Exploring Machine Learning Models for Duplicate Question Detection in Online Communities

[1],[2]
[1,2]*Department of Computer Science and Engineering (AIE)*
Amrita School of Computing, Amrita Vishwa Vidyapeetham, Chennai-601103, India
@gmail.com[1], @gmail.com[2]

***Abstract*—Duplicate question detection is a crucial task in question-and-answer platforms, aiming to improve user experience and minimize redundancy. This project utilizes NLP techniques and machine learning models to accurately predict whether a pair of questions are duplicates.The project involves comprehensive data preprocessing, including tokenization, stopword removal, and lemmatization, to prepare the textual data for analysis. Various feature engineering techniques, such as bag-of-words, TF-IDF, and word embeddings, are applied to extract meaningful representations of the questions. Multiple machine learning models, including Logistic Regression, Random Forest, SVM, XGBoost, and a Combined model, are trained on labeled data. Evaluation metrics focus on precision and recall to minimize misclassification costs. New sample data is used to validate the models. Predictions are made to classify question pairs as duplicates or non-duplicates, showcasing the models' generalization capabilities. The results highlight the potential of NLP techniques and machine learning models in accurately identifying duplicate question pairs, improving knowledge sharing, and reducing redundancy. This project provides valuable insights and methodologies for future research in duplicate question detection, contributing to the field of NLP and enhancing user experiences on question-and-answer platforms.**

***Index Terms*—Question-and-answer platforms, Natural language processing (NLP), Word embeddings, and Misclassification costs.**

I. Introduction

This issue is related to identifying duplicate questions on the Quora platform. Quora is a popular question-and-answer platform where users can ask questions on various topics and get answers from the community. However, users often ask questions that they have already asked before, resulting in duplicate content and duplicate answers. For example, he has two examples of duplicate questions: "How do I read and find Facebook comments?" and "How do I view all Facebook comments?" The aim of the project is to develop a model that can predict whether pairs of questions are overlapping. By accurately identifying duplicate questions, this model can improve the user experience on Quora by providing immediate answers to questions that have already been answered in the past. This saves you and your community time by avoiding redundant discussions and duplicate content. To achieve this, the project uses his NLP (natural language processing) techniques and machine learning algorithms to analyze and compare the text content of question pairs. The model learns question patterns and similarities to determine if questions are duplicates. The aim of the project is to minimize the cost of misclassification when detecting duplicate pairs of questions. The cost associated with misclassification refers to the negative outcome or impact that occurs when a model incorrectly predicts whether a pair of questions overlaps. In the context of this project, misclassification can occur in two ways:

False Positives: This happens when the model predicts a pair of questions as duplicates, but in reality, they are not duplicates. The cost of false positives can be significant because it leads to incorrect assumptions of duplicate content. Users may be provided with duplicate answers or misinformation, leading to frustration, wasted time, and reduced trust in the platform. Additionally, unnecessary duplication of discussions and answers can clutter the platform and decrease its overall quality.

False Negatives: This occurs when the model predicts a pair of questions as non-duplicates, but they are actually duplicates. False negatives can have adverse consequences as well. Users who ask duplicate questions may not receive immediate answers because the model failed to identify the existing answers. This can lead to duplicated efforts by the community, resulting in redundant discussions, multiple answers to the same question, and inefficiency in knowledge sharing.

To achieve the objective of minimizing the cost of misclassification, the project will focus on developing a model with high precision and recall. High precision ensures that the model avoids false positives and provides accurate predictions. High recall ensures that the model minimizes false negatives and captures as many duplicate question pairs as possible. By optimizing both precision and recall, the project aims to strike a balance between avoiding false positives and false negatives. The study will contribute to improving the user experience on the Quora platform, ensuring accurate and efficient delivery of answers, reducing redundancy, and enhancing the overall quality of knowledge sharing.

II. Related Works

Due to the growing need for efficient question-answering systems on online platforms such as Quora, duplicate question pair detection has received a lot of attention in recent years. To address this issue, several approaches and models have been proposed that aim to improve user experience and reduce redundancy in QA communities.

[1]One commonly used technique in duplicate question pair detection is the utilization of word embeddings. Word embeddings capture semantic information and represent words as dense vectors. In this context, Google news vector embedding, FastText crawl embedding, and FastText crawl subwords embedding have been widely employed for vectorizing questions and training models. Siamese MaLSTM ("Ma" for Manhattan distance) Neural Network [5] is another popular model used for detecting duplicate question pairs. This model compares the similarity between two questions by calculating the Manhattan distance between their respective embeddings. The Siamese architecture enables shared weights and enhances the network's ability to learn and distinguish between duplicate and non-duplicate pairs.

Convolutional Neural Networks (CNNs) [6] have shown promising results in this domain. By leveraging pre-trained word embeddings and Siamese Neural Networks for comparison, the model achieves a high accuracy of 79%. The results surpass alternative methods like Jaccard Similarity and Multilayer Perceptron algorithms. The study demonstrates the effectiveness of CNN in capturing semantic similarities and improving the efficiency of question-and-answer platforms. [2].

In addition to CNNs, other models like Siamese Neural Networks have been utilized for comparison. Siamese Neural Networks [3] measure the similarity or dissimilarity between inputs, providing a valuable mechanism for identifying duplicate questions. To optimize the models, various techniques have been employed, including Stochastic Gradient Descent (SGD).SGD is an efficient optimization algorithm that updates model parameters based on a small subset of training data, thereby improving the overall performance of duplicate question pair detection models. In the field of duplicate question detection, there is a growing need for interpretable deep-learning models that can provide insight into the decision-making process. Deep learning models have shown promise for this task, but there are concerns about the interpretability of these models. In the field of duplicate question detection, there is a growing need for interpretable deep-learning models that can provide insight into the decision-making process. Deep learning models have shown promise for this task, but there are concerns about the interpretability of these models. Attentional mechanisms play an important role in these models, allowing for relevant information and improved interpretability. The text processing phase includes filtering operations to improve the use of pre-trained word embeddings. This step helps capture important information and improve the matching process between words in question pairs.[4].

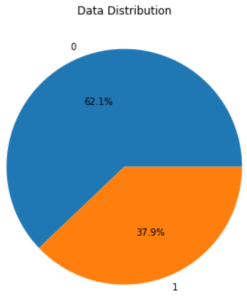
III. Methodology

A. Dataset Description

The dataset used in this research paper is a collection of pairs of questions from the Quora platform, specifically curated for the task of duplicate question pair detection. The goal of this research is to predict whether a given pair of questions have the same meaning or not. The dataset is labeled with ground truth information, which has been provided by human experts. However, it is important to note that the true meaning of sentences is subjective and can never be known with absolute certainty. Human labeling is also prone to noise, and different individuals may have different interpretations or judgments. Therefore, the ground truth labels in this dataset should be considered as 'informed' but not 100% accurate. It is possible that the dataset may contain instances with incorrect labeling or cases where reasonable people could disagree on the labeling.

	id	qid1	qid2	question1	question2	is_duplicate
398782	398782	496695	532029	What is the best marketing automation tool for...	What is the best marketing automation tool for...	1
115086	115086	187729	187730	I am poor but I want to invest. What should I do?	I am quite poor and I want to be very rich. Wh...	0
327711	327711	454161	454162	I am from India and live abroad. I met a guy f...	T.I.E.T to Thapar University to Thapar Univers...	0
367788	367788	498109	491396	Why do so many people in the U.S. hate the sou...	My boyfriend doesnt feel guilty when he hurts ...	0

(a) Screenshot of Dataset



(b) Data Distribution

Fig. 1: Dataset Screenshot and Data Distribution

The (Fig.1(a)) displays a screenshot of the dataset used in the project. It provides a glimpse into the structure and content of the dataset, showcasing the questions and their corresponding labels. The (Fig.1(b)) illustrates the distribution of the dataset used in the project for duplicate question detection.

B. Text Pre-processing

The key is to transform raw text data into a clean, structured format that can be effectively processed by machine learning algorithms. These techniques help improve the performance of NLP tasks such as noise reduction, text normalization, and duplicate question pair detection.

1) Tokenization: Tokenization [7] is the process of breaking text into discrete units called tokens. Depending on the task, these tokens can be words, phrases, or letters.

2) Stop word Removal: Stopwords are commonly used words that have no intrinsic meaning in the context of a sentence or text. Examples of stop words are "that", "is", "and", and "in". Removing stop words removes noise and reduces the dimensionality of the data.

3) Lemmatization / Stemming: Lemmatization and stemming are techniques used to reduce words to their base

(a) Word Cloud plot of duplicate Question pairs

(b) Word Cloud plot of non-duplicate Question pairs

Fig. 2: Word Cloud of Frequent Words in Duplicate and Non-Duplicate Question Pair

or root forms. Lemmatization involves converting words to their base form (lemma) using morphological analysis and dictionary lookups. For example Word: ”running”, Lemma: ”run”. Stemming, on the other hand, is a simpler approach that involves removing prefixes or suffixes from words to obtain their root forms. For example Word: ”running”, Stem: ”run”. Both lemmatization and stemming can aid in reducing the dimensionality and improving the performance of NLP models. The (Fig.2) shown is a word cloud figure generated from the output of the text preprocessing stage. It represents the frequent words found in both duplicate question pairs and non-duplicate question pairs.

C. Feature Engineering

Feature engineering plays an important role in developing effective models for detecting duplicate question pairs. In this section, we consider various subtopics of feature development, including bag-of-word representations, TF-IDF representations, word embeddings, similarity metrics, and feature selection.

1) TF-IDF Representation: TF-IDF representation (Term Frequency-Inverse Document Frequency) is another widely used technique in NLP. Assigns weights to words based on their frequency in the document and rarity across the corpus. The goal is to capture the meaning of the words in the document in relation to the entire dataset.

2) Word Embeddings: Word embeddings capture the semantic meaning of words by representing them as dense vector representations in a continuous vector space. The Word2Vec model learns word embeddings by predicting context words from target words or target words from context words. It creates vector representations where similar words have similar vector representations. For example, the word embeddings for ”king” and ”queen” would be closer to each other than to unrelated words like ”cat” or ”dog.” The (Fig.3) provides insights into the frequency and overlap of common words in the two categories, revealing patterns and differences in their distributions.

3) Similarity Metrics: A similarity metric is used to quantify the similarity between pairs of questions. Cosine similarity measures the cosine of the angle between two vectors and is commonly used in vector representations such as BoW and word embeddings. Computes the dot product between vectors and normalized to account for vector length differences.

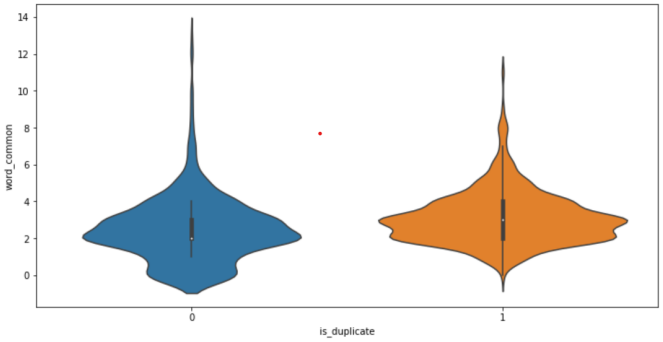


Fig. 3: Violin plot showcasing the distribution of common words between duplicate and non-duplicate question pairs.

D. Model Selection

1) Logistic Regression: Logistic regression is a general linear model for binary classification. It models the relationship between independent variables and the probability of a particular outcome. It can be trained using various optimization algorithms such as gradient descent. Logistic regression is widely used for text classification tasks due to its simplicity and ease of interpretation. [10].

2) Random Forest: Random Forest is an ensemble model that combines multiple decision trees to make predictions[11]. Each decision tree is built on a different subset of the data and features, and the final prediction is obtained through voting or averaging. Random Forest can handle non-linear relationships and capture complex interactions between variables. It is known for its robustness and ability to handle high-dimensional data.

3) Support Vector Machines (SVM): SVM [8] is a powerful supervised learning algorithm used for both classification and regression tasks? The goal is to find the optimal hyperplane that separates data points of different classes with the largest margin. SVM can handle nonlinear relationships by transforming the data into a high-dimensional space using kernel

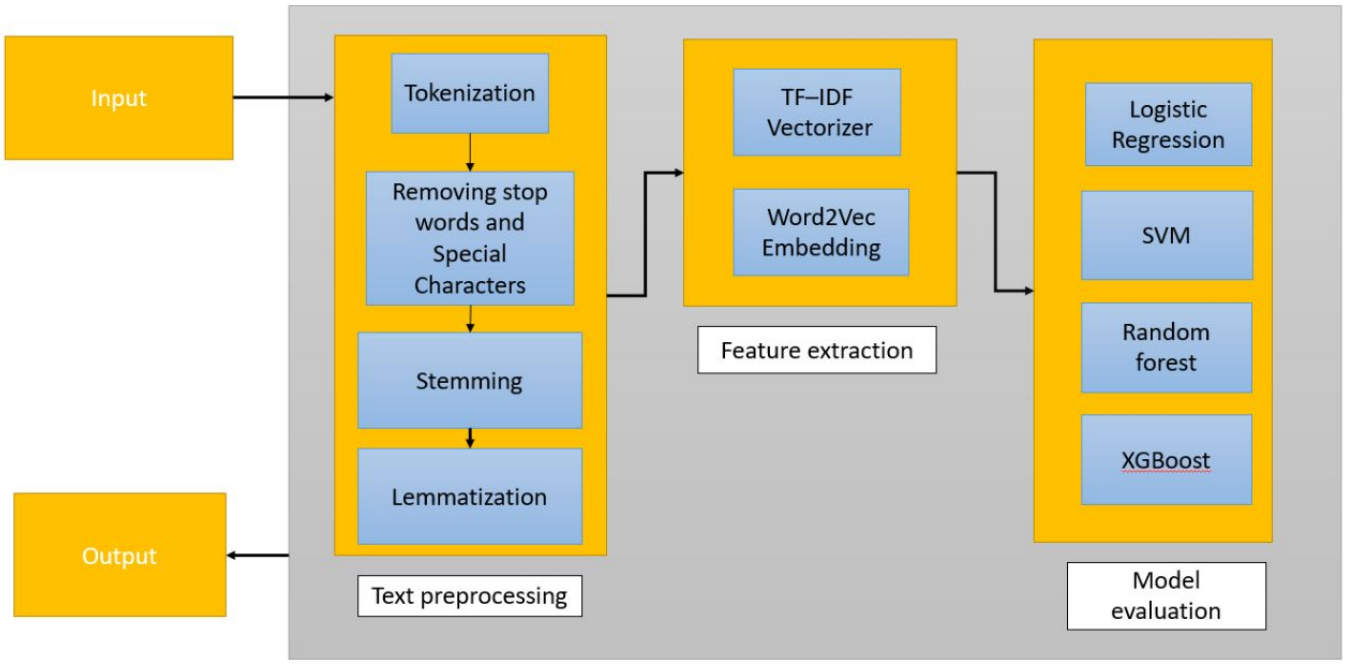


Fig. 4: The architecture of monophonic melody generation.

functions. It is known to handle high-dimensional data and handle smaller data sets well.

4) XGBoost: XGBoost is an optimized gradient boosting algorithm that has gained popularity for its high performance in various machine learning competitions. It uses a combination of weak learners (decision trees) to build a strong predictive model. It offers efficient parallel computing, regularization techniques, and automatic handling of missing values [12].

5) Combined Model: The combined model is an ensemble approach that combines the predictions of multiple individual models to obtain a final prediction. This can be achieved through techniques such as majority voting or weighted averaging. By leveraging the strengths of different models, the combined model aims to improve overall prediction performance and robustness.

E. System architecture

The system architecture and design of the duplicate question detection project include several key components. Data preprocessing phase (Section. III-B) Includes tokenization, stopword removal, and lemmatization to prepare text data. We apply feature extraction techniques such as Bag-of-Words, TF-IDF, and word embedding to capture meaningful representations of question pairs. (Section. III-C). Machine learning models including Logistic Regression, Random Forest, SVM, and XGBoost are utilized for classification. These models are trained using labeled data and evaluated using metrics such as accuracy, precision, recall, and F1 score. The trained models are then used to predict and classify new question pairs as either duplicates or non-duplicates. The system architecture ensures efficient processing of the data and provides accurate predictions to enhance the duplicate question detection process.

IV. Result and Analysis

A. Experimental setup

The experiments are conducted on a specific computational environment with relevant hardware specifications. This includes details such as the type of CPU, GPU, memory, and software dependencies like programming language, machine learning libraries (e.g., sci-kit-learn, TensorFlow, PyTorch), and versions used. This information ensures reproducibility and provides context for the experimental setup.

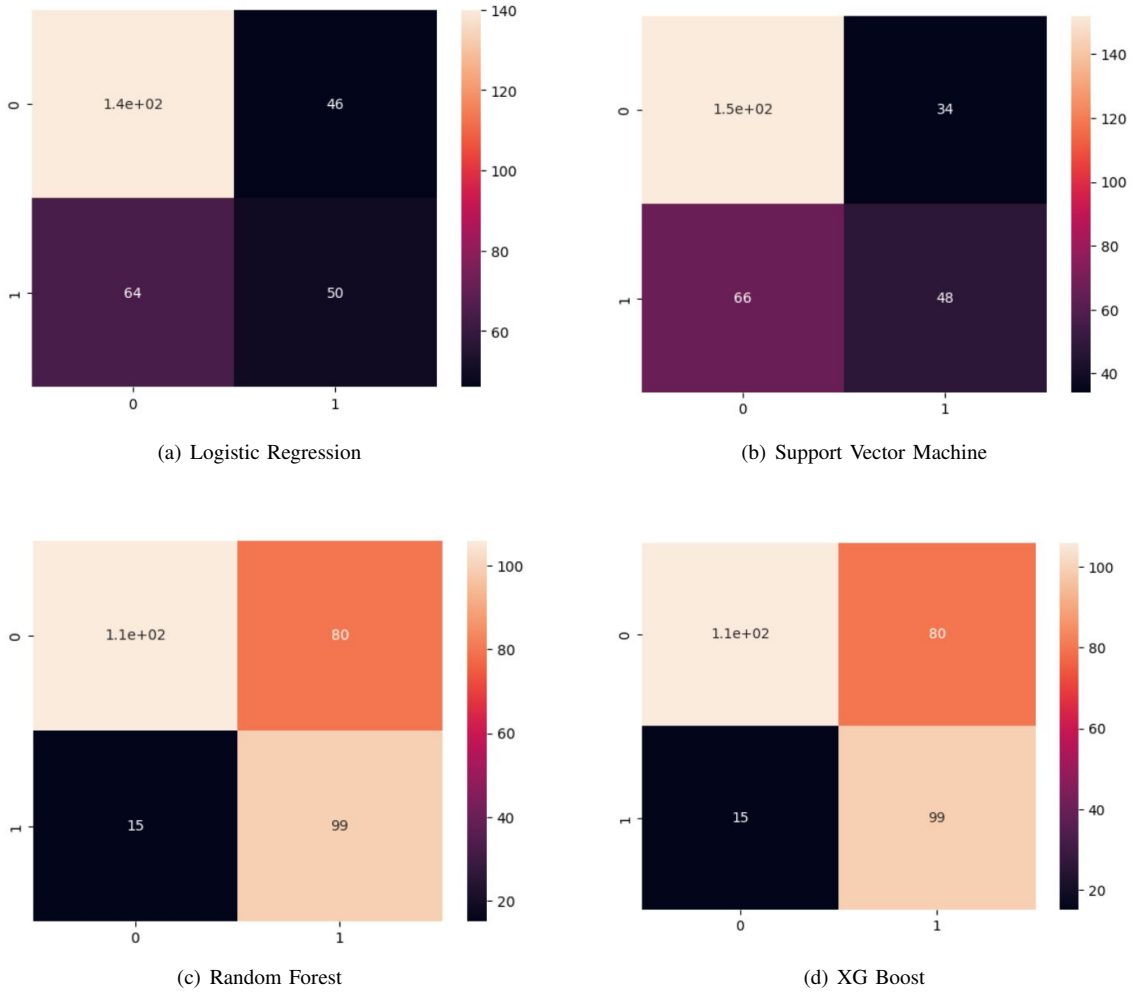
B. Performance Evaluation

The model's performance is evaluated using an appropriate metric to assess its accuracy in predicting pairs of overlapping questions. Common metrics include accuracy, precision, recall, and F1 score. Accuracy measures the overall accuracy of the model's predictions. Accuracy represents the proportion of correctly predicted duplicates out of all predicted duplicates, and recall measures the proportion of correctly predicted duplicates out of all actual duplicates. The F1 score is the harmonic mean of precision and recall, providing a balanced measure of both measures. Evaluation metrics help you compare and select the best-performing models.

C. Experimental Result

D. Model Performance Comparision

In this section, we compare the performance of different models on the duplicate question detection task. The



(a) Logistic Regression

(b) Support Vector Machine

(c) Random Forest

(d) XG Boost

Fig. 5: Confusion Matrix of the Models.

models evaluated are Random Forest, Combined Models, XGBoost, Support Vector Machines (SVM), and Logistic Regression. The metrics used to evaluate the model are accuracy and F1 score. The results of the model performance comparison are presented in Table I.

Model	Accuracy	F1_score
Random Forest	0.686667	0.678082
Combine Model	0.646667	0.636986
XGBoost	0.630000	0.497738
Support Vector Machine	0.666667	0.489796
Logistic Regression	0.630000	0.468900

TABLE I: Model Performance

According to the table, the Random Forest model achieved the highest accuracy of 0.686667 and an F1 score of 0.678082. Based on these results, we can conclude that the Random Forest model outperformed the other models in terms of both accuracy and F1 score. The Combine Model also performed reasonably well, demonstrating the potential benefits of leveraging multiple models in ensemble approaches. However, the XGBoost, SVM, and Logistic Regression models showed comparatively lower performance.

V. Conclusion and Future Work

In conclusion, this project aimed to tackle the task of duplicate question pair detection on the Quora platform using natural language processing (NLP) techniques and machine learning algorithms. Through exploratory data analysis, text preprocessing, feature engineering, and model building, we developed models capable of predicting whether a pair of questions are duplicates or not. However, it is important to acknowledge the limitations and potential for future work in this study. Future research could focus on improving the quality and reliability of the ground truth labels through additional expert input or crowdsourcing techniques. Furthermore, the models developed in this project can be further fine-tuned and optimized to achieve even better performance. Hyperparameter tuning, exploring different feature engineering techniques, or incorporating advanced deep learning models such as transformers or recurrent neural networks (RNNs) could be explored to enhance the models' predictive capabilities. Furthermore, the models developed in this project can

be further fine-tuned and optimized to achieve even better performance. Hyperparameter tuning, exploring different feature engineering techniques, or incorporating advanced deep learning models such as transformers or recurrent neural networks (RNNs) could be explored to enhance the models' predictive capabilities. Additionally, the research can be extended by considering larger and more diverse datasets. Incorporating external knowledge sources could help create more robust and generalizable models for duplicate question pair detection.

REFERENCES

[1] Imtiaz, Zainab, et al. "Duplicate questions pair detection using siamese malstm." IEEE Access 8 (2020): 21932-21942.

[2] L. Wang, L. Zhang and J. Jiang, "Detecting Duplicate Questions in Stack Overflow via Deep Learning Approaches," 2019 26th Asia-Pacific Software Engineering Conference (APSEC), Putrajaya, Malaysia, 2019, pp. 506-513, doi: 10.1109/APSEC48747.2019.00074.

[3] D. A. Prabowo and G. Budi Herwanto, "Duplicate Question Detection in Question Answer Website using Convolutional Neural Network," 2019 5th International Conference on Science and Technology (ICST), Yogyakarta, Indonesia, 2019, pp. 1-6, doi: 10.1109/ICST47872.2019.9166343.

[4] Zhou, Qifeng, Xiang Liu, and Qing Wang. "Interpretable duplicate question detection models based on attention mechanism." Information Sciences 543 (2021): 259-272.

[5] (2019). Imtiaz, Zainab, et al. "Duplicate questions pair detection using siamese malstm." IEEE Access 8 (2020): 21932-21942.

[6] Xu, Zhuojia, and Hua Yuan. "Forum duplicate question detection by domain adaptive semantic matching." IEEE Access 8 (2020): 56029-56038.

[7] Ansari, Navedanjum, and Rajesh Sharma. "Identifying semantically duplicate questions using data science approach: A quora case study." arXiv preprint arXiv:2004.11694 (2020).

[8] Kumari, Reetu, et al. "Detection of semantically equivalent question Pairs." Intelligent Human Computer Interaction: 12th International Conference, IHCI 2020, Daegu, South Korea, November 24–26, 2020, Proceedings, Part I 12. Springer International Publishing, 2021.

[9] Kumar, Akshi. "Using cognition to resolve duplicacy issues in socially connected healthcare for smart cities." Computer Communications 152 (2020): 272-281.

[10] Tambakhe, Ms Vishwaja M., and Dr Kishor P. Wagh. "Review on Exploring Similarity between Two Questions Using Machine Learning." International Journal of Scientific Research in Computer Science, Engineering and Information Technology (2021)

[11] Tambakhe, M. M., and D. P. Wagh. "Duplicate Question Pair Detection with Machine Learning." BULLETIN MONUMENTAL 22.7 (2021).

[12] Engel, Singh, Madhusudan, et al., eds. Intelligent Human Computer Interaction: 12th International Conference, IHCI 2020, Daegu, South Korea, November 24–26, 2020, Proceedings, Part I. Vol. 12615. Springer Nature, 2021.

[13] Z.C. Lipton, The mythos of model interpretability. CoRR abs/1606.03490. 2016. URL:http://arxiv.org/abs/1606.03490, arXiv:1606.03490..

[14] Y. Nie, M. Bansal, Shortcut-stacked sentence encoders for multi-domain inference, in: Proceedings of the 2nd Workshop on Evaluating Vector Space Representations for NLP, Association for Computational Linguistics, Copenhagen, Denmark, 2017. pp. 41–45. URL:http://www.aclweb.org/anthology/ W17-5308..

[15] J. Devlin, M.W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186, https://doi.org/10.18653/v1/N19-1423, URL: https://www.aclweb.org/anthology/N19-1423.

[16] R. Ghaeini, S.A. Hasan, V. Datla, J. Liu, K. Lee, A. Qadir, Y. Ling, A. Prakash, X. Fern, O. Farri, Dr-bilstm: Dependent reading bidirectional lstm for natural language inference, in: Long Papers Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1, Association for Computational Linguistics, 2018, pp. 1460–1469, https://doi.org/10.18653/v1/ N18-1132, URL:http://aclweb.org/anthology/N18-1132.

[17] X. Zhu, P. Sobhani, H. Guo, Long short-term memory over recursive structures, Proceedings of International Conference on, Mach. Learning (2015) 1604–1612.

[18] Z. Wang, W. Hamza, R. Florian, Bilateral multi-perspective matching for natural language sentences, in, in: Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence IJCAI-17, 2017, pp. 4144–4150, https://doi.org/10.24963/ijcai.2017/579.

[19] A. Severyn, A. Moschitti, Learning to rank short text pairs with convolutional deep neural networks, in: Baeza-Yates, R.A., Lalmas, M., Moffat, A., Ribeiro-Neto, B.A. (Eds.), Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, Santiago, Chile, August 9–13, 2015, ACM. pp. 373–382. URL:https://doi.org/10.1145/2766462.2767738, doi: 10.1145/2766462.2767738..