Enhancing Arabic Automated Essay Scoring with Synthetic Data and Error Injection

Chatrine Qwaider,[1] Bashar Alhafni,[1,2] Kirill Chirkunov,[1]
Nizar Habash,[1,2] Ted Briscoe[1]
[1]MBZUAI, [2]New York University Abu Dhabi
{chatrine.qwaider,kirill.chirkunov,ted.briscoe}@mbzuai.ac.ae
{alhafni,nizar.habash}@nyu.edu

Abstract

Automated Essay Scoring (AES) plays a crucial role in assessing language learners' writing quality, reducing grading workload, and providing real-time feedback. The lack of annotated essay datasets inhibits the development of Arabic AES systems. This paper leverages Large Language Models (LLMs) and Transformer models to generate synthetic Arabic essays for AES. We prompt an LLM to generate essays across the Common European Framework of Reference (CEFR) proficiency levels and introduce and compare two approaches to error injection. We create a dataset of 3,040 annotated essays with errors injected using our two methods. Additionally, we develop a BERT-based Arabic AES system calibrated to CEFR levels. Our experimental results demonstrate the effectiveness of our synthetic dataset in improving Arabic AES performance. We make our code and data publicly available.[1]

1 Introduction

Automated Essay Scoring (AES) is a technology that automates the evaluation and scoring of essays to assess language learners' writing quality while eliminating the need for human intervention (Shermis and Burstein, 2003). AES has gained great interest due to its significant benefits in the field of education (Lagakis and Demetriadis, 2021; Susanti et al., 2023). AES systems help teachers evaluate many essays with consistent scoring and reduced workload. On the other hand, AES helps students improve their writing quality through rapid real-time scoring and feedback (Hahn et al., 2021).

Unlike for English, it is difficult to develop robust and scalable AES systems for Modern Standard Arabic (MSA), primarily due to the lack of essay datasets necessary for building effective Arabic AES (Lim et al., 2021; Elhaddadi et al., 2024). This paper presents a framework to tackle the issue of data scarcity and quality by utilizing Transformers and Large Language Models (LLMs) to generate and build a synthetic dataset.

Our approach begins with prompting GPT-4o to generate a variety of Arabic essays covering multiple topics and different writing proficiency levels as defined by the Common European Framework of Reference (CEFR) (Council of Europe, 2001). Subsequently, we use a controlled error injection model to introduce errors into the correct Arabic essays, ensuring that erroneous essays reflect the type of errors that are commonly made by learners of Arabic in real-world scenarios. Our error injection approach consists of two steps: (i) *Error Type Prediction*, where a fine-tuned CAMeLBERT MSA model (Inoue et al., 2021) classifies the most likely error type for each word, and (ii) *Error Realization*, where we apply a bigram MLE model to determine the most probable transformation for each predicted error type. Our framework enables the generation of realistic human-like essays, enhancing data augmentation for Arabic AES systems.

Our main contributions are as follows:

- Proposing a framework based on LLMs and Transformers for augmenting Arabic essays that accurately reflect human writing patterns.
- Creating a synthetic Arabic AES dataset with 3,040 essays annotated with CEFR proficiency levels.
- Developing an Arabic AES system using a BERT-based model, enabling accurate and scalable evaluation of Arabic essays based on CEFR standards.

The rest of the paper is organised as follows: §2 reviews related work on AES, §3 describes the dataset, and §4 outlines our data augmentation approach. §5 details the error injection methods, followed by an evaluation in §6. We discuss our results in §7 and §8 presents the conclusion and future work.

[1] https://github.com/mbzuai-nlp/arabic-aes-bea25

Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications, pages 549–563

2 Related Work

AES has been investigated extensively, particularly in English (Lim et al., 2021; Ramesh and Sanampudi, 2022), where multiple tools have been introduced such as IntelliMetric (Elliott et al., 2003), e-rater (Attali and Burstein, 2006), Grammarly,[2] Write and Improve[3] (Yannakoudakis et al., 2018), and others. The development of English AES systems has been enabled by large scale annotated datasets such as the First Cambridge English (FCE) dataset[4] (Yannakoudakis et al., 2011), Automated Student Assessment Prize (ASAP) dataset,[5] the TOEFL11 corpus (Blanchard, 2013), and the ICLE (International Corpus of Learner English) (Granger, 2003). These datasets contain thousands of student essays with proficiency level grades, often along multiple dimensions.

In contrast, Arabic AES research has received less attention. Some studies have applied feature engineering and machine learning to develop models (Alghamdi et al., 2014; Al-Shalabi, 2016; Alobed et al., 2021; Gaheen et al., 2021), but they partially address key challenges, especially the scarcity of large, publicly available annotated datasets for improving Arabic writing quality.

Ghazawi and Simpson (2024) introduced AR-AES, a benchmark of 2,046 undergraduate essays from three university faculties, annotated by two educators per faculty using rubrics to assess academic performance. In contrast, our work focuses on writing proficiency, using the CEFR standard.

Bashendy et al. (2024) presented QAES, the first publicly available trait specific annotations for Arabic AES. QAES extends the Qatari Corpus of Argumentative Writing (QCAW) (Ahmed et al., 2024), which consists of 195 Arabic argumentative essays. They implemented multi-layered annotation of traits such as coherence, organization, grammar, and others. Despite its comprehensive annotation, it is small in size and limited to two prompts. While QAES multi-traits scores are publicly available, the QCAW holistic score is not.

Habash and Palfreyman (2022) presented the Zayed University Arabic-English Bilingual Undergraduate Corpus (ZAEBUC). This corpus comprises non-parallel essays in Arabic and English related to three prompts collected from first-year university students with differing writing proficiency. ZAEBUC includes 216 annotated Arabic essays featuring manual annotations for syntactic and morphological characteristics and a CEFR-based proficiency assessment. Again, ZAEBUC is small in size and limited to three prompts.

Researchers have explored data augmentation methods like sampling, noise injection, and paraphrasing to address data scarcity and quality (Li et al., 2022). The recent development of LLMs has paved the way for researchers to explore promising new data synthesis solutions (Wang et al., 2024a; Long et al., 2024). Transformers and LLMs can closely mirror real-world distributions while introducing valuable variations across multiple tasks and domains (Wang et al., 2024b).

GPT models have shown strong capabilities in generating synthetic essays for English AES (Ramesh and Sanampudi, 2022). LLMs and Transformers have also generalized well in Arabic NLP tasks, including Question Answering (Samuel et al., 2024), Code Switching (Alharbi et al., 2024), NER (Sabty et al., 2021), Grammatical Error Correction (Alhafni and Habash, 2025; Solyman et al., 2023), and Sentiment Analysis (Refai et al., 2023). However, to the best of our knowledge no research has utilized such models to generate Arabic essays across CEFR writing proficiency levels.

3 Data: The ZAEBUC Corpus

For all our experiments, we use the ZAEBUC corpus (Habash and Palfreyman, 2022). ZAEBUC comprises essays written by native Arabic speakers, which were manually corrected and annotated for writing proficiency using the CEFR (Council of Europe, 2001) rubrics and scale. Each essay was annotated by three CEFR-proficient bilingual speakers. Habash and Palfreyman (2022), assigned a holistic CEFR level to each essay by converting the three CEFR ratings into numerical scores (ranging from 1 to 6) and then taking the rounded average. The essays in the corpus were limited to three prompt choices on *Social Media*, *Tolerance*, and *Development*; see Table 1. We use the splits created by Alhafni et al. (2023). Table 2 shows the CEFR level distribution of the ZAEBUC corpus based on holistic CEFR scores. The ZAEBUC corpus is limited in size and skewed toward B1–B2 levels, with no A1 or C2 essays. This common imbalance in Arabic learner data motivated our synthetic approach to create a more balanced CEFR distribution.

[2] https://app.grammarly.com/
[3] https://writeandimprove.com/
[4] https://ilexir.co.uk/datasets/index.html
[5] https://www.kaggle.com/c/asap-aes

وسائل التواصل الاجتماعي وتأثيرها على الفرد والمجتمع. How do social media affect individuals and society?
كيف نعزز ثقافة التسامح في المجتمع؟ How can the UAE promote a culture of tolerance in society?
التطور الحضاري الذي تشهده دولة الإمارات العربية المتحدة What do you think are the most important developments in the UAE at the moment?

Table 1: The prompts given to the essay writers in the ZAEBUC corpus (Habash and Palfreyman, 2022).

CEFR Level	Count	Percentage
A1	0	0%
A2	7	3%
B1	110	51%
B2	80	37%
C1	11	5%
C2	0	0%
Unassessable	6	3%
Total	214	100%

Table 2: ZAEBUC corpus CEFR level distributions.

Topic	B	I	A
Culture and Traditions	1	3	2
Daily Life	2	2	2
Education	3	6	8
Environment	2	2	3
Future	1	2	2
History and Culture	2	2	2
Hobbies	3	2	2
Imaginary	5	2	2
Life/Time Management	4	4	2
Personal Experiences	7	2	2
Relations	4	2	2
School Life	4	2	2
Sport and Health	2	3	1
Technology and Media	2	8	6
Travel and Experience	1	2	1
Politics and Government	2	2	7
Social Issues	2	7	6
Total	47	53	52

Table 3: Count of Arabic text prompts by level and topic. B: Beginner level (A1, A2), I: Intermediate level (B1, B2), A: Advanced level (C1, C2).

4 Synthetic Data Augmentation

We propose a synthetic data augmentation approach leveraging the ZAEBUC dataset to generate synthetic essays that align with CEFR rubrics and have features similar to human text. The pipeline utilizes three phases: Building Essay Prompts, Feature Profiling, and finally Data Augmentation.

4.1 Building Essay Prompts

We began by compiling a diverse set of essay prompts across various categories and CEFR levels. While not directly drawn from established frameworks, our prompts were inspired by themes common in language assessments, including placement tests and academic writing. We aimed to cover familiar and level-appropriate topics, such as social issues, education, and personal experiences, while ensuring balance across the CEFR bands. We considered three proficiency levels: Beginner (A1–A2), Intermediate (B1–B2), and Advanced (C1–C2). General themes, such as hobbies, suited all levels, while more complex topics, including politics, Technology, and Education, were reserved for advanced learners.

Using LLMs like GPT-4o[6], Gemini[7], and Copilot[8], we generated 100 prompts, followed by a manual review to remove redundancies and ensure both relevance for Arabic essay writing and balanced proficiency coverage. The final collection consists of 152 balanced and diverse prompts. Table 3 presents the selected categories and the distribution of the prompts across levels, while Table 4 provides example prompts for the Hobbies category.

4.2 Feature Profiling

We construct linguistic profiles for each CEFR level using the ZAEBUC corpus. Each profile contains various levels of linguistic information. Representing different lexical and syntactic features, we use the number of words/sentences (N_w,N_s), the number of tokens/vocabulary (N_v), words/sentences lengths (L_w,L_s), and sentence complexity measured by syntactic tree depth (D_s).

We define the lexical diversity (Type-Token Ratio, TTR) as:

$$\text{TTR} = \frac{\text{Unique Tokens}}{\text{Total Tokens}} \quad (1)$$

[6] https://openai.com/index/hello-gpt-4o/
[7] https://gemini.google.com/app
[8] https://copilot.microsoft.com/

Level	Arabic Prompt	English Prompt
Beginner	• ما هي هوايتك المفضلة؟	• What is your favorite hobby?
	• تحدث عن نشاط تحبه في عطلة نهاية الأسبوع.	• Talk about a weekend activity you love.
	• ما هي الرياضة التي تحب ممارستها؟ لماذا؟	• What is your favorite sport? Why?
Intermediate	• ما هي هوايتك المفضلة وكيف بدأت في ممارستها؟ لماذا تحبها؟	• What is your favorite hobby and how did you start practicing it? Why do you enjoy it?
	• هل ترغب في تعلم هواية جديدة؟ ما هي ولماذا تهمك؟	• Do you wish to learn a new hobby? What is it and why does it interest you?
Advanced	• ناقش كيف تؤثر الهواية على صحتك النفسية والجسدية.	• Discuss how hobbies impact your mental and physical health.
	كيف يمكن أن تكون الهوايات وسيلة للتعبير عن الذات؟	How can hobbies serve as a means of self-expression?
	• هل هناك هواية جديدة ترغب في تجربتها؟	• Is there a new hobby you wish to try? Discuss the reasons
	ناقش الأسباب التي تجعلك مهتمًا بها وكيف تعتقد أنها ستفيدك.	you are interested in it and how you believe it will benefit you.

Table 4: Examples of prompts related to the topic of Hobbies and classified into one of three different levels.

Similarly, we calculate the sentence complexity by:

$$C_s = \frac{\sum_{i=1}^{N} D_i}{N} \tag{2}$$

where D_i is the syntactic depth of sentence i and N is the total number of sentences.

For morphological features, we use the ZAEBUC morphological annotations: the most frequent POS tags, such as nouns, verbs, adjectives, etc.

We aggregate all extracted features across the essays to get a quantitative representation at different writing CEFR levels, which serves as a reference for later stages.

4.3 Zero-shot Data Augmentation

Effective LLM prompt engineering has become increasingly important, as the model's output varies based on the prompt, provided instructions, and prompt language. Previous studies in Arabic NLP have shown that using English as the instruction language for input prompts can improve output quality (Kmainasi et al., 2024; Koto et al., 2024).

In our approach, we experiment with various prompts for zero-shot data augmentation to identify those that produce human-like text while adhering to guideline instructions. We use GPT-4o as our generation model due to its affordability and larger token capacity for both input and output. The GPT prompts include **(a)** the target CEFR level, **(b)** CEFR guidelines and instructions, **(c)** the linguistic profile for the targeted CEFR level to control the prompt output, and **(d)** the topic prompt or question from the previously mentioned topic prompts dataset. For these missing levels (A1 and C2), instead of injecting a pre-defined profile, GPT-4o was directly prompted to act as an assistant and generate data based on the general standards and rubrics of the CEFR.

To check the quality of the generated essays and whether they follow the prompt instructions, we build a linguistic feature profile (vector) for each augmented essay. We then assess the alignment between the generated essays and the reference CEFR-level profiles by computing their feature vectors' cosine similarity as in equation 3. Specifically, given two real-valued feature vectors P_i (the CEFR reference profile) and Q_i(the generated essay), the cosine similarity is calculated as:

$$\cos(\theta) = \frac{\sum_i P_i Q_i}{\sqrt{\sum_i P_i^2} \cdot \sqrt{\sum_i Q_i^2}} \tag{3}$$

This metric ensures that the synthetic data closely aligns with real human essay patterns. Based on the computed similarity score, we assign a predicted CEFR level to each essay.

Later, we calculate the alignment between the predicted CEFR level and the target level specified in the GPT-4o prompt (ground truth) using the following agreement formula:

$$\text{Agreement} = \frac{\sum_{i=1}^{n} (\hat{y}_i = y_i)}{n} \tag{4}$$

where $\hat{y}_i$ is the predicted level and y_i is the ground truth. This process evaluates how well GPT-4o succeeded in aligning the generated content with the intended proficiency level, serving as a measure of agreement rather than a prediction from an external model.

We conducted multiple rounds of prompt engineering refinements to improve the quality of the generated Arabic essays and ensure alignment with CEFR levels.

CEFR Level	Count	Percentage
A1	470	15.5%
A2	470	15.5%
B1	530	17.4%
B2	530	17.4%
C1	520	17.1%
C2	520	17.1%
Total	3,040	100%

Table 5: The generated corpus CEFR level distributions.

First, we found that straightforward prompts without explicit controlled linguistic instructions and explanations resulted in incoherent essays, including irrelevant topics and English text, achieving only 20.5% matching agreement with linguistic feature profiles. In a subsequent round, we introduced detailed definitions of linguistic features and restricted outputs to Arabic-only text, which improved agreement to 26%. However, the model still occasionally produced incomplete essays and injected text from the prompt into the essay.

The most effective prompt structure format is illustrated in Figure 1. We separated system-level control instructions from user-defined parameters, thereby providing clearer guidance for structured and proficiency-aligned text generation. This refinement increased agreement to 27.5%, demonstrating that precisely controlled instructions enhance LLM performance in structured writing tasks.

Ultimately, we generated 3,040 Arabic essays covering all CEFR levels and various topics, where each prompt was used to create ten essays. This effort was intentionally designed to address the imbalanced CEFR distribution in the original ZAEBUC corpus, where B-level essays were overrepresented. By constructing a more balanced synthetic dataset, we aimed to enhance model performance across the full proficiency spectrum. The structured and controlled prompt design also improved alignment with learner writing styles while providing a consistent framework for generating realistic Arabic essays. Table 5 presents the distribution of generated essays across different CEFR levels. The full dataset statistics are provided in Appendix A.1.

5 Error Injection

Human-generated text naturally contains some grammatical errors and linguistic infelicities. In order to create human-like essays, we need to add similar kinds of errors to the synthetic essays that reflect the level of writing attainment. In this phase, we prompt GPT-4o to inject errors into the previously generated essays while maintaining their aligned CEFR levels by utilizing error profiling.

```
{"role": "system", "content": "You are a helpful assistant that
generates essays in Arabic.Try to make them different, focus on other
aspects and ideas,DO NOT generate anything from the prompt about the
level or the features. Generate the Essays only in Arabic and make sure
you generate completed sentences."},

 {"role": "user", "content": f"Generate an Arabic essay talking about
prompt: {prompt_text} for CEFR level {cefr_level} based on the CEFR
Guidelines: {guidelines}. You have to follow the linguistics features
profile as {linguistics_profile}. Here is the feature explainataion
:{features_guidelines} "}
```

Figure 1: GPT-4o prompts messages that have been used to generate Arabic essays

5.1 Error Profiling

Error Distribution Profiles To model the distribution of errors to inject into the synthetic essays, we again leverage the ZAEBUC corpus, which contains the erroneous essays aligned with the manually corrected ones. We followed the same methodology we used to construct the linguistic feature profiles for each CEFR level to develop error distribution profiles aligned with CEFR levels. The error profile captures and reflects the authentic distribution patterns observed in human writing at different CEFR levels.

Developing an Error Instruction Repository To prompt GPT-4o to generate essays containing errors we applied the Grammatical Error Detection (GED) model proposed by (Alhafni et al., 2023) to the ZAEBUC corpus to annotate errors using 13 error tags and to obtain error distributions for each CEFR level. We created the repository using the error tags, where we also added a formal definition of what those tags describe in terms of linguistic errors. In addition, we expanded the error taxonomy by splitting it into finer-grained classes. Each error instruction is followed by an example showing the correct word and the erroneous version. The explanation was based on the *extended ALC taxonomy* (Alfaifi et al., 2013), which was refined later and introduced as ARETA (Belkebir and Habash, 2021). Appendix C presents examples of the error types. Figure 2 shows some examples from our error instruction repository.

5.2 GPT-Based Error Injection

We prompted GPT-4o to inject errors into the synthetically generated essays based on the error distri-

```
"REPLACE_0":
"Orthographical Error for example : Use incorrect  Ya and Alif-Maqsura
forms at the end of words, replacing 'ى' with 'ي' or 'ي' with 'ى',
such as using 'على' instead of 'علي'.",
"Orthographical Error for example : Swap the order of two adjacent
characters in words to create orthographic errors, e.g., 'تقريبا' instead
of 'تربيا'.",
"Orthographical Error for example : Introduce a common error by
lengthening a vowel sound in words, for instance, changing 'نقم' to
'نقيم'.",
"Orthographical Error for example : Confuse 'ة' with 'ه' at the end of
words, creating common spelling errors, like 'مشاركة' becoming 'مشاركه'.",
"Orthographical Error for example : Omit or misuse Alif Fariqa, such
as writing 'وكانوا' instead of 'وكانو'.",
. . . . . .
```

Figure 2: An example of orthographical error instructions from the developed errors instructions repository

```
{"role": "system", "content": "You are a helpful
assistant that injected erroneous tokens in Arabic
essay based on given error instruction"},

{"role": "user", "content":   f"Rewrite the following
Arabic essay with the {cefr} CEFR level, following the
specified error Instruction without removing, changing
or fixing any existing mistakes.\n\n"
f"Essay in Arabic:\n{essay}\n\n"
f"Error Instruction: Please Inject exactly
{num_words_to_inject} error/s of this type:
{random_error_prompt}}\n\n"
"Only apply the specified errors directly in the text
without any introductory or additional comments."}
```

Figure 3: Sample GPT-4o error injection prompt

For each error tag:

- Randomly select error prompts from `Error instruction repoistory` based on the average error count:
  - Average count `<5`: Use **1** `Error instruction`.
  - Average count between `(5,10)`: Use **2** `Errors instructions`
  - Average count `>10`: Use **3** `Errors instructions`.
- Inject errors into the essay according to the determined distribution.

Figure 4: Error injection based on average error count

bution profiles while maintaining the CEFR level. The model processed one essay at a time in a zero-shot setting, except that we included the definition and explanation of the error tags. For example, **M** indicates a morphological error, while **Merge** targets two mistakenly split tokens that need to be merged, and so on.

After conducting multiple experiments, we observed the following issues: (i) The model struggled to follow the predefined error distribution perhaps due to the complexity of the prompts. (ii) The model was confused by certain error tags, particularly **Split** and **Merge**. These errors were mainly ignored in the injected text. (iii) We calculated the cosine similarity between the main error profile and the injected essays' error distributions as shown in Equation 3. When we injected all errors at once, the similarity agreements did not exceed 20%; however, when we reduced the number of error tags per essay the agreements significantly improved, reaching 86%.

Therefore we implemented a method where each error type was injected separately. This required multiple iterations over the same essay, corresponding to the number of error tags shown in the error distribution profile for each CEFR level. Figure 3 shows an example of a GPT-4o prompt for error injection. Some error types, especially orthographic errors, are more frequent among Arabic writers than others. The prompt was intentionally designed through prompt engineering. The 'helpful assistant' component establishes a cooperative persona for the LLM, while the subsequent instruction to 'inject erroneous tokens' explicitly guide GPT-4o towards the specific task of error introduction. This approach ensures that GPT-4o is not making random edits but is rather following predefined instructions to create targeted errors, aligning with the overall goal of generating realistic synthetic data.

To reflect this, we randomly select weighted error instructions based on the average frequency of each error type. Figure 4 shows the pseudocode for the selection process. The full pseudocode is in Appendix B.

5.3 Controlled Error Injection

We introduce a controlled method for injecting errors into clean text, ensuring that the resulting erroneous sentences follow the empirical error distributions observed at each CEFR level. More formally, given an input sentence (X) and its CEFR level (L), we introduce errors in two steps: Error Type Prediction and Error Realization.

Error Type Prediction We estimate the probability of an error type occurring at a given word, i.e., $P(error\_type|word)$. To do so, we leverage ARETA in a reverse annotation process where we process correct–erroneous sentence pairs, tagging each correct word with its corresponding error type. Using this annotated data, we train a token-level BERT classifier to predict the most likely error type for each word in a given correct sentence. We fine-tune CAMeLBERT MSA (Inoue et al., 2021) to build our classifier.

Error Realization To determine how a word should be corrupted, we first align correct–erroneous sentence pairs using the algorithm proposed by Alhafni et al. (2023). For each aligned pair, we extract edit transformations that capture the operations required to convert a correct word into its erroneous

counterpart. Using this data, we estimate $P(transformation|error\_type)$ with a bigram Maximum Likelihood Estimation (MLE) lookup model: $count(transformation, error\_type)$ / $count(error\_type)$. During inference, we apply the BERT classifier to predict error types for each word in a sentence. We then filter these predictions, retaining only error types relevant to the sentence's CEFR level. Finally, the MLE model selects the most probable corruption for a given error type. A complete example of a B1–level essay generated by the proposed model is in Figure 5.

6 Experimental Setup

This study focuses on introducing a data augmentation framework and synthetic Arabic essay corpus, rather than proposing a new AES model. We use a BERT-based model trained on the original ZAEBUC dataset as the reference baseline, evaluating how different augmentation strategies (e.g., GPT-4o generation, BERT-based error injection) improve performance relative to this setup.

6.1 Data and Metrics

We use the ZAEBUC dataset for all the experiments, following the splits created by Alhafni et al. (2023): 70% Train, 15% Dev, and 15% Test.

Our primary evaluation metric is Quadratic Weighted Kappa (QWK) (Cohen, 1968), the most widely used metric in AES research (Ke and Ng, 2019). We also report accuracy, macro precision (P), recall (R), and F_1 scores. Model predictions are evaluated in two settings settings: average-refence and multi-reference. The average-reference setting uses the rounded average of the three scores as the gold label, while the multi-reference considers each of the three human-assigned labels as a valid reference during evaluation, following a more tolerant evaluation strategy (§3).

6.2 Model

We treat the task of AES as a text classification problem. We fine-tune CAMeLBERT MSA (Inoue et al., 2021) on the training split of ZAEBUC. The models were trained by using the average CEFR gold labels. During training, we ignore the essays that are labeled as Unassessable, but we penalize the models for missing them in the evaluation. We fine-tune the models for 5 epochs, with a maximum sequence length of 512, a learning rate of 5e-5, and a batch size of 32.

6.3 Results

Our results are presented in Table 6. Our baseline system, only trained on the ZAEBUC training set, indicates room for improvement, with the F_1 at 24.50% and QWK at 22.44%. We then switched between different datasets to measure the impact of data augmentation on the model.

Impact of Synthetic Data We tested data augmentation by adding 3,040 corrected GPT-4o-generated essays, which lowered QWK but increased F_1. Notably, the multi-reference setting saw significant gains, with QWK at 96.00% and F_1 at 92.32%. This pattern stems from the flexibility of multi-reference evaluation, which treats all three human-assigned CEFR labels as valid references. This accommodates natural scoring variations and increases the chance that model predictions, especially on synthetic data, align with at least one reference label, boosting QWK and F_1 scores for both GPT-generated and error-injected essays.

Comparison of Error Injection Methods As the initial synthetic essays were error-free, we further refined the model by adding essays with human-like errors. We compared two methods from §5: (1) GPT-based error injection (with and without instruction examples) and (2) the controlled BERT-based method.

The results demonstrate that the controlled error model improves performance in all metrics, particularly in the average reference setting, which achieved 27.87 % and 38.02 for QWK and F_1, respectively. This result aligns with expectations, as the BERT-injected errors closely follow CEFR-based error distributions, producing errors that realistically reflect learner writing and better match the average of human ratings.

GPT-based error injection performed best in the multi-reference setting, with one-shot examples reaching 96.47% QWK and zero-shot boosting F_1 to 95.12%. While less aligned with CEFR profiles, GPT errors benefit from fluency and variability, increasing the chance of matching at least one human reference in this flexible evaluation.

7 Discussion

This study demonstrated the effectiveness of synthetic data and controlled error injection in enhancing Arabic AES, providing key insights into metric interpretation, data expansion, and methodological choices.

Level	B1	**Topic**	Hobbies
Prompt	هل ترغب في تعلم هواية جديدة؟ ما هي ولماذا تهمك؟		
	Do you wish to learn a new hobby? What is it and why does it interest you?		
Correct Essay			
تعلم هواية جديدة يمكن أن يكون تجربة مثيرة ومفيدة في حياتنا. في الوقت الحالي، أفكر في تعلم الرسم كمهارة جديدة. الرسم ليس فقط وسيلة للتعبير عن النفس، ولكنه أيضًا يساعد في تحسين التركيز والصبر. عندما أرسم، أجد نفسي أركز على التفاصيل الصغيرة، مما يساعدني على تحسين مهارات الملاحظة لدي. الرسم يعطيني الفرصة للتعبير عن مشاعري وأفكاري بطريقة بصرية. أحيانًا، تكون الكلمات غير كافية للتعبير عما نشعر به، وهنا يأتي دور الفن. بالإضافة إلى ذلك، تعلم الرسم يمكن أن يكون وسيلة رائعة للاسترخاء بعد يوم طويل من العمل أو الدراسة. الجلوس أمام لوحة بيضاء وتحويلها إلى قطعة فنية يمكن أن يكون تجربة مهدئة ومريحة. أيضًا، الرسم يمكن أن يفتح لي أبوابًا جديدة للتفاعل مع الآخرين. يمكنني الانضمام إلى ورش عمل أو مجموعات فنية، حيث يمكنني مقابلة أشخاص يشاركونني نفس الاهتمام. هذا يمكن أن يساعدني في بناء شبكة اجتماعية جديدة وتعلم تقنيات جديدة من الآخرين. بشكل عام، تعلم الرسم كهواية جديدة يهمني لأنه يجمع بين التعبير الفني، تحسين الذات، والتفاعل الاجتماعي. إنه تحدٍ جديد أود أن أواجهه، وأنا متحمس لاكتشاف ما يمكنني تحقيقه من خلال هذه الهواية.			
Erroneous Essay			
ت علم هواية جديدة يمكن أن يكون تجربة مثيرة ومفيده في حياتنا . في الوقت الحالي ، أفك ر في تعلو الرسم كمهارة جديدة , الرسم ليس بس وسيلة للتعبير عن النفس ، ولكنة ايضا يساعد في تحسين التركيز و الصبر . عندما ارس , اجد نفسي أركز على التفاصيل الصغيره ، مما يساعدن على تحسين مهارات الملاحظة لدي . الرسم يعطينى الفرصة للتعبير عن مشاعري و أفكاري بطريقه بصرية . احيانا ، تكون الكلمات غير كافية للتعبير عما نشعر به وهنا ياتي دور الفن . بالاضافة إلى ذلاك ، تعلم الرسم يمكن أن تكون وسيلة رائعة للإستر خاء بعد يوم طويل من العمل أو الدراسة جلوس أمام لوحة بيضاء و تحويلها الى قطعة فنية يمكن ان تكون تجربة مهدئة ومرتحة . أيضا ، الرسم يمكن ان بيفتح ي ابوابا جديد لتفاعل مع الآخرين . يمكنني إلانضمام الى ورش عمل أو مجموعات فنية ، حيث يمكنني مقابلة اشخاص يشاركيننى نفس الاهتمام . هذا بمكن أن يساعدنى في بناء شبكة إجتماعية جديدة و تعلم تقنيات جديده من الآخرين بشكل عام ، تعلم الرسم كهواية جديدة يهمني لانه يجمع بين التعبير الفني ، تحسين الذات ، والتفاعل الاجتماعي , انه تحد جديد أود ان أواجهه ، وأنا متحم لاكتشاف ما يمكنني تحقيقه من خلال هذه الهواية .			

Figure 5: An Example of a B1 Arabic Essay generated by GPT-4o using the Hobbies prompt and the same essay after injecting errors by the controlled BERT-based model.

First, we emphasize that QWK offers a more robust metric than accuracy for evaluating AES systems, particularly under imbalanced class distributions. Unlike accuracy, which is biased by majority classes, QWK penalizes errors by their ordinal distance from the correct label. As Table 6 shows, even modest QWK improvements indicate meaningful advancements in differentiating CEFR levels, a distinction especially relevant given the skewed ZAEBUC dataset.

The significant gains observed in the multi-reference setting with generated GPT-4o essays stem from its flexibility. This evaluation approach treats all three human-assigned CEFR labels as valid references, accommodating natural scoring variations and increasing the chance that model predictions align with at least one reference label.

Our analysis revealed that while GPT-4o is powerful for generating diverse content, it struggles to precisely follow the nuanced distribution and specific linguistic features, including error patterns, observed in the manually annotated ZAEBUC dataset. In the GPT-based error injection approach, error type selection is guided by average error counts from the ZAEBUC corpus, but error realization depends on GPT-4o's interpretation of the prompt, making it less predictable. This inherent challenge in mimicking human-like linguistic and error distributions through zero-shot generation directly contributed to the observed lower agreement rate.

In contrast, the controlled method employs a BERT-based classifier for error prediction and ap-

Train Data	Average Reference					Multi-Reference				
	QWK	Acc	F_1	P	R	QWK	Acc	F_1	P	R
ZAEBUC (baseline)	22.44	57.58	24.50	23.33	26.76	61.06	84.85	43.70	42.50	45.31
ZAEBUC + GPT essays	14.92	60.61	26.43	25.55	27.45	96.00	96.97	92.32	98.04	88.89
ZAEBUC + BERT errors	**27.87**	57.58	**38.02**	35.86	44.93	82.70	87.88	71.66	70.83	74.38
ZAEBUC + GPT errors_1	17.14	57.58	25.64	25.18	26.27	**96.47**	96.97	94.16	97.92	91.67
ZAEBUC + GPT errors_0	20.84	57.58	32.76	31.53	46.08	93.79	93.94	**95.12**	96.49	94.44

Table 6: Performance comparison of different training datasets. GPT essays are the original correct essays generated from GPT-4o, BERT errors are the erroneous essays using the controlled injection BERT model, GPT errors_1 are the erroneous essays using GPT-4o with one-shot error example, while GPT errors_0 with Zero-shot settings.

plies transformations using bigram-MLE. This systematic approach resulted in a more robust replication of empirically observed error patterns, leading to its superior performance in the average-reference setting. This is expected, as BERT-injected errors more closely resemble learner writing and align more closely with average human ratings.

Overall, our findings highlight a trade-off between error alignment and fluency in data augmentation. Controlled error injection excels in the average-reference setting due to its closer alignment with learner errors, while GPT-based augmentation benefits from multi-reference flexibility but less reliably replicates authentic errors. The controlled BERT-based method thus serves as a key component of our pipeline, effectively addressing the limitations of direct GPT error injection.

Qualitative Analysis The qualitative analysis of the generation process revealed various biases in the GPT-4o outputs, including cultural, gender, and ideological biases. For instance, the essays frequently referenced traditional Arabic themes, reinforced stereotypical gender roles, and reflected culturally narrow assumptions. A clear example of religious bias is that الجمعة 'Friday' was selected as *the favorite day* in all 20 generated essays. Additionally, there was a noticeable tendency to use masculine forms throughout the texts. Such biases may unintentionally disadvantage students whose writing reflects different experiences, perspectives, or identities. Examples of these biases, along with their frequencies, are provided in Appendix A.2. We also observed a lack of diversity among the ten essays generated per prompt, with GPT-4o often repeating similar lexical and structural patterns.

8 Conclusions and Future Work

This paper presents a hybrid framework for Arabic AES, using LLMs and transformers to tackle data scarcity by generating synthetic essays that partly replicate Arabic learner writing. Building on the ZAEBUC corpus, we developed CEFR-aligned linguistic and error profiles and used GPT-4o to produce 3,040 essays across 152 prompts. However, GPT-4o's performance relies heavily on prompt engineering, achieving only 27.5% alignment with our reference profiles.

To introduce errors, we compare our two methods: (1) GPT-4o prompted multi-step error injection, and (2) our controlled method fine-tuning the CAMeLBERT MSA model to inject errors proportionally to their profiled occurrence.

Evaluated with a fine-tuned BERT classifier, our hybrid framework, combining GPT-generated data with controlled error injection, outperformed the baseline (QWK: 27.87%, F_1: 38.02%), offering more reliable and interpretable results. These findings demonstrate the effectiveness of controlled error injection in capturing learner error distributions across CEFR levels.

For future work, we will prioritize integrating a human evaluation into our framework. Human annotators will assess the fluency and naturalness of synthetic essays, as well as the realism of injected errors, ensuring that they reflect typical learner patterns at specific CEFR levels.

To improve generalizability, we also plan to expand the diversity of prompts beyond predefined topics and incorporate a wider set of writing traits, including coherence, logical flow, and topic relevance, beyond syntactic and lexical features.

We also intend to enhance CEFR-level modelling by incorporating more manually annotated essays. This will help capture nuanced linguistic variations across levels and increase the robustness of our dataset. Lastly, we aim to deploy the AES system as an interactive tool to provide users with instant feedback on errors and proficiency levels.

Limitations

Despite the effectiveness of our hybrid Arabic AES framework, we note several limitations related to the quality of generated Arabic essays, error injection accuracy, and the generalization of the AES model. The lack of A1 and C2 essays in ZAEBUC means that there is no gold reference data for these levels, which may impact both linguistic and error profiles, affecting the accuracy of GPT-generated essays. Furthermore, different biases are present in both the ZAEBUC dataset and GPT-4o outputs as discussed in (§7)

In addition, due to the lack of comprehensive gold data, GPT struggles to fully replicate real learner writing styles, achieving only 27.5% agreement with linguistic feature profiles.

Another limitation is the model's ability to generalize across various domains and question types. The AES system may struggle with broader writing tasks and alternative prompts since the dataset and augmentation methods focus on predefined prompts. Relying solely on CEFR as a holistic scoring method limits interpretability. Enhancing the dataset with multi-trait annotations, such as coherence, argumentation, and organization, could improve scoring accuracy and feedback quality. Moreover, better-controlled GPT prompting could refine the quality and diversity of generated essays, reducing biases and improving alignment with real learner writing patterns.

Due to resource constraints, human evaluation was not feasible in this study; however, we plan to engage CEFR-trained annotators in the future.

Ethical Considerations

While Arabic AES systems provide significant support in assessing Arabic learners' writing proficiency, it is essential to highlight the ethical implications of their use. Automatic assessment and scoring may lead to misjudgments that could distress learners and students, especially if their work is incorrectly evaluated. AES tools should serve as an educational assistive technology, complementing the teacher's judgment, not replacing it in educational settings.

References

Abdelhamid M Ahmed, Xiao Zhang, Lameya M Rezk, and Wajdi Zaghouani. 2024. Building an annotated l1 arabic/l2 english bilingual writer corpus: The qatari corpus of argumentative writing (qcaw). *Corpus-Based Studies across Humanities*, 1(1):183–215.

Emad Fawzi Al-Shalabi. 2016. An automated system for essay scoring of online exams in arabic based on stemming techniques and levenshtein edit operations. *arXiv preprint arXiv:1611.02815*.

Abdullah Alfaifi, Eric Atwell, and Ghazi Abuhakema. 2013. Error annotation of the arabic learner corpus: A new error tagset. In *Language Processing and Knowledge in the Web: 25th International Conference, GSCL 2013, Darmstadt, Germany, September 25-27, 2013. Proceedings*, pages 14–22. Springer.

Mansour Alghamdi, Mohamed Alkanhal, Mohamed Al-Badrashiny, Abdulaziz Al-Qabbany, Ali Areshey, and Abdulaziz Alharbi. 2014. A hybrid automatic scoring system for arabic essays. *Ai Communications*, 27(2):103–111.

Bashar Alhafni and Nizar Habash. 2025. Enhancing text editing for grammatical error correction: Arabic as a case study. *Preprint*, arXiv:2503.00985.

Bashar Alhafni, Go Inoue, Christian Khairallah, and Nizar Habash. 2023. Advancements in Arabic grammatical error detection and correction: An empirical investigation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6430–6448, Singapore. Association for Computational Linguistics.

Sadeen Alharbi, Reem BinMuqbil, Ahmed Ali, Raghad AlOraini, Saiful Bari, Areeb Alowisheq, and Yaser Alonaizan. 2024. Leveraging llm for augmenting textual data in code-switching asr: Arabic as an example. *Proc. SynData4GenAI*.

Mohammad Alobed, Abdallah MM Altrad, and Zainab Binti Abu Bakar. 2021. An adaptive automated arabic essay scoring model using the semantic of arabic wordnet. In *2021 2nd International Conference on Smart Computing and Electronic Enterprise (ICSCEE)*, pages 45–54. IEEE.

Yigal Attali and Jill Burstein. 2006. Automated essay scoring with e-rater® v. 2. *The Journal of Technology, Learning and Assessment*, 4(3).

May Bashendy, Salam Albatarni, Sohaila Eltanbouly, Eman Zahran, Hamdo Elhuseyin, Tamer Elsayed, Walid Massoud, and Houda Bouamor. 2024. Qaes: First publicly-available trait-specific annotations for automated scoring of arabic essays. In *Proceedings of The Second Arabic Natural Language Processing Conference*, pages 337–351.

Riadh Belkebir and Nizar Habash. 2021. Automatic error type annotation for Arabic. In *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 596–606, Online. Association for Computational Linguistics.

D Blanchard. 2013. Toefl11: A corpus of non-native english. *Educational Testing Service*.

Jacob Cohen. 1968. Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit. *Psychological bulletin*, 70(4):213.

C. o. E. Council of Europe. 2001. Common european framework of reference for languages: learning, teaching, assessment.

Chima Elhaddadi, Imad Zeroual, and Anoual El Kah. 2024. Automatic arabic essays scoring: A scoping review. In *International Conference on Arabic Language Processing*, pages 38–48. Springer.

S Elliott, MD Shermis, and J Burstein. 2003. Overview of intellimetric. *Automated Essay Scoring: A Cross-Disciplinary Perspective*, pages 67–70.

Marwa M Gaheen, Rania M ElEraky, and Ahmed A Ewees. 2021. Automated students arabic essay scoring using trained neural network by e-jaya optimization to support personalized system of instruction. *Education and Information Technologies*, 26:1165–1181.

Rayed Ghazawi and Edwin Simpson. 2024. Automated essay scoring in arabic: a dataset and analysis of a bert-based system. *arXiv preprint arXiv:2407.11212*.

Sylviane Granger. 2003. The international corpus of learner english: a new resource for foreign language learning and teaching and second language acquisition research. *Tesol Quarterly*, 37(3):538–546.

Nizar Habash and David Palfreyman. 2022. ZAEBUC: An annotated Arabic-English bilingual writer corpus. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 79–88, Marseille, France. European Language Resources Association.

Marcelo Guerra Hahn, Silvia Margarita Baldiris Navarro, Luis De La Fuente Valentín, and Daniel Burgos. 2021. A systematic review of the effects of automatic scoring and automatic feedback in educational settings. *IEEE Access*, 9:108190–108198.

Go Inoue, Bashar Alhafni, Nurpeiis Baimukan, Houda Bouamor, and Nizar Habash. 2021. The interplay of variant, size, and task type in Arabic pre-trained language models. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 92–104, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.

Zixuan Ke and Vincent Ng. 2019. Automated essay scoring: A survey of the state of the art. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 6300–6308. International Joint Conferences on Artificial Intelligence Organization.

Mohamed Bayan Kmainasi, Rakif Khan, Ali Ezzat Shahroor, Boushra Bendou, Maram Hasanain, and Firoj Alam. 2024. Native vs non-native language prompting: A comparative analysis. In *International Conference on Web Information Systems Engineering*, pages 406–420. Springer.

Fajri Koto, Haonan Li, Sara Shatnawi, Jad Doughman, Abdelrahman Sadallah, Aisha Alraeesi, Khalid Almubarak, Zaid Alyafeai, Neha Sengupta, Shady Shehata, Nizar Habash, Preslav Nakov, and Timothy Baldwin. 2024. ArabicMMLU: Assessing massive multitask language understanding in Arabic. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 5622–5640, Bangkok, Thailand. Association for Computational Linguistics.

Paraskevas Lagakis and Stavros Demetriadis. 2021. Automated essay scoring: A review of the field. In *2021 International Conference on Computer, Information and Telecommunication Systems (CITS)*, pages 1–6. IEEE.

Bohan Li, Yutai Hou, and Wanxiang Che. 2022. Data augmentation approaches in natural language processing: A survey. *Ai Open*, 3:71–90.

Chun Then Lim, Chih How Bong, Wee Sian Wong, and Nung Kion Lee. 2021. A comprehensive review of automated essay scoring (aes) research and development. *Pertanika Journal of Science & Technology*, 29(3):1875–1899.

Lin Long, Rui Wang, Ruixuan Xiao, Junbo Zhao, Xiao Ding, Gang Chen, and Haobo Wang. 2024. On llms-driven synthetic data generation, curation, and evaluation: A survey. *arXiv preprint arXiv:2406.15126*.

Dadi Ramesh and Suresh Kumar Sanampudi. 2022. An automated essay scoring systems: a systematic literature review. *Artificial Intelligence Review*, 55(3):2495–2527.

Dania Refai, Saleh Abu-Soud, and Mohammad J Abdel-Rahman. 2023. Data augmentation using transformers and similarity measures for improving arabic text classification. *IEEE Access*, 11:132516–132531.

Caroline Sabty, Islam Omar, Fady Wasfalla, Mohamed Islam, and Slim Abdennadher. 2021. Data augmentation techniques on arabic data for named entity recognition. *Procedia Computer Science*, 189:292–299.

Vinay Samuel, Houda Aynaou, Arijit Chowdhury, Karthik Venkat Ramanan, and Aman Chadha. 2024. Can LLMs augment low-resource reading comprehension datasets? opportunities and challenges. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, pages 307–317, Bangkok, Thailand. Association for Computational Linguistics.

Mark D Shermis and Jill C Burstein. 2003. *Automated essay scoring: A cross-disciplinary perspective*. Routledge.

Aiman Solyman, Marco Zappatore, Wang Zhenyu, Zeinab Mahmoud, Ali Alfatemi, Ashraf Osman Ibrahim, and Lubna Abdelkareim Gabralla. 2023. Optimizing the impact of data augmentation for low-resource grammatical error correction. *Journal of King Saud University-Computer and Information Sciences*, 35(6):101572.

Meilia Nur Indah Susanti, Arief Ramadhan, and Harco Leslie Hendric Spit Warnars. 2023. Automatic essay exam scoring system: A systematic literature review. *Procedia Computer Science*, 216:531–538.

Ke Wang, Jiahui Zhu, Minjie Ren, Zeming Liu, Shiwei Li, Zongye Zhang, Chenkai Zhang, Xiaoyu Wu, Qiqi Zhan, Qingjie Liu, et al. 2024a. A survey on data synthesis and augmentation for large language models. *arXiv preprint arXiv:2410.12896*.

Yidong Wang, Qi Guo, Wenjin Yao, Hongbo Zhang, Xin Zhang, Zhen Wu, Meishan Zhang, Xinyu Dai, Min Zhang, Qingsong Wen, et al. 2024b. Autosurvey: Large language models can automatically write surveys. *arXiv preprint arXiv:2406.10252*.

Helen Yannakoudakis, Øistein E Andersen, Ardeshir Geranpayeh, Ted Briscoe, and Diane Nicholls. 2018. Developing an automated writing placement system for esl learners. *Applied Measurement in Education*, 31(3):251–267.

Helen Yannakoudakis, Ted Briscoe, and Ben Medlock. 2011. A new dataset and method for automatically grading esol texts. In *Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies*, pages 180–189.

A GPT-4o Generated Essays

A.1 Statistics

CEFR	#Essays	#Words	#Sentences	#Tokens	Avg_W_L	Avg_S_L	Unique_Tokens	Unique_Words
A1	470	39367	6721	47612	4.66	6.08	4518	4512
A2	470	54980	5565	63769	4.82	10.46	7147	7143
B1	530	100185	7276	113672	4.97	14.62	10742	10734
B2	530	127029	7804	142995	5.03	17.32	12522	12509
C1	520	136849	7630	152962	5.16	19.05	12832	12823
C2	520	146253	7946	163461	5.16	19.57	13686	13676
Total	**3040**	**604663**	**42942**	**684471**	**5.04**	**14.94**	**27286**	**27272**

Table 7: Summary statistics of the generated Arabic synthetic essay corpus across CEFR levels. #Essays denotes the number of essays; #Words refers to the total word count; #Sentences indicates the total number of sentences; #Tokens represents the total number of tokens (vocabulary items); Avg_W_L corresponds to the average word length in characters; Avg_S_L refers to the average sentence length in words; lexical diversity is captured through the counts of unique tokens and unique words.

A.2 Bias in the Generated Essays

Arabic Prompt	English Prompt	Arabic Response	English Response	Occurrences /20	Bias
ما هو يومك المفضل؟	What is your favorite day?	الجمعة	Friday	20	Cultural/Religious Bias
ماهو طعامك المفضل؟	What is your favorite food?	البيتزا	Pizza	16	Globalization Bias
ما هي هوايتك المفضلة؟	What is your favorite hobby?	القراءة	Reading	10	Socioeconomic/Class Bias
رحلة ذهبت إليها	A trip you went on	ذهبت إلى البحر	I went to the beach	18	Geographical/Cultural Bias
ماذا تفعل في عطلة نهاية الأسبوع؟	What do you do on the weekend?	نذهب إلى الحديقة	We go to the park	20	Geographical/Cultural Bias
ماهي المادة الدراسية المفضلة؟	What is your favorite school subject?	الرياضيات	Mathematics	17	Educational System Bias
ما هي رياضتك المفضلة؟	What is your favorite sport?	كرة القدم	Football	20	Cultural Bias
شخص تعتبره مثلك الأعلى	A person you consider your role model	والدي	My father	17	Gender Bias
من هو أفضل صديق لك؟	Who is your best friend?	أحمد	Ahmed	20	Gender/Name Bias
المهنة المستقبلية	Your future profession	طبيب	Doctor	13	Stereotype Bias
معلم شهير في العالم الأدبي	A famous teacher in the world of literature	الأهرامات	The Pyramids	17	Cultural Bias
أفضل فصول السنة	Your favorite season of the year	الصيف	Summer	20	Climate Bias
لغة تود تعلمها	A language you would like to learn	الأسبانية	Spanish	16	Language Bias
بلد ترغب في السفر إليه	A country you would like to visit	مصر	Egypt	10	National Identity Bias
قوة خارقة تتمناها	A superpower you wish to have	الطيران	Flying	19	Media Bias

Table 8: Examples of response biases in GPT-4o generated essays.

B GPT-4o Error Injection Algorithm

```
Algorithm: Inject_Errors_and_Verify

Inject Errors into Essays
Input: Augemented_Essays (3040 essays from GPT-4o), Error_instructions, CEFR_Error_Profiles
Output: Errerouns_Essays

FOR each Essay in Generated_Essays:
   # Determine target CEFR level's error distribution
   Target_CEFR = Get_CEFR_Level(Essay)

   # Retrive the Error Profile
    Error_Profile = Get_Error_Profile(Target_CEFR)

   # Inject errors based on error distribution
   FOR each Error_Type in Error_Profile:

        # Select error instruction prompts based on average error count
      Avg_Error_Count = Get_Avg_Error_Count(Error_Type)

      IF Avg_Error_Count < 5:
         Num_Prompts = 1
      ELSE IF 5 ≤ Avg_Error_Count ≤ 10:
         Num_Prompts = 2
      ELSE:
         Num_Prompts = 3

      Selected_Prompts = Select_Random_Prompts(Error_Prompts.json, Error_Type,
Num_Prompts)

      # Inject errors according to determined distribution
      Essay = Inject_Errors(Essay, Selected_Prompts)



Verify Error Injection
FOR Errerouns_Essays:
   # Apply Grammar Error Detection (GED) to identify errors
   Detected_Errors = Apply_GED(Injected_Essay)

   # Recalculate the error distribution for injected essays
   Injected_Error_Profile = Calculate_Error_Distribution(Detected_Errors)

   # Compare the injected error distribution with the target CEFR profile
   Similarity_Score = Cosine_Similarity(Injected_Error_Profile, Target_CEFR_Error_Profile)

End Algorithm
```

C Error Types Taxonomy

	11-Classes	42-Classes	Error Description	Correct Word	Erroneous Word
Morphology (M)	M	MI	Inflection	عارف	معروف
		MT	Tense	ذهب	يذهب
Orthography (O)	O	OA	Alef-Maqsura	القاضي	القاضى
		OA+OH	Alef-Maqsura + Hamza	أضحى	اضحا
		OA+OR	Alef-Maqsura + Wrong Character	كشيء	كشىء
		OC	Chatacter Order	المدرسة	المردسة
		OD	Extra Character	هذا	هاذا
		OD+OG	Extra Character + Lengthening Short Vowels	تتطورو	تتطور
		OD+OH	Extra Character + Hamza	لأنهم	ألانهم
		OD+OM	Extra Character + Missing Character	الاجتماعي	الاجتاعيي
		OD+OR	Extra Character + Wrong Character	الصور	السوور
		OH	Hamza	العب	إلعب
		OH+OM	Hamza + Missing Character	الأشياء	الاشيا
		OH+OT	Hamza + Ta-Marbuta	إمارة	اماره
		OM	Missing Character	المدرسة	المدسة
		OM+OR	Missing Character + Wrong Character	المجتمع	المجطمع
		OR	Wrong Character	المدرسة	المدرصة
		OR+OT	Wrong Character + Ta-Marbuta	مكتظة	مكتضه
		OT	Ta-Marbuta	غرفة	غرفه
		OW	Alef-Fariqa	كتبوا	كتبو
Semantics (S)	S	SF	Conjunction	فسبحان	سبحان
		SW	Word Selection	على	من
Punctuation (P)	P	P	Punctuation	السوق،	السوق.
Syntax (X)	X	XC	Case	رائعا	رائع
		XC+XG	Case + Gender	مجتهدا	مجتهدة
		XC+XN	Case + Number	نواح	نواحي
		XF	Definiteness	المفيد	مفيد
		XG	Gender	كان	كانت
		XM	Missing Word	على	NULL
		XN	Number	كتابين	كتب
		XT	Unnecessary Word	NULL	على
Combination (Comb.)	M+O	MI+OH	Inflection + Hamza	أشخاص	اشخاصك
	O+X	OH+XC	Hamza + Case	أضرارا	اضرار
SPLIT	SPLIT	SPLIT	Split	دولة الإمارات	دولةالإمارات
MERGE	MERGE	MERGE	Merge	بالعلم	ب العلم
DELETE	DELETE	DELETE	Delete	NULL	داخل

Table 9: Illustrative examples of error types categorized according to the ARETA error taxonomy (Belkebir and Habash, 2021). The table presents hierarchical mappings from coarse-grained (11-Class) to fine-grained (42-Class) error categories, alongside representative corrections.