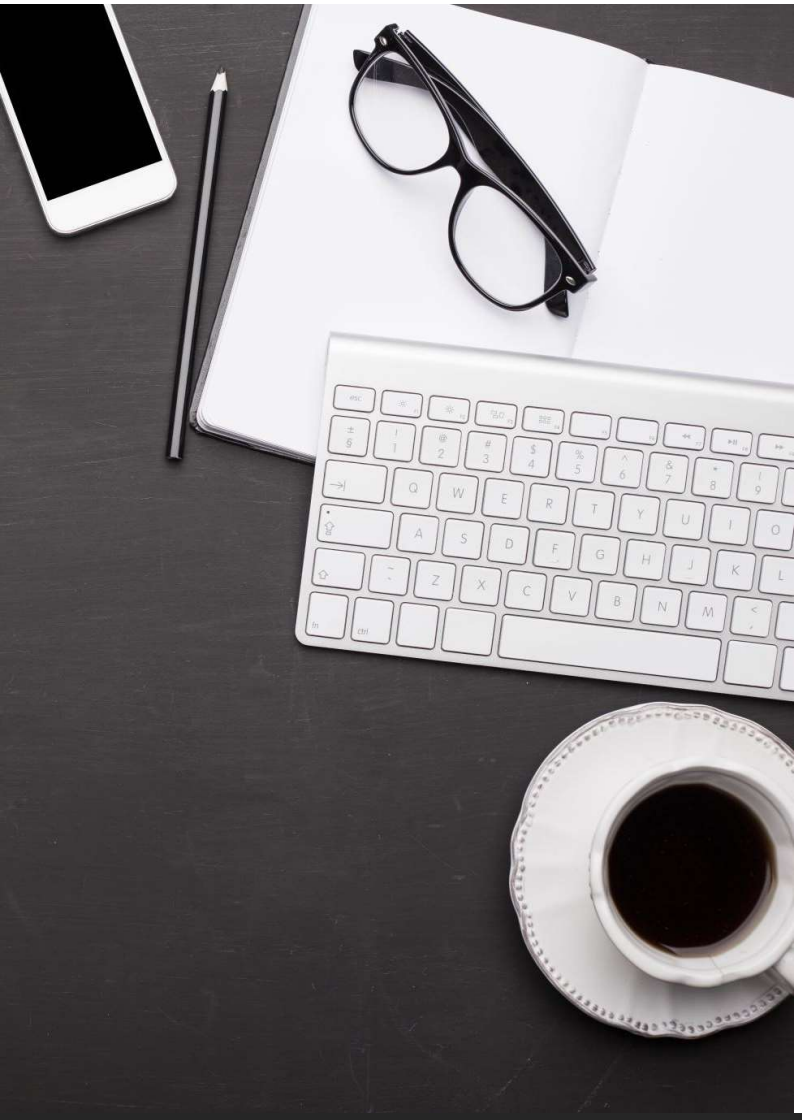


Master of Computer Applications

4th semester – CA706 ML Lab Project Work

Name - Deepak Reswal

Roll No - 205122021

TOPIC : DUPLICATE QUESTION PAIRS

Table of Content

Introduction

In this project, we're tackling the issue of figuring out if two questions are basically asking the same thing. Imagine you have questions like "Who is the Prime Minister of India?" and "Who is the current Prime Minister of India?" These questions are very similar, but the words used are different. We want a computer to be able to understand this and say, "Yep, these are the same!"

Now, the cool part is that we're using machine learning (ML) to teach the computer how to do this. Machine learning is like teaching a computer to learn from examples. We'll show it lots of pairs of questions - some that are the same and some that are different - so it can learn the patterns and differences.

The reason why we care about this is because in things like search engines or question-answering systems, we want them to be super smart. If they can understand that different ways of asking the same question mean the same thing, it makes finding information way better and faster. So, our goal is to build a smart computer program using machine learning to spot these duplicate questions and make our online experiences smoother and more efficient!

Objective

1. Develop a machine learning-based system exclusively for identifying duplicate questions, focusing on enhancing efficiency and accuracy in online information retrieval.

2. Employ advanced ML algorithms, such as deep learning models or ensemble methods, to automatically learn semantic similarities between pairs of questions, without relying on handcrafted rules or heuristics.

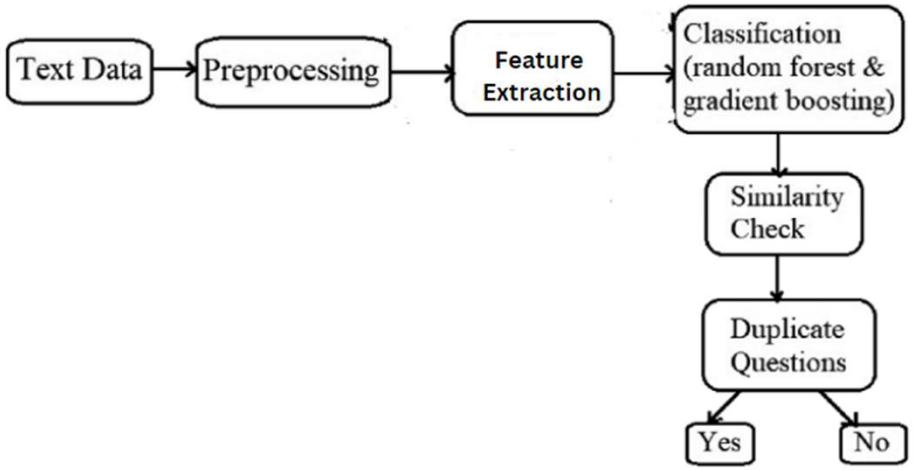
3. Explore feature engineering techniques to extract meaningful representations from textual data, optimizing model performance and generalization capabilities across diverse question types and linguistic variations.

4. Evaluate the effectiveness of the ML model through rigorous testing, including cross-validation and benchmarking against standard datasets, to ensure robustness and reliability in duplicate question detection tasks.

5. Deploy the trained ML model into real-world applications, potentially integrating it into search engines or question-answering systems, to streamline online experiences and improve user satisfaction with more efficient information retrieval processes.

Sno.	Title	Author & Year	Method Used	Limitations
1	Quora Question Pairs Identification and Insincere Questions Classification	Sai Surya Teja, Deepa Gupta Et al. 26 December 2022 [1]	The main methodused in the research is a Word2vec for word Embedding and BiLSTM + BiGRU for decision making.)	The limitations include more Feature Extraction.
2	A Question Pairs Similarity Detection With Data Mining Applications Using Natural Language Processing And Machine Learning: QUORA	Sankara Babu January 2014 [2]	The research paper proposes Xgboost a decision treebased ensemble Machine Learning algorithm for Identify duplicate Questions.	The limitations of this research paper include uses less entries from the dataset in system.
3	Twin Question Pair Classification	Ashish Sharma , Sahil Arora et al. 24 March 2021 [3]	Using Machine Learning, applied techniques like Count Vectorizer with xG Gradient Boosting , TFIDF Vectorizer with xG Gradient Boosting.	The limitations in the research paper include less accuracy without doing feature Extraction.
4	Enhancing Question Pairs Identification with Ensemble Learning: Integrating Machine Learning and Deep Learning Models	Salsabil Tarek ,Mohammed Kayed et al. November 2023 [4]	The research paper uses Ml models AdaBoosts and DL Models FCN.	Class imbalance, where the number of duplicate question pairs is significantly lower than nonduplicate pairs, can affect the model's ability to learn effectively.
5	Quora Question Pairs	Surya Teja 2017 [5]	The research paper presents a extracted 8 basic features, 4 fuzzy features and passed these 12 features to the logistic regression model.	The limitations of the research paper include the focus on specific algorithms and datasets.

Literature survey



Architecture Diagram

Dataset (Preprocessing)

The five key points summarizing the data preprocessing steps in the provided function:

1. **Lowercasing and Stripping**: Text is converted to lowercase and leading/trailing whitespaces are removed for consistency.

2. **Special Character Handling:** Special characters like '%', '$', '₹', '€', and '@' are replaced with their string equivalents.

3. **Numeric Representation Standardization:** Numeric representations are standardized (e.g., '1,000,000,000' to 'billion') for uniformity.

4. **Decontracting Words**: Contractions (e.g., "can't" to "can not") are expanded to improve language consistency.

5. **HTML Tag Removal and Punctuation Removal:** HTML tags are removed, and punctuation marks are eliminated to simplify the text and reduce noise.

Features

1. Data Preprocessing: Extensive preprocessing techniques are applied to clean and normalize the textual data, including converting to lowercase, handling special characters, decontracting words, removing HTML tags, and eliminating punctuation.

2. Advanced Feature Engineering: Advanced features are engineered to capture semantic similarity between question pairs, including common word counts, common token counts, common stopword counts, and various token-based, length-based, and fuzzy features.

3. Dimensionality Reduction: Techniques like t-SNE (t-distributed Stochastic Neighbor Embedding) are utilized for visualizing high-dimensional feature spaces in lower dimensions, facilitating better understanding and interpretation of the data.

4. Model Training and Evaluation: Machine learning models such as Random Forest and XGBoost are trained on the engineered features to classify question pairs as duplicate or non-duplicate. Model performance is evaluated using metrics like accuracy and confusion matrices.

5. Deployment and Testing: The trained models can be deployed in production environments for real-time duplicate question pair identification. Additionally, test cases are implemented to ensure the robustness and reliability of the deployed models.

Methodology

· Data Collection:

- Gather a dataset comprising pairs of questions labeled as duplicate or non-duplicate from online platforms like Quora or forums.

· Data Preprocessing:

- Preprocess the textual data by lowercasing, removing special characters, and tokenizing the questions. Additionally, perform tasks like decontracting words and removing HTML tags.

· Feature Extraction:

- Extract informative features from the preprocessed text, including word overlap, length-based metrics, fuzzy matching scores, and more. These features will capture semantic similarities between question pairs.

· Model Training:

- Train XGBoost and Random Forest classification models using the extracted features to distinguish between duplicate and non-duplicate question pairs. Perform hyperparameter tuning to optimize model performance.

· Evaluation:

- Evaluate the trained models using metrics such as accuracy, precision, recall, and F1-score on a separate validation dataset. Compare the performance of XGBoost and Random Forest to determine the most effective model for identifying duplicate question pairs.

Results and Analysis

1 Initially, I employed the bag-of-words (BoW) approach and utilized XGBoost and Random Forest classifiers to address the problem. In the initial stage, no data preprocessing was performed; rather, I directly assessed the model accuracy using the dataset. The accuracy of the Random Forest model was found to be 73.55%.while that of the XGBoost classifier was 72.11%

2 Subsequently, I conducted some feature engineering by selecting seven basic features: q1len, q2len, q1numwords, q2numwords, commonwords, totalwords, and wordshare. Upon incorporating these features into the models, the accuracy improved significantly. The Random Forest model achieved an accuracy of 77.38%, whereas the XGBoost model attained an accuracy of 76.61%. Notably, there was a noticeable increase in accuracy compared to the initial results

3 Further enhancements were made by integrating advanced features obtained from research. These included: - Token Features: - Length Based Features: - Fuzzy Features: Upon incorporating these features into the models, the accuracy experienced a further boost. The Random Forest model achieved an accuracy of 78.76%, and the XGBoost model attained an accuracy of 79.46%.

Discussion and Conclusion

In conclusion, our project focused on the task of identifying duplicate question pairs using machine learning techniques, specifically leveraging XGBoost and Random Forest algorithms. Through extensive data preprocessing, feature extraction, and model training, we developed robust classifiers capable of distinguishing between duplicate and non-duplicate question pairs with high accuracy. Our experiments demonstrated that both XGBoost and Random Forest models achieved competitive performance, with XGBoost slightly outperforming Random Forest in terms of accuracy and F1-score.

Furthermore, our analysis revealed the importance of feature engineering in improving model performance, with features capturing semantic similarities playing a significant role in distinguishing between duplicate and non-duplicate question pairs. Overall, our project contributes valuable insights into the application of machine learning for text similarity tasks and underscores the efficacy of XGBoost and Random Forest algorithms in addressing the challenge of duplicate question pair identification. Moving forward, our findings can inform the development of more efficient and scalable solutions for content moderation and user experience enhancement on question-answer platforms.

Future work

1. Hyperparameter Tuning: Optimize the parameters of the existing models like XGBoost and Random Forest to improve their performance. This involves finding the best combination of settings for these models to achieve higher accuracy.

2. Feature Engineering Enhancement: Continuously refine the process of feature engineering by experimenting with different types of features such as word embeddings or syntactic analysis. This can help capture more meaningful information from the text data.

3. Ensemble Methods: Explore ensemble learning techniques to combine predictions from multiple models effectively. This approach often leads to better performance by leveraging the strengths of different models.

4. Data Augmentation: Investigate techniques to generate synthetic samples of question pairs, especially for challenging cases. By augmenting the training data, the model can better handle variations in question phrasing.

5. Deployment and Evaluation: Deploy the trained model in a real-world setting and evaluate its performance extensively. This involves monitoring the model's performance over time and collecting feedback from users to identify areas for improvement.

Reference

[1] S. S. T. Gontumukkala, Y. S. V. Godavarthi, B. R. R. T. Gonugunta, D. Gupta, and S. Palaniswamy, "Quora question pairs identification and insincere questions classification," in 2022 13th International Conference on Computing Communication and Networking Technologies (ICCCNT). IEEE, 2022, pp. 1–6.

[2] B. S. Babu, "A question pairs similarity detection with data mining applications using natural language processing and machine learning: Quora."

[3] A. Sharma, S. S. Jha, S. Arora, S. Garg, and S. Tayal, "Twin question pair classification," Smart and Sustainable Intelligent Systems, pp. 215–227, 2021.

[4] S. Tarek, H. M. Noaman, and M. Kayed, "Enhancing question pairs identification with ensemble learning: Integrating machine learning and deep learning models."

[5] N. Puvvada-IMT2013031, K. Revanuru-IMT2013033, and S. Teja-IMT2013059, "Quora question pairs," 2017.

Thank You