Observations

The data is now ready, and you must repeat the steps above to train the Random Forest and XGboost models for NLP Project. You can rerun the cell or copy and paste the code. Only the difference here is we have 15 more features which total become 6023 by adding 15 more features. The random forest accuracy is approximately 78.7, and XGboost gives 80 percent. So, by doing this much optimization, we could boost the performance by 2 to 2.5 percent. We believe that the performance will only increase slowly after the model gets the main explanation.

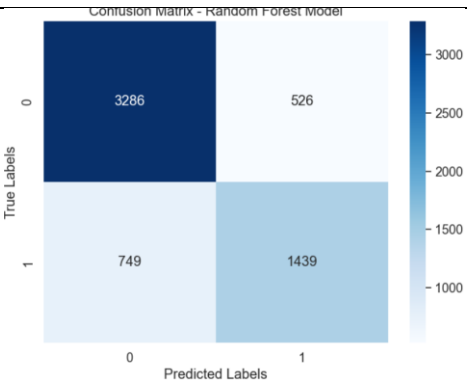
### A. Selecting the best model

This is where many engineers and practitioners make the mistake of picking the best model for deployment. So, we have two underlined scenarios to consider while selecting between the models.

### B. Findings

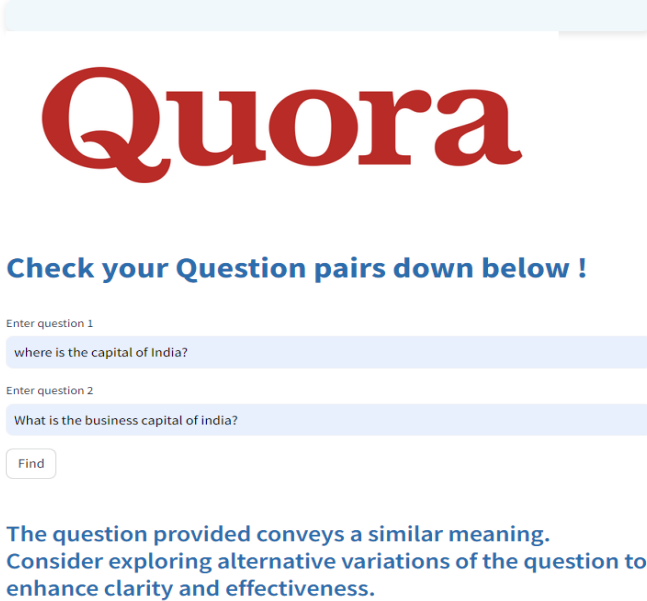
- i. The real value is non-duplicate, but the model reports it as such.
- ii. When the actual value is duplicated, the model predicts it is not.

Table 5.1 RESULTS ANALYSIS

Model Type	Model Detail	Initial (%)	F1 Score After Adding 7 new Features (%)	F1 Score After Adding 15 new Feature (%)	Confusion Matrix
1.	Random Forest	74	77	78	

2.	XGboost	73	76	80	<div><p>Confusion Matrix - XGBoost Model</p><p>A confusion matrix heatmap for the XGBoost model. The y-axis is labeled 'True Labels' with values 0 and 1. The x-axis is labeled 'Predicted Labels' with values 0 and 1. The matrix shows 3252 true positives (0,0), 560 false positives (0,1), 683 false negatives (1,0), and 1505 true negatives (1,1). A color bar on the right indicates counts from 1000 to 3000.</p><table><thead><tr><th></th><th>0</th><th>1</th></tr></thead><tbody><tr><th>0</th><td>3252</td><td>560</td></tr><tr><th>1</th><td>683</td><td>1505</td></tr></tbody></table></div>		0	1	0	3252	560	1	683	1505
	0	1												
0	3252	560												
1	683	1505												

## 5.2 Application Screenshots



**Quora**

**Check your Question pairs down below !**

Enter question 1  
where is the capital of India?

Enter question 2  
What is the business capital of india?

Find

The question provided conveys a similar meaning.  
Consider exploring alternative variations of the question to enhance clarity and effectiveness.

**Fig 5.1: Streamlit Front-End**

# Quora

**Check your Question pairs down below !**

Enter question 1

where is the capital of India?

Enter question 2

What is the business capital of Canada?

Find

**Not Duplicate**

**Fig 5.2: Not Duplicate Found page**

# Conclusion and Future Scope

---

## Conclusion

In this project, we successfully developed an end-to-end NLP application for identifying question pairs on Quora. This journey took us through the comprehensive lifecycle of an NLP project, from initial data analysis to the deployment of a cloud-based application. By employing advanced machine learning models, specifically Random Forest and XGBoost, we explored the intricacies of text preprocessing, feature engineering, and model optimization. Despite XGBoost yielding higher accuracy, the Random Forest model was ultimately chosen for its reliability and consistency in results.

### Key achievements of this project include:

**Understanding the NLP Lifecycle:** We navigated through the various stages of an NLP project, gaining insight into both theoretical and practical aspects.

**Advanced Feature Engineering:** We identified and engineered 23 new features, significantly improving the model's performance from 74% to approximately 80% accuracy.

**Machine Learning Model Optimization:** Through meticulous feature engineering and model tuning, we optimized our models to deliver robust performance.

**Data Visualization and EDA:** We conducted extensive exploratory data analysis (EDA) to uncover hidden patterns and relationships within the text data.

**Deployment:** We implemented and deployed our application using Streamlit, making our solution accessible and scalable via cloud deployment.

This project has provided a deep understanding of NLP applications in real-world scenarios, equipping us with valuable skills in machine learning, feature engineering, and model deployment.

## Future Scope

The success of this project opens several avenues for future work and enhancement:

**Model Enhancements:** Further improvements can be made by experimenting with other sophisticated models such as BERT, GPT, or Transformer models that have shown remarkable performance in NLP tasks.

**Real-time Processing:** Enhancing the system to process and analyze data in real-time can make the application more responsive and useful for live question pair identification.

**Cross-domain Applications:** The techniques and models developed in this project can be adapted and applied to other domains where question or text similarity is crucial, such as customer support, content recommendation systems, and academic research.

**Multilingual Support:** Expanding the application to support multiple languages can broaden its usability and relevance in a global context.

**User Feedback Loop:** Integrating a feedback mechanism where users can report incorrect predictions can help in continuously improving the model's accuracy.

**Scalability and Performance Optimization:** Further optimizing the deployment pipeline to handle larger datasets and higher traffic volumes can ensure the application remains efficient and scalable.

**Automated Feature Engineering:** Developing automated feature engineering tools using techniques such as AutoML can streamline the process and potentially discover novel features that enhance model performance.

Overall, the project demonstrates the power and potential of NLP in solving complex text analysis problems, laying a solid foundation for further innovations and improvements in the field.



# References

---

- [1] <https://www.lexalytics.com/news/lexalytics-unveils-Duplicate-Question-pair-identification-analysis-of-emoticons-acronyms/>
- [2] <https://textblob.readthedocs.io/en/dev/classifiers.html>
- [3] <https://stackoverflow.com/questions/48860422/text-blob-naive-bayes-classification>
- [4] <https://chat.openai.com/>
- [5] <https://www.kaggle.com/code/anoopjohny/threads-dataset-analysis/input>
- [6] Achananuparp, Palakorn, et al. "Utilizing sentence similarity and question type similarity to respond to similar questions in knowledge-sharing community." Proceedings of QAWeb 2008 Workshop. Vol. 214. 2008.
- [7] Mueller, Jonas, and Aditya Thyagarajan. "Siamese Recurrent Architectures for Learning Sentence Similarity." AAAI. 2016.
- [8] Mikolov, Tomas, et al. "Efficient estimation of word representations in vector space." arXiv preprint arXiv:1301.3781 (2013).
- [9] Dey, Kuntal, Ritvik Shrivastava, and Saroj Kaushik. "A paraphrase and semantic similarity detection system for user generated short-text content on microblogs." Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers. 2016.
- [10] Bird, Steven, and Edward Loper. "NLTK: the natural language toolkit." Proceedings of the ACL 2004 on Interactive poster and demonstration sessions. Association for Computational Linguistics, 2004.
- [11] Achananuparp, Palakorn, Xiaohua Hu, and Xiajiong Shen. "The evaluation of sentence similarity measures." International Conference on data warehousing and knowledge discovery. Springer, Berlin, Heidelberg, 2008.
- [12] Sanborn, Adrian, and Jacek Skryzalin. "Deep learning for semantic similarity." CS224d: Deep Learning for Natural Language Processing. Stanford, CA, USA: Stanford University (2015).
- [13] Quora Question Pairs | Kaggle [Online] Available: <https://www.kaggle.com/c/quora-question-pairs>
- [14] Langville, Amy N., and Carl D. Meyer. Google's PageRank and beyond: The science of search engine rankings. Princeton University Press, 2011.
- [15] Home - Quora [Online] Available: <https://www.quora.com/>
- [16] reddit: the front page of the internet [Online] Available: <https://www.reddit.com/>
- [17] Kaggle: Your Home for Data Science [Online] Available: <https://www.kaggle.com/>
- [18] Ho, Tin Kam. "Random decision forests." Document analysis and recognition, 1995., proceedings of the third international conference on. Vol. 1. IEEE, 1995.
- [19] Chen, Tianqi, and Carlos Guestrin. "Xgboost: A scalable tree boosting system." Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. ACM, 2016.
- [20] Wieting, John, et al. "Learning Paraphrases with Siamese Recurrent Networks." Proceedings
- [21] Stackoverflow: Every developers Home [Online] Available: <https://stackoverflow.com/>