Enhancing Duplicate Question Detection with RoBERTa Cross-Encoders and Explicit Features: A Hybrid Approach

Jhon Dave C. Bohol
Computer Science Department, College of Computing and Information Sciences, Caraga State University

Diana J. Gonzales
Computer Science Department, College of Computing and Information Sciences, Caraga State University

Roland P. Abao
Computer Science Department, College of Computing and Information Sciences, Caraga State University

Cristopher C. Abalorio
Computer Science Department, College of Computing and Information Sciences, Caraga State University

https://doi.org/10.5109/7395765

出版情報：Proceedings of International Exchange and Innovation Conference on Engineering & Sciences (IEICES). 11, pp.1918-1925, 2025-10-30. International Exchange and Innovation Conference on Engineering & Sciences
バージョン：

Enhancing Duplicate Question Detection with RoBERTa Cross-Encoders and Explicit Features: A Hybrid Approach

Jhon Dave C. Bohol[1], Diana J. Gonzales[2], Roland P. Abao[3], Cristopher C. Abalorio[4]
[1,2,3,4]Computer Science Department, College of Computing and Information Sciences, Caraga State University – Main Campus, Ampayon, Butuan City, 8600, Philippines
rpabao@carsu.edu.ph[3], ccabalorio@carsu.edu.ph[4]

Abstract: *This study investigates duplicate question detection to enhance the efficiency of Community Question Answering (CQA) platforms like Quora. It evaluates RoBERTa cross-encoder models, highlighting the model's semantic strength while addressing its limitations in capturing explicit lexical and statistical signals. Four model configurations are tested on 400,000 Quora Question Pairs: (1) Base RoBERTa, (2) RoBERTa with engineered features (e.g., word overlap, length metrics), (3) RoBERTa with TF-IDF similarity scores, and (4) RoBERTa with both feature types. Results show that the combined features model achieved the best performance (F1: 0.8716, Accuracy: 0.9010, AUC: 0.9629), outperforming the base model across all metrics. The analysis confirms that integrating contextual embeddings with structural and statistical features yields more accurate classification, reducing both false positives and false negatives. These findings support the adoption of hybrid approaches that blend deep learning with feature engineering for robust and scalable duplicate question detection.*
Keywords: Duplicate question detection; RoBERTa; Feature engineering; TF-IDF features; Text similarity

1. INTRODUCTION

Community Question Answering (CQA) platforms like Quora have become essential for knowledge dissemination, yet they face a significant challenge: the widespread presence of duplicate questions. Such redundancy fragments information, impairs search efficiency, and diminishes user engagement [1], making accurate automated detection of these questions crucial for maintaining platform quality and utility.

Addressing this challenge requires integrating AI with traditional manual moderation tasks such as banning, blocking, or removing redundant data. While manual moderation is thorough, it lacks the scalability needed for growing platforms with expanding user bases. AI systems offer a promising solution by automating duplicate detection and content categorization, thereby reducing manual workload while maintaining quality standards. Machine learning techniques like Support Vector Machine (SVM) and Random Forest have demonstrated high accuracy rates in classification tasks [2], and AI tools like ChatGPT have shown potential in assisting professionals with personalized workflows [3]. This suggests that automated moderation combined with human oversight could create efficient hybrid systems that enhance both user experience and information quality on CQA platforms.

However, the technical implementation of such systems presents its own challenges. Traditional Natural Language Processing (NLP) techniques, including Term Frequency-Inverse Document Frequency (TF-IDF) [4], have historically addressed duplicate detection by focusing on lexical overlap. While useful for basic similarity detection, these methods often fall short in capturing deeper semantic relationships, particularly when questions significantly express the same intent using different words. The advent of transformer-based models [5], such as RoBERTa[6], has marked a significant advancement in this domain, offering rich contextual understanding through pre-training on vast text corpora. RoBERTa, particularly in a cross-encoder configuration [7], excels at modeling nuanced interactions between text pairs for semantic textual similarity tasks, making it well-suited for detecting duplicate questions that traditional methods might miss.

Despite these strengths, transformer models like RoBERTa may not inherently capture or prioritize all explicit structural or statistical cues that could be beneficial for distinguishing duplicate questions [8]. Information such as question length variations, shared interrogative structures, or keyword importance as highlighted by TF-IDF, might offer complementary signals to purely semantic embeddings. This study presents an empirical investigation into augmenting RoBERTa cross-encoders with such explicit features for duplicate question detection. We specifically explore the integration of a curated set of five engineered linguistic/structural features alongside TF-IDF derived similarity scores. Our central hypothesis is that a hybrid model, which leverages both RoBERTa's deep contextual understanding and the explicit signals from these additional features, will achieve superior performance compared to a baseline RoBERTa model that relies solely on semantic information. This investigation is conducted using the widely recognized Quora Question Pairs (QQP) dataset as a benchmark.

2. METHODS

The methodology employed in this study is structured to systematically evaluate the impact of integrating explicit textual features with a RoBERTa-based cross-encoder. This encompasses careful dataset acquisition and preprocessing, the detailed extraction of engineered and statistical feature sets, the definition of the core RoBERTa model, and a comprehensive experimental setup for training and subsequent evaluation.

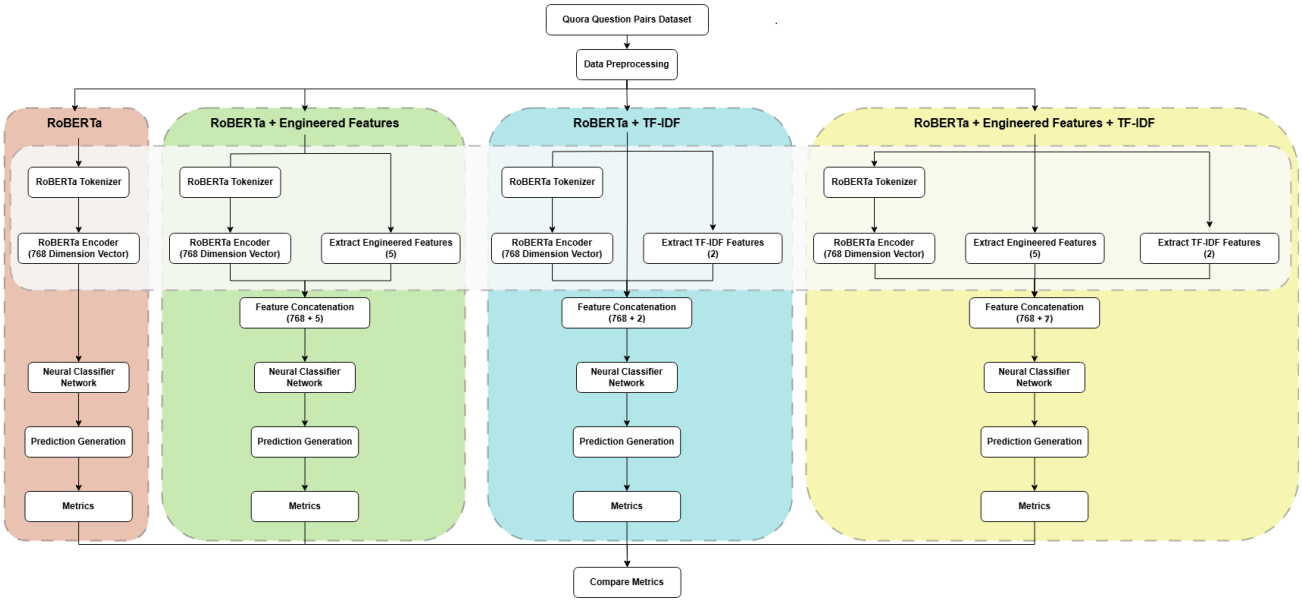

Fig. 1. Overview of the proposed model configurations for duplicate question detection using RoBERTa, engineered features, and TF-IDF, showing their processing pipelines and evaluation stages.

2.1 Dataset

The study utilized the publicly available **Quora Question Pairs (QQP) dataset** from Quora's Competition held on Kaggle, containing human-annotated duplicate/non-duplicate question pairs. A random sample of **400,000 pairs** was used. Standard **data cleaning** (null value handling, special character removal, lowercasing, whitespace normalization) was applied.

As clearly depicted in figure 2, the dataset exhibits a noticeable **class imbalance**. The count for non-duplicate pairs (label 0) is substantially higher than the count for duplicate pairs (label 1). Observing the y-axis, it can approximate the counts:

- `Non-duplicates (Label 0): ≈ 255,000 pairs`
- `Duplicates (Label 1): ≈ 145,000 pairs`

This corresponds to approximately **63.75% non-duplicate pairs** and **36.25% duplicate pairs** in the dataset sample used for training, validation, and testing. The dataset was then split using **stratified sampling** into 70% training, 20% validation, and 10% test sets, maintaining the original class distribution.

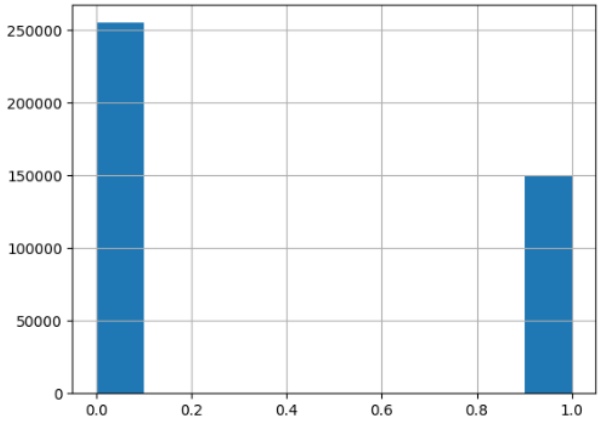

Fig. 2. Distribution of Target Labels (0: Non-Duplicate, 1: Duplicate) in the Dataset Sample (N=400,000).

2.2 Feature Extraction

To supplement the rich semantic representations learned by RoBERTa, two distinct categories of explicit features were extracted for each question pair.

Engineered Features: A set of five engineered features was designed to capture various lexical, structural, and simple semantic properties of the question pairs. These features are relatively simple to compute and interpret, providing explicit signals that might complement RoBERTa's more abstract representations:

a. Word Overlap (WO): This feature measures the shared vocabulary between the two questions using the Jaccard similarity coefficient. It is calculated as:

$$WO = \frac{|Words(Q1) \cap Words(Q2)|}{|Words(Q1) \cup Words(Q2)|}$$

b. *Length Difference (LD):* This captures the absolute difference in the number of tokens (words) between the two questions:

$$LD = |LengthTokens(Q1) - LengthTokens(Q2)|$$

c. *Length Ratio (LR):* This provides a normalized measure of length similarity, computed as the ratio of the token count of the shorter question to that of the longer question:

$$LR = \frac{min(LengthTokens(Q1), LengthTokens(Q2))}{max(LengthTokens(Q1), LengthTokens(Q2))}$$

d. *Character Ratio (CR):* Offering a finer-grained structural comparison than word count, this feature is the ratio of the character count (raw string length) of the shorter question to that of the longer question:

$$CR = \frac{min(LengthChars(Q1), LengthChars(Q2))}{max(LengthChars(Q1), LengthChars(Q2))}$$

e. *Question Type Similarity (QTS):* This is a binary feature. It is set to 1 if both questions begin with the same interrogative word (e.g., "what", "how", "why", "who", "where", "when", "which"), and 0 otherwise.

TF-IDF Features: To incorporate statistical information about term importance and discriminability, features based on Term Frequency-Inverse Document Frequency (TF-IDF) were derived. A TfidfVectorizer from the scikit-learn library was employed. Key parameters for this vectorizer included **ngram_range=(1,2)** (to consider both unigrams and bigrams) and **sublinear_tf=True** (applying logarithmic scaling to term frequencies, i.e., TF(t, d) = 1 + log(raw_count(t, d))). Common **English stop words** were also removed prior to vectorization. A crucial hyperparameter for the TfidfVectorizer is **max_features**, which limits the size of the vocabulary learned. To determine an appropriate value, a preliminary investigation was conducted by comparing the performance of the RoBERTa + Combined Features model using TF-IDF vocabularies of different sizes, specifically contrasting max_features=10000 with max_features=25000 with configuration of early stopping shown in Fig. 3. This sensitivity analysis, monitoring validation metrics such as Accuracy, indicated that the configuration with max_features=10000 consistently achieved comparable or slightly superior performance, and demonstrated more stable learning dynamics compared to the larger vocabulary size. Increasing the vocabulary beyond 10,000 features did not yield further improvements and risked introducing noise from rarer, less discriminative terms, aligning with observations that overly large vocabularies can sometimes dilute the signal[9]. Consequently, based on this empirical evidence suggesting a better balance of performance, noise reduction, and computational efficiency, max_features = 10000 was adopted for the TF-IDF vectorizer in all subsequent experiments reported in this study.

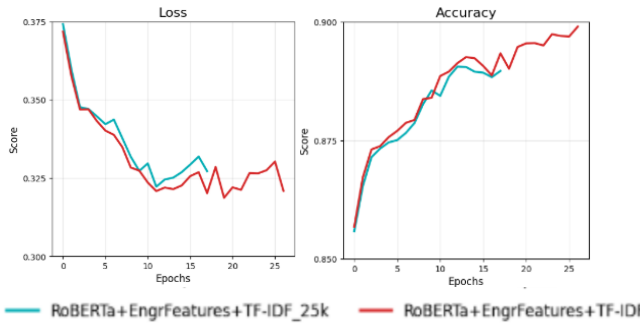

Fig. 3. Comparative Training History for RoBERTa + EF + TF-IDF with max_features=10000 to 25000.

The TF-IDF score for each term was then computed. Term Frequency (TF) for a term t in a question d was calculated using sublinear TF scaling, as defined by:

$$TF(t,d) = 1 + log(rawcount(t,d))$$

The IDF for a term t is calculated as:

$$IDF(t) = \frac{log((N+1)}{(df(t)+1)) + 1}$$

Where N is the total number of documents in the training corpus, and df(t) is the number of documents containing term *t*. The TF-IDF score for term *t* in document *d* is then:

$$TF - IDF(t,d) = TF(t,d) * IDF(t)$$

This vectorizer was fitted *only* on the combined cleaned texts of question1 and question2 from the *training set*. Using the resulting TF-IDF vectors for each question in a pair (v1, v2), two features were computed:

TF-IDF Cosine Similarity (TFCS):

$$TFCS = \frac{(v1 \cdot v2)}{(||v1|| * ||v2||)}$$

TF-IDF Distance (TFCD):

$$TFCD = 1 - TFCS$$

2.3 RoBERTa Model

The foundational component of all models in this study is the roberta-base pre-trained language model, sourced from the HuggingFace Transformers library. It is employed within a cross-encoder framework. In this setup, the two questions of a pair are not encoded independently but are instead concatenated and fed as a single sequence to the model. This allows RoBERTa's self-attention mechanisms to model interactions and dependencies directly between the tokens of both questions from the very first transformer layer, facilitating a deep and fine-grained comparison. The input preparation involves combining Question 1 and Question 2 with RoBERTa's special tokens: <s> Question 1 </s> </s> Question 2 </s>. This combined sequence is then tokenized using the roberta-base tokenizer. To ensure uniform input dimensions suitable for batch processing, the tokenized sequences are either padded with special padding tokens or truncated to a fixed maximum length of 64 tokens. Attention masks are also generated to indicate to the model which tokens are real versus padding.

The roberta-base architecture comprises 12 layers of transformer encoder blocks. In our fine-tuning strategy, to preserve the robust general linguistic knowledge acquired during RoBERTa's extensive pre-training while allowing adaptation to the specific duplicate question detection task, the parameters of the RoBERTa embedding layer and its first 6 transformer encoder layers were frozen. The parameters of the subsequent (top) 6 transformer layers were unfrozen and fine-tuned. The final hidden state vector corresponding to the initial special <s> token (often referred to as the [CLS] token in BERT-like models), which has a dimensionality of 768, is extracted. This vector serves as the primary learned semantic representation of the input question pair.

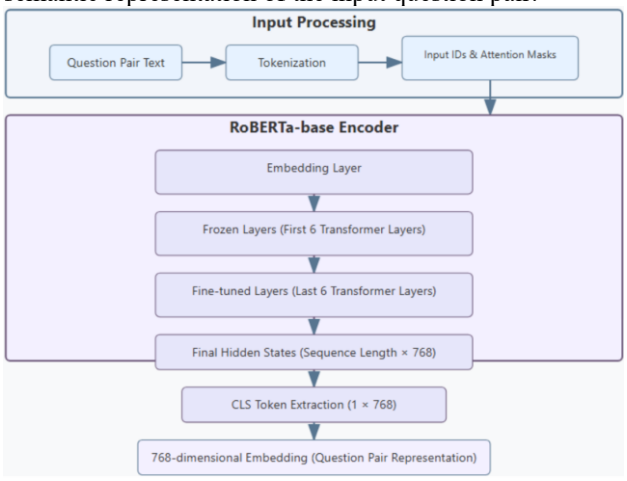

Fig. 4. Base RoBERTa Architecture Diagram.

2.4 Concatenation Stage

For the model configurations that leverage the explicit features described in Section 2.2, a feature concatenation

stage is introduced after the RoBERTa encoding. The 768-dimensional <s> token embedding, which captures the deep contextual relationship learned by RoBERTa, is combined with the separately computed feature vectors. Specifically, the 5 engineered features are formed into a 5-dimensional vector, and the 2 TF-IDF features are formed into a 2-dimensional vector. Depending on the model configuration, either the engineered feature vector, the TF-IDF feature vector, or a direct concatenation of both (resulting in a 7-dimensional vector of explicit features) is then appended to the end of the 768-dimensional <s> embedding derived from RoBERTa. This operation creates a single, longer, combined feature vector for each question pair. For the 4 model configurations it will have its respective input size:

- Base RoBERTa (768)
- RoBERTa + EF (768 + 5 = 773)
- RoBERTa + TF-IDF (768 + 5 + 2 = 770)
- RoBERTa + EF + TF-IDF (768 + 5 + 2 = 775)

This concatenation approach aims to provide the classifier with a multi-faceted representation, allowing it to leverage both the learned semantic information from RoBERTa and the explicit signals from the statistical and engineered features.

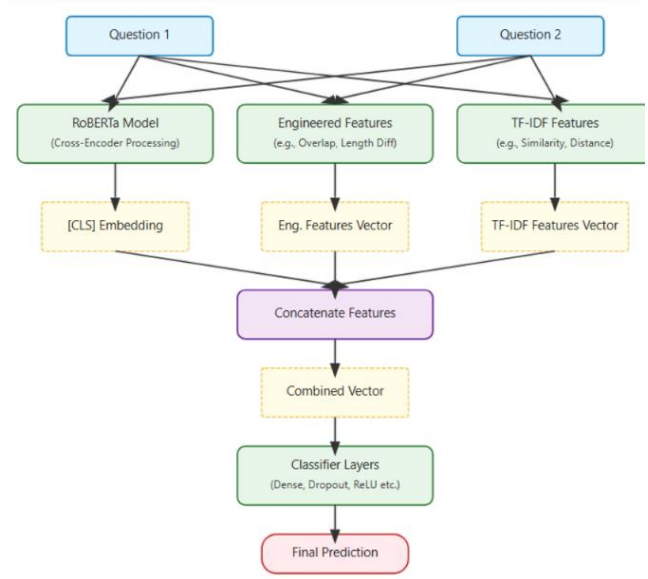

Fig. 5. Feature Concatenation Diagram.

2.5 Classifier Architecture

The final stage of each model configuration is a classifier head, which takes the (potentially feature-augmented) vector representation of the question pair and produces the binary classification output (duplicate or non-duplicate). This classifier head is implemented as a Multi-Layer Perceptron (MLP) consisting of several fully-connected layers with specific regularization techniques.

The input dimensionality to this MLP varies based on the model configuration, as described in Section 2.4. The MLP architecture employs a bottleneck structure with three hidden layers, having 512, 256, and 128 units, respectively. Each of these linear transformations in these hidden layers is followed by Layer Normalization, which helps stabilize the learning process by normalizing the activations across the feature dimension. Subsequently, a Dropout layer is applied with progressively decreasing rates (0.6 after the first hidden layer, 0.48 after the second, and 0.36 after the third) to mitigate overfitting. A Rectified Linear Unit (ReLU) activation function is then applied in each hidden layer to introduce non-linearity, enabling the network to learn complex patterns and interactions among the diverse input features. The 128-dimensional output vector from the third hidden layer is then projected by a final linear layer to a single output unit. This unit produces a raw score, or logit. This logit is passed through a sigmoid activation function, $\sigma(z) = 1 / (1 + e^{(-z)})$, which transforms the unbounded logit into a probability score constrained between 0 and 1. This probability represents the model's estimated likelihood that the input question pair is a duplicate.

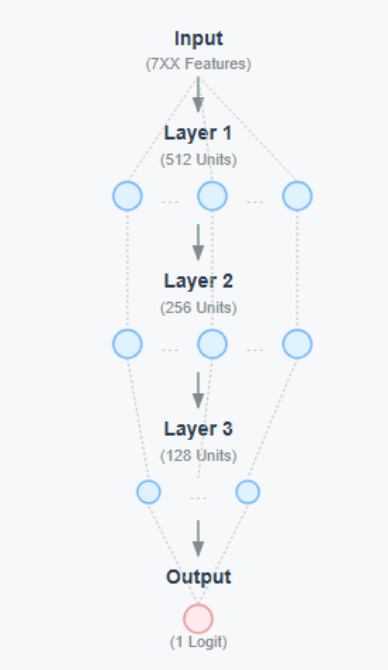

Fig. 6. Multi-layer Classifier Architecture.

3. EVALUATION METHODOLOGY

3.1 Experimental Setup and Configurations

To rigorously assess and compare the effectiveness of the different model configurations, a well-defined experimental setup was established for training and evaluation. Four primary model configurations were evaluated:

(1) Base RoBERTa,
(2) RoBERTa + Engineered Features (EF),
(3) RoBERTa + TF-IDF (TF), and
(4) RoBERTa + Combined Features (EF+TF).

All models were trained on an NVIDIA Tesla V100 GPU. Key training parameters included the initializing Seed for reproducibility, AdamW optimizer, differential learning rates (1e-5 for encoder, 3e-4 for classifier), a weight decay of 0.01, a cosine annealing learning rate schedule, and a batch size of 128. Label smoothing (0.05) was applied to the Binary Cross-Entropy loss. Mixed-precision training and gradient clipping (max norm 1.0) were utilized. Training proceeded for up to 50 epochs with early stopping based on validation F1 score (patience 5) and validation loss (patience 7). The checkpoint yielding the best validation F1 was selected for each model.

3.2 Performance Metrics

Model performance was primarily assessed using three standard metrics:

Accuracy: The proportion of all question pairs correctly classified.

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)}$$

F1 Score: The harmonic mean of precision and recall, providing a balanced measure suitable for imbalanced datasets like QQP.

$$Precision(P) = \frac{TP}{(TP + FP)}$$

$$Recall(R) = \frac{TP}{(TP + FN)}$$

$$F1Score = \frac{2 * (P * R)}{(P + R)}$$

Area Under the Receiver Operating Characteristic Curve (AUC-ROC): This metric measures the model's ability to discriminate between positive (duplicate) and negative (non-duplicate) classes across all possible classification thresholds. It is derived from the ROC curve, which plots the True Positive Rate (Recall) against the False Positive Rate.

$$FPR = \frac{FP}{(FP + TN)}$$

$$AUC - ROC = \int [FPR = 0to1] TPRd(FPR)$$

3.3 Threshold Optimization Strategy

Given that the models output probability scores, a decision threshold is required for binary classification. To optimize F1 performance, an optimal threshold was empirically determined for each model configuration. This involved evaluating the F1 score on the validation set predictions across a range of potential thresholds (from 0.0 to 1.0). The threshold yielding the maximum F1 score on this validation data was selected as the optimal threshold for that model. These model-specific optimal thresholds, presented in Table 1, were subsequently used for generating final predictions on the unseen test set.

Table 1. Optimal thresholds for each configuration

Model Variation	Optimal Threshold
Base RoBERTa	0.5508
RoBERTa + Engineered Features	0.4257
RoBERTa + TF-IDF	0.4758
RoBERTa + Engineered Features + TF-IDF	0.3625

4. RESULTS AND DISCUSSION

4.1 Performance Metrics

To understand how each model configuration learned and adapted to the duplicate question detection task over time, key performance metrics—validation loss, accuracy, F1 score, and AUC-ROC—were monitored on the validation set throughout the training process. Figure 7 presents the evolution of these validation metrics for all four models—(1) Base RoBERTa, (2) RoBERTa + Engineered Features (EF), (3) RoBERTa + TF-IDF Features (TF), and (4) RoBERTa + Combined Features (EF+TF)—over a fixed span of 30 training epochs. These plots visually depict the learning dynamics and convergence behavior, allowing for a preliminary comparison of their relative performance trajectories before the application of early stopping criteria.

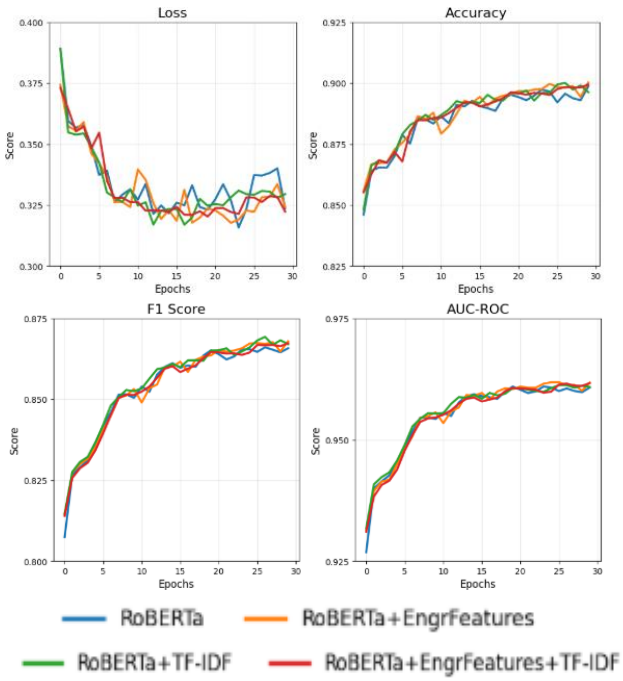

Fig. 7. Validation Metrics History Over 30 Epochs for All Configurations.

Observing the validation metrics over these 30 epochs (Fig. 7), several trends emerge. All models typically exhibited a steep improvement in performance (decrease in loss, increase in F1 score) during the initial epochs (approximately the first 5-10), indicating rapid initial learning. Following this phase, the learning curves tended to stabilize or show more gradual improvement. Notably, the configurations incorporating engineered features (RoBERTa + EF and RoBERTa + Combined EF+TF) often appeared to reach or maintain slightly superior performance in terms of F1 score and accuracy in the later epochs of this fixed 30-epoch observation period compared to the Base RoBERTa and RoBERTa + TF-IDF models. For instance, the validation F1 score for the RoBERTa + Combined Features model is generally tracked at or above the other configurations from the mid-stages onwards. The validation loss for the feature-enhanced models also tended to be slightly lower and more stable in later epochs. These comparative dynamics over a fixed training period provide initial insights into the learning efficiency and potential of each configuration.

While the 30-epoch comparison offers a broad view, the final model for evaluation was selected based on an early stopping mechanism. This mechanism halted training and saved the model checkpoint corresponding to the epoch where the peak F1 score was achieved on the validation set, to prevent overfitting and select the model

state with the best-observed generalization. The specific early stopping points and corresponding validation F1 scores for each configuration were as follows:

- The **Base RoBERTa** model reached its peak validation F1 score of 0.8621 at **epoch 20**.
- The **RoBERTa + Engineered Features (EF)** model achieved its highest validation F1 score of 0.8663 later in training, at **epoch 26**.
- The **RoBERTa + TF-IDF (TF)** model, similar to the baseline, also peaked at **epoch 20** with a validation F1 score of 0.8629.
- The **RoBERTa + Combined Features (EF+TF)** model, which ultimately demonstrated the best test set performance, required the most training iterations, reaching its optimal validation F1 score of 0.8672 at **epoch 27**.

This checkpoint selection strategy ensures that each model is evaluated at its empirically determined optimal point of generalization on unseen validation data. The observation that models incorporating engineered features required more epochs to converge optimally underscores the increased complexity of learning the intricate interplay between the deep semantic embeddings and these diverse explicit features.

4.2 Quantitative Performance on Test Set

Following the selection of the optimal checkpoint for each model configuration as described above, the models were evaluated on the unseen, held-out test set. The empirically determined optimal classification thresholds (as presented in Table 1) were applied to generate the final binary predictions. The key performance metrics derived from this rigorous test set evaluation are detailed in Table 2.

Table 2. Test Set Performance Metrics

Model Variations	Accuracy	F1 Score	AUC-ROC	Training Time
Base RoBERTa	0.8951	0.8642	0.9604	**2h 13m 10s**
RoBERTa + EF	0.8982	0.8683	0.9619	2h 16m 42s
RoBERTa + TF-IDF	0.8946	0.8627	0.9599	2h 15m 14s
RoBERTa + EF + TF-IDF	**0.9010**	**0.8716**	**0.9629**	2h 18m 30s

As clearly indicated in Table 2, the RoBERTa + Combined Features (EF+TF) model achieved superior performance across all primary evaluation metrics. It obtained an Accuracy of 0.9010, an F1 Score of 0.8716, and an AUC-ROC of 0.9629. This performance marks a statistically significant and practically meaningful improvement over the Base RoBERTa model, which served as our baseline and recorded an Accuracy of 0.8951, an F1 Score of 0.8642, and an AUC-ROC of 0.9604. The integration of engineered features alone (RoBERTa + EF) also yielded a noticeable improvement over the baseline, with an F1 Score of 0.8683. However, the model augmented solely with TF-IDF features (RoBERTa + TF) did not surpass the baseline performance on the test set, registering a slightly lower F1 Score of 0.8627. The training times to reach these optimal epochs, also listed in Table 2, show that the more complex hybrid models (EF and EF+TF) took slightly longer overall due to the greater number of epochs required for convergence, but as previously noted, the per-epoch time increase was minimal.

4.3 Impact of Feature Integration and Error Analysis

The superior test set performance of the RoBERTa + Combined Features model strongly points towards a synergistic interaction when RoBERTa's semantic capabilities are complemented by both engineered structural/lexical features and statistical TF-IDF scores. While the RoBERTa + Engineered Features model consistently outperformed the Base RoBERTa, and the RoBERTa + TF-IDF model did not demonstrate such gains alone in this cross-encoder context, their combination in the RoBERTa + Combined Features configuration proved most effective. This suggests that TF-IDF's statistical term importance is better utilized when contextualized by RoBERTa's deep semantics and the explicit structural cues from engineered features, leading to a richer, multi-faceted representation for more informed classifications.

Table 3. Confusion Matrices Difference between the Four (4) Configurations

Model Variations	TP	FP	TN	FN
Base RoBERTa	13350	2777	22455	1418
RoBERTa + EF	13426 (+76)	2730 (-47)	22502 (+47)	1342 (-76)
RoBERTa + TF-IDF	13251 (-99)	2700 (-77)	22532 (+77)	1517 (+99)
RoBERTa + EF + TF-IDF	**13441 (+91)**	**2633 (-144)**	**22599 (+144)**	**1327 (-91)**

A more granular examination of classification errors, facilitated by analyzing the confusion matrices, further illuminates the benefits of the combined approach. These matrices break down predictions into **True Positives (TP)**, where actual duplicate pairs are correctly identified as duplicates; **False Positives (FP)**, where non-duplicate pairs are incorrectly identified as duplicates; **True Negatives (TN)**, where non-duplicate pairs are correctly identified as non-duplicates; and **False Negatives (FN)**, where actual duplicate pairs are incorrectly identified as non-duplicates. Numerical details comparing these counts across configurations are presented in Table 3. The RoBERTa + Combined Features model achieved the most substantial and balanced reduction in both FPs and FNs when benchmarked against the Base RoBERTa

model. Specifically, on the test set, the RoBERTa + Combined Features model reduced the count of FPs by approximately 144 instances and FNs by 91 instances relative to the Base RoBERTa configuration. This concurrent reduction in both primary types of classification errors directly contributes to its higher F1 score and overall accuracy. It indicates an improved ability of this hybrid model to both identify true duplicates (thereby reducing FNs) and correctly discern distinct non-duplicate pairs (thereby reducing FPs), even when those pairs might share some superficial lexical or topical similarities.

The empirically determined optimal classification thresholds also offer significant insights into the impact of feature integration on model behavior. The Base RoBERTa model's optimal threshold was 0.5508. In stark contrast, the RoBERTa + Combined Features model achieved its optimal F1 score at a much lower threshold of 0.3625. This significant downward shift in the optimal decision point suggests that the integration of explicit features fundamentally altered the distribution of the model's output probability scores, likely creating a better separation between the duplicate and non-duplicate classes. On an imbalanced dataset such as QQP, where duplicate pairs constitute the minority class (approximately 36.25% in our sample), a lower threshold often aids in improving the recall of this minority class. The RoBERTa + Combined Features model's ability to find its optimal F1 balance at this substantially lower threshold, without a disproportionate loss in precision (as evidenced by its high F1 score), is indicative of its enhanced overall discriminative capability stemming from the richer input representation.

4.3 Feature Importance Insights

To better understand the relative contributions of the different explicit features within the best-performing RoBERTa + Combined Features model, a permutation feature importance analysis was conducted directly on the test set. This technique assesses a feature's importance by measuring the decrease in a chosen performance metric—in this study, the F1 score—when the values of that single feature are randomly shuffled, thereby breaking its learned relationship with the true labels while keeping other features and the model parameters constant. The F1 score was calculated relative to an internal baseline F1 of 0.8668 observed during this specific analysis for the unshuffled data. The permutation importance analysis revealed that *TF-IDF Similarity* (which caused a mean ΔF1 of approximately +0.0010 when permuted) and *Word Overlap* (mean ΔF1 ≈ +0.0004) were the two most influential explicit features. The positive ΔF1 indicates that their random permutation led to the largest average decrease in the F1 score, underscoring that the model significantly relies on these direct signals of statistical term overlap (weighted by rarity, in the case of TF-IDF) and basic lexical sharing (Jaccard index for word overlap) as positive contributors to its predictive performance. These features appear to provide valuable, discriminative information that complements RoBERTa's more abstract learned semantic representations. The *Length Difference* feature also showed a small positive mean importance, suggesting a minor beneficial contribution to the model's decisions.

Table 4. Feature Importance Scores

Feature Display	Importance Mean	Relative Importance
TF-IDF Similarity	**0.001024**	**~29.2%**
Word Overlap	0.000372	~10.6%
Length Difference	0.000053	~1.5%
Same Question Type	-0.000059	(~1.7%)
Character Ratio	-0.000363	(~10.4%)
Length Ratio	-0.000728	(~20.8%)
TF-IDF Distance	-0.000903	(~25.8%)

5. CONCLUSION

This study systematically investigated the efficacy of augmenting RoBERTa cross-encoders with engineered linguistic/structural features and TF-IDF-based statistical scores for the critical task of duplicate question detection. Through comprehensive experiments conducted on a large-scale sample of the Quora Question Pairs dataset, we have demonstrated that such a hybrid modeling approach yields significant and practically relevant improvements in performance.

The primary finding is that the model integrating RoBERTa's deep contextual embeddings with both the curated set of engineered features and the TF-IDF derived scores—the RoBERTa + Combined Features model—achieved the highest accuracy (0.9010) and F1 score (0.8716) on the test set. This performance notably surpassed that of the baseline RoBERTa model as well as configurations that were augmented with only engineered features or only TF-IDF features. This underscores a synergistic effect where diverse information sources complement RoBERTa's semantic understanding, leading to a more robust and accurate classification system for identifying duplicate questions. The findings affirm the continued relevance and value of strategic feature engineering, even in an era dominated by powerful pre-trained language models. For tasks like duplicate question detection, where subtle lexical, structural, and statistical cues can be highly discriminative, explicitly providing these signals to a sophisticated model like RoBERTa can significantly enhance its ability to discern semantic equivalence. This study contributes a validated methodology for such feature integration and provides empirical evidence for the benefits of adopting hybrid model architectures to improve the accuracy of duplicate detection systems. Future research avenues could include the exploration of a broader and more diverse array of engineered features (This study only used 5 engineered features, what it indicates here that there is a potential of integrating more engineered features will significant improve the model),

investigation into more sophisticated mechanisms for feature integration beyond simple concatenation (such as attention-based fusion mechanisms), assessment of the impact of these techniques on larger and more varied transformer architectures, and a thorough examination of the generalizability of these findings across different datasets, application domains, and languages.

6. REFERENCES

[1] Imtiaz, Z., Umer, M., Ahmad, M., Ullah, S., Choi, G. S., & Mehmood, A. (2020). Duplicate questions pair detection using Siamese MaLSTM. IEEE Access, 8, 21932–21942. https://doi.org/10.1109/ACCESS.2020.2969041

[2] Luzorata, J. G., Bocobo, A. E., Detera, L. M., Pocong, N. J. B., & Sajonia, A. P. (2023). Assessment of Land Use Land Cover Classification using Support Vector Machine and Random Forest Techniques in the Agusan River Basin through Geospatial Techniques. Proceedings of International Exchange and Innovation Conference on Engineering & Sciences (IEICES), 9, 240-246. https://doi.org/10.5109/7157978

[3] Nesnawy, S., Radwan, M. F., & Mahrous, F. M. (2024). The Potential of ChatGPT in Assisting Healthcare Professionals with Personalized Time-Scheduled Care Plans: Opportunities and Challenges. Proceedings of International Exchange and Innovation Conference on Engineering & Sciences (IEICES), 10, 897-904. https://doi.org/10.5109/7323366

[4] Salton, G., & Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. Information Processing & Management, 24(5), 513–523. https://doi.org/10.1016/0306-4573(88)90021-0

[5] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention Is All You Need. In Advances in Neural Information Processing Systems, 30. https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html

[6] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). RoBERTa: A Robustly Optimized BERT Pretraining Approach. arXiv:1907.11692. https://arxiv.org/abs/1907.11692

[7] Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP) (pp. 3982–3992). Association for Computational Linguistics. https://doi.org/10.18653/v1/D19-1410

[8] Joshi, B., Shah, N., Barbieri, F., & Neves, L. (2020). The Devil is in the Details: Evaluating Limitations of Transformer-based Methods for Granular Tasks. In International Conference on Computational Linguistics (pp. 3652–3659). https://doi.org/10.18653/V1/2020.COLING-MAIN.326

[9] Zhou H., Guo J., Wang Y. (2015). A feature selection approach based on term distributions for text classification. Journal of Computational Biology and Chemistry, 13(5), 57-65.https://pmc.ncbi.nlm.nih.gov/articles/PMC4771689/