International Journal of Computer, Network Security and Information System (IJCONSIST)
Vol. 6 No. 2, Maret 2025

Comparing Structured Prompts for Denoising Noisy Certificate Text

Dimas Saputra
Computer Science Faculty
UPN "Veteran" East Java
Surabaya, Indonesia
21081010151@student.upnjatim.ac.id

I Gede Susrama Mas Diyasa
Computer Science Faculty
UPN "Veteran" East Java
Surabaya, Indonesia
igsusrama.if@upnjatim.ac.id

Eva Yulia Puspaningrum
Computer Science Faculty
UPN "Veteran" East Java
Surabaya, Indonesia
evapuspaningrum.if@upnjatim.ac.id

Wan Suryani Wan Awang
Faculty of Informatics and Computing
Universiti Sultan Zainal Abidin Besut Campus, Terengganu
Besut, Malaysia
suryani@unisza.edu.my

***Abstract*— This study addresses the challenge of noisy text resulting from Optical Character Recognition (OCR) on certificates, which hinders effective classification in Recognition of Prior Learning (RPL) contexts. To mitigate this issue, researchers propose the use of prompt-based denoising leveraging a Large Language Model (LLM), specifically the Gemini model, to refine the extracted text prior to classification. The methodology integrates OCR via PyTesseract, LLM-driven denoising using structured prompts (CSIR, CLEAR, and CO-STAR), and a BERT-base-uncased model for classification. Synonym replacement is also applied for data augmentation. Performance evaluation is conducted using accuracy, validation accuracy, confusion matrix, and classification reports. The results demonstrate a substantial improvement in classification performance. The baseline scenario achieved an accuracy of 82.14%, whereas the best-performing prompt structure, CO-STAR, reached 98.81%, marking an increase of over 15 percentage points. Similar trends were observed across all evaluation metrics, with CO-STAR delivering the highest precision, recall, and F1-score values. In conclusion, incorporating LLM-driven denoising through effective prompt strategies enhances the quality of OCR-extracted text and significantly boosts classification outcomes in certificate-based applications.**

Keywords: Text Denoising; Large Language Model (LLM); Certificate Classification; Optical Character Recognition (OCR); Prompt Engineering

I. INTRODUCTION

In recent years, the growing importance of digital credential evaluation has prompted significant interest in automating the classification of certificate documents. This need is particularly evident in the context of Recognition of Prior Learning (RPL), where non-academic experiences such as internships, bootcamps, or professional certifications are assessed for their relevance to academic or career pathways. As technological advancements accelerate the pace of knowledge acquisition, enabling systems to interpret and classify such credentials accurately is becoming critical. A key challenge lies in preprocessing, converting certificate images into text via OCR. While OCR tools like PyTesseract provide a practical means for text extraction, the resulting outputs are often noisy [1] (especially when dealing with scanned documents that are low-resolution, rotated, or degraded in quality). These distortions hinder downstream tasks such as classification and semantic analysis, necessitating an effective denoising mechanism.

This study investigates the novel application of large language models (LLMs) to denoise text extracted from certificates via optical character recognition (OCR). Prior work has primarily focused on text augmentation using LLMs for synthetic data generation, as demonstrated in recent studies such as BERT-based conflict sentiment identification that leveraged Llama 3 for data expansion [2]. However, the application of LLMs for denoising (particularly in certificate classification remains underexplored). This study seeks to fill this gap by introducing a structured prompts denoising approach using the Gemini model, evaluating its impact on the quality of textual data, and ultimately, on classification accuracy. Furthermore, three prompt structures (CSIR, CLEAR, and CO-STAR) were selected based on their relevance in recent prompting literature to investigate their effectiveness in guiding the denoising process.

This research employs a pipeline that begins with OCR processing using PyTesseract, followed by prompt-based denoising using Gemini, synonym replacement for minor augmentation, and classification using the BERT-base-uncased model. Authors hypothesize that combining denoising and data augmentation can enhance classification accuracy for certificates across seven categories: Game Applications, User Interface Design, Computer Networks, Machine Learning, Project Management, Mobile Programming, and Web

Programming. Evaluation metrics (including accuracy, validation accuracy, confusion matrix, precision, recall, and F1-score) are used to assess the success of the approach. By offering insights into how prompt structure and LLMs can enhance OCR text preprocessing, this study contributes a new perspective to the automation of certificate analysis and document related text enhancement.

II. METHOD

In this research, the process begins by collecting certificate documents in PDF format and extracting their textual content using Optical Character Recognition (OCR) with the PyTesseract library. The raw OCR output is then subjected to a series of preprocessing steps, including text normalization (such as lowercasing), removal of punctuation, digits, and special characters, as well as filtering of stopwords and application of stemming techniques. Given that OCR frequently introduces noisy or garbled text, a crucial step in this pipeline involves applying text denoising using structured prompts processed by a large language model. These prompts guide the LLM to refine the content, improving its readability and semantic consistency before further processing.

To evaluate the impact of prompt design structure on denoising effectiveness, this study utilizes three narrative-based prompt structures: CSIR (Context, Specific Information, Intent/Goal, Result), CLEAR (Challenge, Limitation, Effect, Action, Result), and CO-STAR (Context, Objective, Style, Tone, Audience, Response). Each prompt design is used to guide the LLM in denoising the certificate text prior to training. The cleaned and augmented text data are then used to fine-tune a BERT-base-uncased model for multi-class classification into seven predefined course categories. Finally, the classification performance from each scenario is compared against a baseline model trained on uncleaned data (i.e., without prompt-based denoising and augmentation) to determine which prompt structure yields the most effective and accurate classification results.

A. Related Works

Prior studies relevant to the intersection of optical character recognition (OCR) denoising, large language model (LLM)-driven text refinement, and structured prompt engineering. By examining foundational work on OCR post-processing, LLM applications in document analysis, and prompt design frameworks, this review establishes the theoretical and methodological context for the current investigation into certificate text classification.

TABLE I. PREVIOUS STUDIES

Research	Key Components
Estimating Post-OCR Denoising Complexity on Numerical Texts [3]	Text Denoising on OCR
Structured clinical reasoning prompt enhances LLM's diagnostic capabilities in diagnosis please quiz cases [4]	Enhancement using LLM
QAIE: LLM-based Quantity Augmentation and Information Enhancement for few-shot Aspect-Based Sentiment Analysis [5]	Information Enhancement LLM-based
Enhancing structured data generation with GPT-4o evaluating prompt efficiency across prompt styles [6]	Structured Prompt Enhancement

As shown in the Table I, the first study, Estimating Post-OCR Denoising Complexity on Numerical Texts [3], focuses on quantifying denoising complexity for numerical data, proposing an estimator validated against advanced methods while analyzing alphabetical and numerical word contributions. While this previous work advances OCR error correction for numerical datasets, it diverges from the current research in two key aspects: (1) its exclusive focus on numerical content contrasts with the mixed-alphanumeric nature of certificate text in this study, and (2) its algorithmic denoising framework differs from the structured prompt engineering approach leveraging Gemini LLM employed here. Similarly, Structured Clinical Reasoning Prompts Enhances LLM's Diagnostic Capabilities [4] demonstrates how narrative frameworks improve LLM performance in medical diagnostics. However, this study applies prompts to refine clinical reasoning rather than denoise OCR artifacts, and its reliance on pre-curated, noise-free datasets contrasts with the current work's emphasis on repairing real-world noisy certificate text.

The third study, QAIE: LLM-Based Quantity Augmentation and Information Enhancement for Few-Shot ABSA [5], introduces a dual-module framework to enhance sentiment analysis via implicit information extraction. While QAIE excels in augmenting data for sentiment tasks, its application diverges from this research in intent: QAIE focuses on enriching sparse datasets for sentiment prediction, whereas the current study uses structured prompts to reconstruct noisy OCR outputs for classification. Lastly, Enhancing Structured Data Generation with GPT-4o [6] evaluates prompt styles (JSON, YAML, Hybrid CSV/Prefix) for structured data tasks, emphasizing trade-offs between readability, efficiency, and accuracy. Though both studies investigate prompt efficacy, this prior work targets output structuring rather than input denoising, underscoring the novelty of the current approach in applying prompt engineering to repair degraded OCR text rather than optimize data formatting. These distinctions highlight the unique contribution of this research in bridging OCR preprocessing and LLM-driven text reconstruction for noisy certificate classification.

B. Data Collection

This research employs data scraping as the primary method for collecting certificate data from publicly accessible sources. All scraping activities are conducted in accordance with ethical standards, respecting terms of service, user privacy, and data consent by avoiding the extraction of personal or sensitive information.

Data scraping is an automated method of extracting structured or unstructured content from web pages, APIs, or databases using scripts or tools [7]. As noted in academic literature, scraping enables large-scale data acquisition essential for AI training and analysis, but must be performed responsibly

to comply with legal frameworks and minimize bias or privacy risks [8].

Fig. 1 presents two sample certificates used in this research. The left image shows a certificate related to user interface design, while the right image features a certificate in machine learning. These samples represent the types of documents processed for text extraction and classification.

Fig. 1 Certificate Document Data Sample

C. Data Preprocessing

Data Preprocessing in this research mainly divided into two steps. The first one is extracting text from document-based certificate using OCR and then doing data text cleaning on it. The first step in data preprocessing is converting certificate images into machine-readable text using Optical Character Recognition (OCR). OCR is a technology that extracts text from images or scanned documents, enabling further analysis [9]. In this research, OCR is performed using the PyTesseract library, a Python wrapper for Google's Tesseract engine, which is widely used for its accuracy in recognizing printed text from image-based certificates.

Following OCR, the extracted text undergoes a data cleaning process to improve its quality and consistency [10]. This includes converting text to lowercase, removing punctuation, digits, and special characters, filtering out common stopwords, and applying stemming to reduce words to their base form. These steps are essential to reduce noise and standardize the input, especially since OCR outputs often contain garbled or fragmented text [11] due to layout complexity or image artifacts.

D. Data Denoising using Gemini LLM

Large Language Models (LLMs) such as Gemini are capable of understanding and generating human-like text [12], making them useful for data denoising tasks. In this research, LLMs are used to improve the quality of OCR-extracted text by refining garbled or unclear content while preserving its original meaning. By designing specific prompts, the model is guided to rewrite noisy certificate text into a clearer and more structured format suitable for further processing and classification [13]. To guide the denoising process effectively, this study explores several prompt-based denoising tailored to different text organization approaches. The following are the three prompt designs applied in this research:

1) CSIR

The first prompting approach applies a structured narrative format based on the CSIR strategy. While no existing academic studies explicitly investigate the CLEAR prompt structure, the CSIR structure guiding LLMs through Context, Specific Information, Intent/Goal, and Result has been informally discussed in non-academic sources as a structured approach for text processing [14]. This structure helps the model isolate relevant parts of the certificate content while reducing noise by segmenting the information into meaningful components.

In this study, the CSIR prompt was formulated to provide a structured, role-guided instruction to the LLM, aligning the denoising task with clear semantic boundaries. The Context component informs the model that the input originates from OCR output, potentially containing distortions. The Specific Information section outlines the key certificate elements expected in the text, such as participant names, event titles, institutions, and issuance dates. The Intent/Goal clarifies the task objective rewriting the garbled input into a clean, syntactically correct format without introducing information not found in the original content. Finally, the Result directs the model to produce a coherent, readable version of the certificate.

2) CLEAR

The CLEAR format Challenge (Concise), Limitation (Logical), Effect (Explicit), Action (Adaptive), and Result (Reflective) encourages the model to produce outputs that are concise and coherent according to Leo S. Lo research [15]. It filters unnecessary fragments and enhances readability by ensuring that the cleaned text maintains logical flow and content relevance, which is especially helpful in handling OCR outputs with scattered wording.

The CLEAR prompt structure used in this study was designed to guide the LLM through a well-defined reasoning path by framing the denoising task in terms of Challenge, Limitation, Effect, Action, and Result. The Challenge informs the model of the distortions introduced by the OCR process, while the Limitation highlights the loss of layout and the presence of incomplete or erroneous words. The Effect outlines the impact of these issues on the usability of the text, setting the stage for the Action, which instructs the model to rewrite the input while preserving only the essential certificate information. The Result component then prompts the model to output a grammatically correct and logically structured version of the content.

3) CO-STAR

In the CO-STAR format Context, Objective, Style, Tone, Audience, Response the prompt directs the LLM to interpret the text in a problem-solution structure. This helps the model reconstruct noisy or incomplete text by inferring missing elements and organizing information in a way that mirrors common certificate content patterns. While no academic studies explicitly investigate the CO-STAR prompt structure, a non-academic web article discusses its application in guiding LLMs to reconstruct text using a problem-solution framework to get a more impactful output response [16].

The CO-STAR prompt applied in this study leverages a structured prompt comprising Context, Objective, Style, Tone, Audience, and Response to guide the LLM in reconstructing noisy OCR-extracted certificate text. The Context sets the stage by informing the model of the source and nature of the distorted input, while the Objective clearly defines the task of converting the text into a coherent and accurate form. The Style and Tone

directives enforce a formal and respectful output style consistent with typical certificate phrasing. The Audience parameter instructs the model to produce outputs suitable for subsequent use in machine learning classification workflows. Finally, the Response section provides explicit guidance on what information to retain and how to organize it.

E. Data Augmentation

Data augmentation is a technique commonly used in natural language processing (NLP) to increase the size and diversity of training data without the need for additional manual labeling [17]. This method is particularly valuable when working with limited datasets, as it helps improve the generalizability of machine learning models [18]. One widely adopted augmentation technique is synonym replacement, which involves substituting words in a sentence with their synonyms while preserving the original meaning. The rationale behind using synonym replacement is to expose the model to various linguistic expressions and word choices that convey similar meanings, thereby enhancing the model's ability to generalize across unseen data and reducing the risk of overfitting.

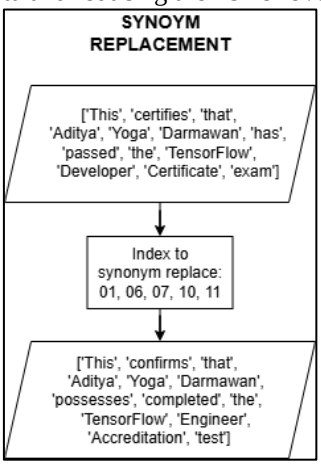

Fig. 2 Synonym Replacement Augmentation Process

An example of synonym replacement used in this study is illustrated in Fig. 2 The original sentence, “This certifies that Aditya Yoga Darmawan has passed the TensorFlow Developer Certificate exam,” undergoes partial transformation. Words such as “certifies,” “has,” “passed,” “Developer,” “Certificate,” and “exam” are replaced with their contextual synonyms, resulting in the augmented sentence: “This confirms that Aditya Yoga Darmawan possesses completed the TensorFlow Engineer Accreditation test.” This example demonstrates how synonym replacement maintains the essential semantic structure while introducing lexical variation. By doing so, the augmented sentence contributes to building a model that can better handle language variability in real-world applications.

F. Machine Learning Training

In this study, the final stage of the pipeline involves training a text classification model using the bert-base-uncased architecture. The objective is to automatically categorize each certificate into one of seven course-related classes based on its textual content: Game Applications, User Interface Design, Computer Networks, Machine Learning, Project Management, Mobile Programming, and Web Programming. After undergoing OCR, preprocessing, denoising, and data augmentation, the cleaned certificate texts serve as the input for fine-tuning BERT in a supervised classification setting [19].

The training process for the BERT-base-uncased model involved fine-tuning over choosen epochs with a learning rate, optimizer, and loss function. The architecture tokenized input sequences, with [CLS] and [SEP] tokens marking sequence boundaries, and processed them through 12 transformer encoder layers. Dropout regularization was applied to the [CLS] token embedding and dense layer to prevent overfitting. The model was implemented via Hugging Face Transformers, integrating denoised and augmented data seamlessly. This setup prioritized efficient learning of domain-specific patterns while balancing computational resources and model performance. The training pipeline was implemented using the Hugging Face Transformers library, enabling seamless integration of the preprocessed data, prompt-refined texts, and augmented samples into the model’s input format. This setup ensured that the BERT model could effectively learn domain-specific patterns from the denoised and augmented data, prioritizing both efficiency and adaptability for real-world certificate classification applications.

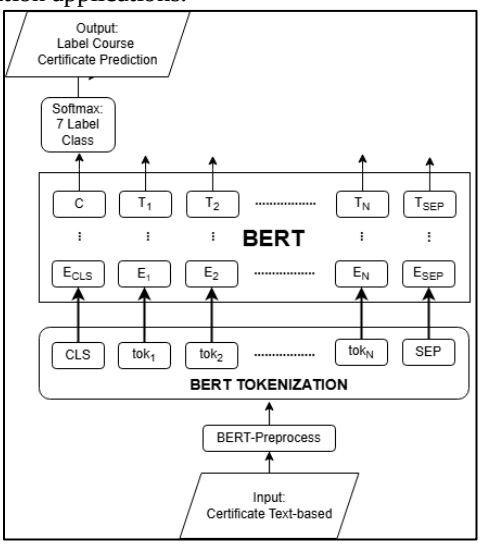

Fig. 3 BERT Architecture Process

As shown in Fig. 3 the training process begins by feeding the cleaned certificate text into a BERT preprocessing module, where it is tokenized into subword units using special tokens such as [CLS] and [SEP] [20]. These tokens are then embedded and passed into the BERT encoder, which processes the sequence bidirectionally. The contextualized output vector corresponding to the [CLS] token is extracted and forwarded to a softmax classifier that maps the representation to one of seven predefined course-related classes. This architecture enables the model to capture both semantic and syntactic features of the certificate text for accurate classification [21].

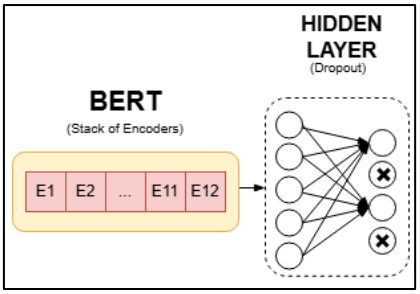

Fig. 4 BERT Model in Classification

As shown in Fig. 4 the BERT model used in this study consists of a stack of 12 transformer encoder layers, each responsible for progressively refining the contextual representation of the input tokens [22]. The final [CLS] token embedding produced by the 12th encoder layer is passed through an additional hidden layer with dropout regularization to prevent overfitting. This hidden representation is then connected to the output layer for classification. Ultimately, the model assigns the certificate text to the most appropriate course-related label out of the seven predefined classes, based on the semantic correlation between the input content and course domains.

G. Training Scenario

This research studies systematic evaluation of the experimental structured prompts, detailing twelve distinct training scenarios designed to assess the impact of structured prompt engineering on classification performance. Each scenario integrates variations in prompt structure (CSIR, CLEAR, CO-STAR), learning rate, and epsilon value, enabling a comprehensive analysis of their interplay in denoising and model optimization.

The discussion further outlines key dataset characteristics, including the original sample size and the expanded training corpus post-augmentation, alongside hardware specifications running the computational pipeline. These parameters are meticulously documented to ensure reproducibility and to contextualize the empirical investigation of prompt-driven denoising efficacy.

TABLE II. TRAINING SCENARIO

No	Structured Prompt Type	Hyperparameters	
		Learning Rate	Epsilon
1	- (without prompt)	5e-5	1e-8
2		3e-5	1e-9
3		7e-5	1e-6
4	CSIR	5e-5	1e-8
5		3e-5	1e-9
6		7e-5	1e-6
7	CLEAR	5e-5	1e-8
8		3e-5	1e-9
9		7e-5	1e-6
10	CO-STAR	5e-5	1e-8
11		3e-5	1e-9
12		7e-5	1e-6

This study systematically evaluates twelve distinct training configurations (as shown in Table II) to assess the interplay between structured prompt engineering, hyperparameter tuning, and classification performance. The scenarios encompass three narrative-based prompt structures (CSIR, CLEAR, and CO-STAR) each paired with three variations of learning rate (5e-5, 3e-5, 7e-5) and epsilon (1e-8, 1e-9, 1e-6). A baseline condition (without prompts) serves as a control. The dataset comprises 75 samples per class across seven certificate categories (total 525 samples), expanded to 150 augmented samples per class via synonym replacement. Experiments were conducted using a T4 GPU on Google Colab, ensuring computational feasibility while isolating the effects of denoising strategies and hyperparameters on model generalization.

III. Results and Discussion

After all scenario-based datasets (including the baseline and those denoised using CSIR, CLEAR, and CO-STAR prompt structures) were preprocessed and augmented, they were trained using the BERT-base-uncased model for five epochs. The training process was conducted with consistent hyperparameters, including a learning rate of 5e-5, epsilon of 1e-8, AdamW optimizer, and categorical cross-entropy loss. Upon completion of training, the model's performance was evaluated using several key metrics: training and validation accuracy, confusion matrix, and a detailed classification report consisting of precision, recall, and F1-score for each of the seven course-related classes.

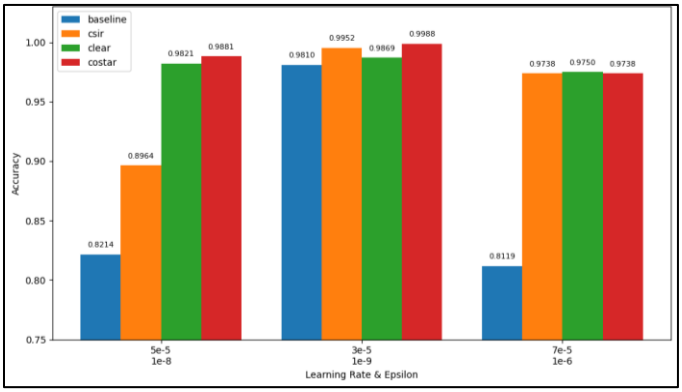

Fig. 5 Accuracy Comparison by Scenario (adjusted by Learning Rate & Epsilon)

As shown in the Fig. 5, the graph illustrates the impact of varying learning rates and epsilon values on classification accuracy across four denoising methods (baseline, CSIR, CLEAR, CO-STAR). At a learning rate of 5e-5 and epsilon 1e-8, all methods exhibit significantly lower performance compared to other configurations, with the baseline achieving only 82.14% accuracy and even CO-STAR peaking at 98.81%. In contrast, the 3e-5/1e-9 configuration yields markedly higher accuracy across all methods, particularly for CSIR (99.52%) and CO-STAR (99.88%), suggesting optimal convergence under these hyperparameters. The 7e-5/1e-6 setting shows inconsistent results, with baseline accuracy dropping to 81.19% while structured prompt methods (CSIR, CLEAR, CO-STAR) achieve ~97–98% accuracy.

Given the low performance of the 5e-5/1e-8 combination, this specific hyperparameter pairing is selected for in-depth

analysis in the subsequent section. Its results will serve as a critical case study to investigate failure modes through detailed evaluation metrics (accuracy, validation accuracy, confusion matrix, and classification report) in the next section to investigate root causes such as suboptimal gradient updates, inadequate noise mitigation, or limitations in the denoising pipeline. This analysis aims to establish the minimum viable efficacy of the proposed approach under subpar training conditions.

A. *Accuracy*

The evaluation of model performance in this study includes a detailed analysis of classification accuracy across each denoising scenario guided by distinct prompt structures. The figure below illustrates the comparative accuracy achieved throughout the training process.

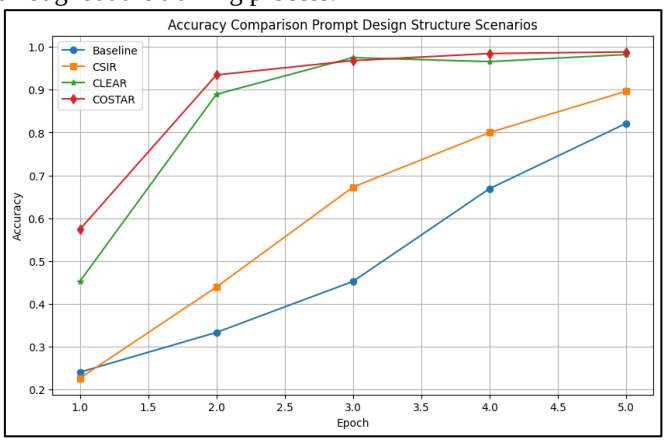

Fig. 6 Accuracy Comparison Prompt Design Structure

As shown in Fig. 6, the accuracy comparison across different prompt design structures demonstrates significant performance improvements over the baseline model throughout the five training epochs. As illustrated in the figure, The baseline model, trained without noise reduction or augmentation, achieves 82.14% accuracy by the fifth epoch. In contrast, all three LLM-guided prompt scenarios consistently outperform the baseline, with CSIR showing moderate improvement, reaching 89.64%, while CLEAR and CO-STAR exhibit rapid convergence and higher final accuracies of 98.21% and 98.81%, respectively.

Among the scenarios, CO-STAR not only achieves the highest accuracy overall but also demonstrates superior learning stability across epochs. CLEAR also performs reliably, achieving near-peak accuracy as early as the third epoch, indicating efficient representation learning after noise reduction. CSIR, although initially lagging behind, catches up significantly by the final epoch, underscoring its effectiveness as a denoising strategy. These results confirm that structured prompt-based denoising significantly improves classification accuracy for noisy OCR text.

B. *Validation Accuracy*

In addition to training accuracy, validation accuracy serves as a critical metric to evaluate the generalization capability of the model across different denoising strategies. The following figure presents the validation accuracy trends over the training epochs for each scenario.

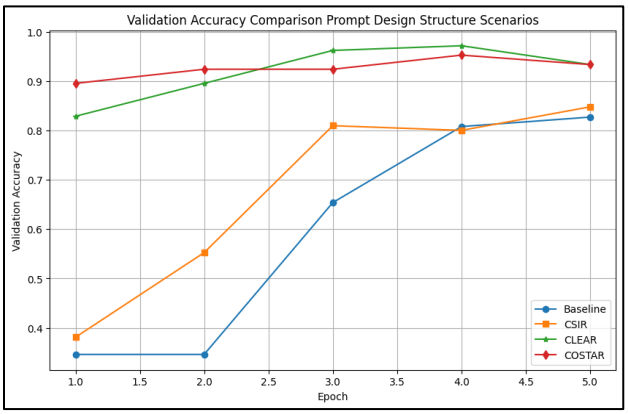

Fig. 7 Validation Accuracy Comparison Prompt Design Structures

As shown in Fig. 7, the validation accuracy results across all prompt-based denoising scenarios consistently outperform the baseline, particularly from the third epoch onward. CO-STAR yields the highest validation accuracy at early stages, maintaining its performance throughout the training process with only slight variations. CLEAR follows closely, reaching its peak at epoch four before experiencing a marginal decline in the final epoch. This trend suggests that both CO-STAR and CLEAR not only accelerate convergence but also enhance generalization when dealing with noisy OCR-derived text data.

In contrast, the baseline and CSIR scenarios demonstrate a more gradual improvement, with validation accuracy steadily increasing across epochs but failing to reach the performance levels achieved by CLEAR and CO-STAR. The final epoch shows that CO-STAR achieves 93.33% validation accuracy, slightly outperforming CLEAR at the same stage, while CSIR and baseline plateau at 84.76% and 82.69%, respectively. These findings indicate that employing structured prompt-based denoising prior to classification can significantly enhance model generalization capability on unseen data, especially when using well-formulated narrative structured prompts.

C. *Confusion Matrix*

A comprehensive analysis of classification outcomes was conducted through the use of confusion matrices, which provide a granular view of model performance across the seven predefined class labels. These visualizations reveal how well the model distinguishes between different course-related labels and where misclassifications are most prevalent.

In the provided confusion matrices Fig. 8 (a)-(d), rows correspond to true labels and columns to predicted labels, with class categories ordered as: Game Applications, User Interface Design, Computer Networks, Machine Learning, Project Management, Mobile Programming, Web Programming.

As depicted in Fig. 8 (a)-(b), the baseline model exhibits misclassification between Computer Networks (true) and Machine Learning (predicted, 2/15 errors) due to shared technical jargon, while Project Management (true) is confused with Game Applications (2/15 errors), likely from ambiguous contextual cues. The CSIR-guided model Fig. 8 (b) resolves

these errors for Game Applications and User Interface Design (15/15 accuracy) but struggles with Web Programming (5 misclassified as Mobile Programming), indicating residual noise in cross-platform terminology.

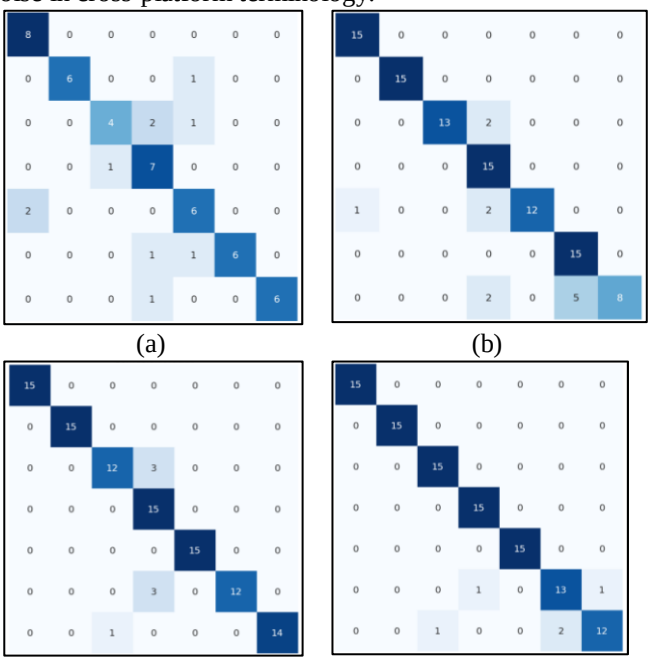

Fig. 8 Confusion Matrixs across all Prompt Design Structure

Fig. 8 (c)-(d) show CLEAR and CO-STAR further refine performance. CLEAR Fig. 8 (c) reduces Computer Networks errors to 3/15 and achieves perfect Machine Learning classification. CO-STAR Fig. 8 (d) eliminates Computer Networks errors entirely and attains 15/15 accuracy for Project Management, though minor confusion persists between Web Programming (true) and Mobile Programming (predicted, 2 errors), likely from lexical overlaps like "responsive design." These results how structured prompts progressively resolve semantic ambiguities, with CO-STAR's problem-solution prompt structure optimizing class distinction.

The confusion matrix analysis revealed recurring misclassifications between Web Programming and Mobile Programming categories, particularly in scenarios where models predicted Web Programming instances as Mobile Programming Fig. 8 (d). This overlap may stem from two primary factors. First, during data collection, keyword-based scraping of certificate content could inadvertently conflate. terms such as "responsive design" or "cross-platform development," which are relevant to both domains. Second, technical synergies in tooling and programming languages such as JavaScript/Dart frameworks (e.g., React Native/Flutter) used for both web and mobile applications likely introduced ambiguity in distinguishing domain-specific features during classification.

D. Classification Report

To further evaluate the model's performance across individual classes, a classification report was observed, summarizing key metrics such as precision, recall, and F1-score for each denoising scenario.

TABLE III. CLASSIFICATION REPORT

Scenario	Classification Report (*Weighted Average*)		
	Precision	***Recall***	***F1-Score***
Baseline	0.84	0.81	0.81
CSIR	0.91	0.89	0.88
CLEAR	**0.95**	0.93	0.93
CO-STAR	**0.95**	**0.95**	**0.95**

As shown in Table III, the weighted average values for precision, recall, and F1-score across the four scenarios demonstrate a substantial improvement in classification performance when structured prompt-based denoising is applied. The CO-STAR scenario exhibits the highest overall scores, achieving 0.95 in all three metrics, indicating highly consistent predictive capability across all classes. Similarly, CLEAR achieves strong performance with a precision of 0.95, recall of 0.93, and F1-score of 0.93, suggesting effective balance between identifying relevant instances and minimizing false positives or negatives.

In contrast, the baseline scenario (trained without any noise reduction or augmentation) yields lower metrics across the board, with a weighted precision of 0.84 and both recall and F1-score at 0.81. The CSIR scenario shows moderate improvement over the baseline, reaching a precision of 0.91 and recall of 0.89, reflecting the benefit of structured cleaning, albeit to a lesser degree than CO-STAR or CLEAR. These results reinforce the conclusion that prompt design structure plays a pivotal role in enhancing text quality prior to model training, ultimately leading to more accurate and reliable classification outcomes.

IV. CONCLUSION

This study has demonstrated the effectiveness of prompt-based denoising using structured narrative prompts in improving the performance of text classification models trained on noisy OCR-extracted certificate data. Through comparative experimentation using BERT-base-uncased and four different data conditions—baseline (no denoising), and three prompt design structures (CSIR, CLEAR, and CO-STAR)—it was observed that the CO-STAR scenario consistently outperformed the others across all evaluation metrics. The CO-STAR-enhanced dataset achieved the highest accuracy and validation accuracy throughout all epochs, with a peak accuracy of 0.9881 and validation accuracy of 0.9524. Confusion matrix analysis further confirmed the increased model reliability, with clearer class distinctions in the CO-STAR and CLEAR scenarios compared to the baseline. Additionally, the classification report showed that CO-STAR attained the best weighted average precision, recall, and F1-score, all at 0.95, followed closely by CLEAR. These findings confirm that applying narrative-based

prompt structures for denoising noisy text data significantly enhances model generalization and predictive performance. Although the results are promising, this research has limitations. A significant constraint lies in the reliance on a free-tier large language model (LLM), specifically the Gemini model, to perform prompt-based denoising. While the model improved text clarity post-OCR, its free-tier limitations including strict rate limits (15 requests per minute and 1,500 daily requests), latency due to shared infrastructure, and a knowledge cutoff in August 2024 would likely constrained the depth of text refinement. Furthermore, the OCR process, implemented using PyTesseract, can still produce imperfect textual outputs, especially when dealing with low-resolution or poorly formatted certificate scans, as prompt-based refinement cannot always fix layout-related errors.

To address these limitations, future research should consider utilizing more powerful LLMs, either open-source alternatives with greater contextual understanding or commercial-grade models that offer advanced prompt responsiveness and broader linguistic capabilities. Such improvements have the potential to significantly elevate the quality of the preprocessed data and, in turn, enhance model performance. In parallel, integrating more sophisticated OCR pipelines or hybrid denoising methods (such as combining LLM-based prompt engineering with transformer-based reconstruction) could yield cleaner textual representations and further reduce noise from challenging document inputs.

REFERENCES

[1] S. Kumar Garai, O. Paul, U. Dey, S. Ghoshal, N. Biswas, and S. Mondal, "A Novel Method for Image to Text Extraction Using Tesseract-OCR," *American Journal of Electronics & Communication*, vol. 3, no. 2, pp. 8–11, Oct. 2022.

[2] N. Nuryani, R. Munir, A. Purwarianti, and D. Puji Lestari, "BERT-Based Model and LLMs-Generated Synthetic Data for Conflict Sentiment Identification in Aspect-Based Sentiment Analysis," *Interdisciplinary Journal of Information, Knowledge, and Management*, vol. 20, p. 004, 2025.

[3] A. Hemmer, J. Brachat, M. Coustaty, and J.-M. Ogier, "Estimating Post-OCR Denoising Complexity on Numerical Texts," 2023, pp. 67–79.

[4] Y. Sonoda *et al.*, "Structured Clinical Reasoning Prompt Enhances LLM's Diagnostic Capabilities in Diagnosis Please Quiz Cases," Sep. 03, 2024.

[5] H. Lu *et al.*, "QAIE: LLM-based Quantity Augmentation and Information Enhancement for few-shot Aspect-Based Sentiment Analysis," *Inf Process Manag*, vol. 62, no. 1, p. 103917, Jan. 2025.

[6] A. Elnashar, J. White, and D. C. Schmidt, "Enhancing structured data generation with GPT-4o evaluating prompt efficiency across prompt styles," *Front Artif Intell*, vol. 8, Mar. 2025.

[7] D. M. Thomas and S. Mathur, "Data Analysis by Web Scraping using Python," in *2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA)*, IEEE, Jun. 2019, pp. 450–454.

[8] S. Rennie, M. Buchbinder, E. Juengst, L. Brinkley-Rubinstein, C. Blue, and D. L. Rosen, "Scraping the Web for Public Health Gains: Ethical Considerations from a 'Big Data' Research Project on HIV and Incarceration," *Public Health Ethics*, vol. 13, no. 1, pp. 111–121, Apr. 2020.

[9] Y. He, "Research on Text Detection and Recognition Based on OCR Recognition Technology," in *2020 IEEE 3rd International Conference on Information Systems and Computer Aided Education (ICISCAE)*, IEEE, Sep. 2020, pp. 132–140.

[10] A. Kim, C. Pethe, N. Inoue, and S. Skiena, "Cleaning Dirty Books: Post-OCR Processing for Previously Scanned Texts," in *Findings of the Association for Computational Linguistics: EMNLP 2021*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2021, pp. 4217–4226.

[11] K. Maekawa, Y. Tomiura, S. Fukuda, E. Ishita, and H. Uchiyama, "Improving OCR for Historical Documents by Modeling Image Distortion," 2019, pp. 312–316.

[12] Z. Yang, A. Ishay, and J. Lee, "Coupling Large Language Models with Logic Programming for Robust and General Reasoning from Text," Jul. 2023.

[13] A. Webson and E. Pavlick, "Do Prompt-Based Models Really Understand the Meaning of Their Prompts?," in *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2022, pp. 2300–2344.

[14] Narendra Babu Mundlamuri, "A Comprehensive Guide to Prompt Engineering," https://dzone.com/articles/a-comprehensive-guide-to-prompt-engineering.

[15] L. S. Lo, "The CLEAR path: A framework for enhancing information literacy through prompt engineering," *The Journal of Academic Librarianship*, vol. 49, no. 4, p. 102720, Jul. 2023.

[16] Frugal Zentennial, "Unlocking the Power of COSTAR Prompt Engineering: A Guide and Example on converting goals into system of actionable items," https://medium.com/@frugalzentennial/unlocking-the-power-of-costar-prompt-engineering-a-guide-and-example-on-converting-goals-into-dc5751ce9875.

[17] L. F. A. O. Pellicer, T. M. Ferreira, and A. H. R. Costa, "Data augmentation techniques in natural language processing," *Appl Soft Comput*, vol. 132, p. 109803, Jan. 2023.

[18] I. Okimura, M. Reid, M. Kawano, and Y. Matsuo, "On the Impact of Data Augmentation on Downstream Performance in Natural Language Processing," in *Proceedings of the Third Workshop on Insights from Negative Results in NLP*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2022, pp. 88–93.

[19] L. Shi, D. Liu, G. Liu, and K. Meng, "AUG-BERT: An Efficient Data Augmentation Algorithm for Text Classification," 2020, pp. 2191–2198.

[20] M. A. Shah, M. J. Iqbal, N. Noreen, and I. Ahmed, "An Automated Text Document Classification Framework using BERT," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 3, 2023.

[21] Y. Yang and X. Cui, "Bert-Enhanced Text Graph Neural Network for Classification," *Entropy*, vol. 23, no. 11, p. 1536, Nov. 2021.

[22] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," Oct. 2018, [Online].