Journal of Mathematical Modeling
Vol. -, No. -, 2026, pp. -. Research Article

JMM

LLM-based synthetic data augmentation for Persian sentiment classification

Saeed Ziadid[†], Mahdieh Hasheminezhad[†*], Seyed Abolghasem Mirroshandel[§], Arash Yousefi[§]

[†]*Department of Computer Science, Faculty of Mathematical Sciences, Yazd University, Yazd, Iran*
[§]*Department of Computer Engineering, Faculty of Engineering, University of Guilan, Rasht, Iran*

Email(s): pziadid@gmail.com, hasheminezhad@yazd.ac.ir, mirroshandel@guilan.ac.ir, arash.r400@gmail.com

Abstract. Persian sentiment analysis faces significant challenges due to limited labeled data. Furthermore, the effectiveness of LLM-based synthetic data augmentation has not been systematically evaluated in this setting. We address this gap by leveraging DeepSeek-V3 and GPT-4o mini to generate synthetic samples across three distinct baseline configurations derived from SentiPers. Nine transformer architectures (Persian-specific and multilingual) were evaluated across three random seeds under a unified optimization framework. Rather than yielding uniform gains across all scenarios, the empirical utility of this technique proved highly task-dependent. While binary classification exhibited an apparent performance ceiling with marginal variations, five-class classification demonstrated substantial practical benefits, with synthetic expansion yielding observed performance gains in up to all nine model architectures. Rigorous empirical evaluation and intrinsic lexical audits confirm that targeted data injection enriches fine-grained lexical diversity, providing an effective low-resource strategy for resolving complex sentiment boundaries.

Keywords: Persian sentiment analysis; language models; machine learning.
AMS Subject Classification 2010: 68T50, 68T05.

1 Introduction

With the growing expansion of social media, e-commerce platforms, and online communication, vast amounts of opinion-rich textual data are being generated daily. Analyzing this data has become essential for understanding user sentiment and supporting decision-making in various domains such as recommendation systems, marketing strategies [19], and public opinion analysis [26]. Consequently,

*Corresponding author
Received: 01 February 2026/ Revised: 03 November 2026/ Accepted: 04 November 2026
DOI: 0.22124/jmm.2026.32772.2992

© 2026 University of Guilan

Corrected Proof

sentiment analysis has emerged as a main task in Natural Language Processing (NLP), particularly at the sentence level, where each sentence is assigned a polarity label based on its expressed sentiment [43].

Despite significant progress in this field, the performance of sentiment analysis systems heavily depends on the availability of large, high-quality annotated datasets [2], [3], [17]. This dependence presents a major challenge for low-resource languages [25] such as Persian. As noted by Hosseini et al. [16], the Persian language contends with a paucity of linguistic resources, and scholarly investigations in this area remain notably scarce. This scarcity is particularly evident in sentiment analysis, where the largest publicly available Persian corpus (SentiPers) comprises only 5,561 labeled sentences, substantially smaller than typical English benchmarks. Therefore, effective data augmentation strategies are necessary for advancing Persian sentiment analysis. To address this rarity, various data augmentation techniques have been explored.

Traditional data augmentation techniques in NLP include synonym replacement, back-translation, and random word insertion. Recently, Large Language Models (LLMs) such as DeepSeek and GPT-4o mini have been used for generating synthetic sentences. Unlike conventional techniques that merely alter existing sentences, LLMs can produce new and coherent paraphrases that remain semantically aligned with the original dataset while preserving the intended sentiment label. This flexibility allows generating linguistically diverse examples while maintaining task-relevant semantics. Recent research on sentiment shifts in LLM-rephrased text [32] confirms that advanced models exhibit strong sentiment fidelity across iterative rephrasings, supporting the feasibility of using LLMs for sentiment-preserving data augmentation. As we will show in Section 4, our empirical results further demonstrate that LLM-generated samples can effectively improve Persian sentiment classification.

The specific gap addressed by this study is the lack of a systematic evaluation of LLM-based data augmentation for Persian sentiment classification across multiple models and task settings. Previous work on Persian sentiment augmentation has primarily relied on traditional techniques such as back-translation and class balancing [35], without systematic investigation of LLM-based generation. In contrast to prior studies that evaluate LLM augmentation in isolation or on a limited set of models, this work simultaneously investigates binary and five-class sentiment classification and examines whether LLM-generated samples can complement traditional techniques such as back-translation. Our motivation is premised on two fundamental observations regarding current methodologies. First, conventional augmentation techniques typically rely on modifying existing training instances, a process that can be insufficient for addressing significant data scarcity. Second, cross-lingual transfer strategies frequently encounter domain mismatch and cultural misalignment, which may impede classification accuracy. We investigate LLM-based synthetic generation as a mechanism to produce high-quality, label-preserving Persian sentences, thereby offering a more viable path to improving sentiment analysis in low-resource settings. Using pre-existing SentiPers-based corpora [16], [35] and their LLM-augmented counterparts, we fine-tune nine transformer-based models under a unified experimental pipeline. The results indicate task-dependent performance gains, suggesting that LLM-based augmentation is a promising direction for low-resource NLP settings.

The structure of the remainder of this paper is as follows: Section 2 reviews related work in sentiment classification, data augmentation, and the Transformer architecture. Section 3 details the materials and methods, covering the dataset, augmentation process, and model configurations. Section 4 reports and analyzes the experimental results across both sentiment classification tasks. Section 5 discusses the limitations of this study, and finally, Section 6 concludes the study and highlights key findings.

Corrected Proof

2 Related works

This section contextualizes our contribution within the relevant literature. We first review recent advancements in Persian sentiment analysis and transformer-based architectures for this task. Next, we survey conventional data augmentation techniques for low-resource languages, with a particular focus on Persian. Finally, we examine emerging paradigms in LLM-based synthetic data generation and delineate the specific research gap that motivates the present study.

2.1 Persian sentiment analysis

One of the earliest and most comprehensive resources for Persian sentiment analysis is the SentiPers corpus [16]. It provides sentence-level polarity labels (ranging from -2 to +2) and aspect-based annotations for Persian product reviews. Building on this resource, Sharami et al. [35] proposed DeepSentiPers, a collection of deep learning baselines including LSTM and CNN architectures, and introduced two augmented variants of SentiPers: a balanced version and a back-translated version.

Subsequent work by Soleimanzadeh and Shayegan [36] applied LSTM and CNN models (FastText-BiLSTM-CNN and Word2Vec-BiLSTM-CNN) to the Digikala and Snapfood datasets, demonstrating that hybrid deep learning models outperform individual architectures, achieving accuracies of up to 97.54% on one dataset. The introduction of transformer-based models has further advanced Persian sentiment analysis. Farahani et al. [14] introduced ParsBERT, a Persian-specific BERT model, and demonstrated its effectiveness on various downstream tasks. Davar et al. [9] proposed DeepSentiPers-BERT, a hybrid architecture combining ParsBERT with bidirectional LSTM, achieving an F1-score of 91.57% on the Digikala corpus. Similarly, Vakili et al. [39] developed a transformer-based approach for Persian Twitter sentiment analysis, integrating Naive Bayes and rule-based components with BERT, reaching 89% accuracy. An alternative strategy to address data scarcity in Persian is cross-lingual transfer learning. Ghasemi et al. [15] proposed a cross-lingual deep learning framework that leverages English training data through cross-lingual embeddings, achieving a 22% improvement in Persian sentiment analysis compared to monolingual baselines. While this approach demonstrates the feasibility of knowledge transfer from English, it requires careful alignment of embedding spaces and may suffer from domain mismatch when applied to product reviews.

2.2 Transformer-based models for sentiment analysis

The introduction of the Transformer architecture [40] has significantly advanced natural language processing. Unlike recurrent or convolutional networks, the self-attention mechanism enables transformers to capture long-range dependencies and contextual relationships more effectively. This architecture forms the foundation of state-of-the-art pretrained language models such as BERT [11] and GPT [30]. For sentiment analysis, transformer-based models have become the dominant approach due to their ability to produce rich contextualized representations. Areshey and Mathkour [4] compared multiple transformer models (BERT, RoBERTa, ALBERT, DistilBERT, XLNet) for sentiment classification, demonstrating the superiority of these architectures over traditional methods. Similarly, Cheruku et al. [8] proposed a hybrid model based on a modified RoBERTa architecture for Twitter sentiment analysis. For low-resource languages such as Persian, transformer-based models pretrained on multilingual or

Corrected Proof

language-specific corpora offer particular advantages. As mentioned in 2.1, Farahani et al. [14] introduced ParsBERT, a Persian-specific BERT model, which has been shown to outperform multilingual baselines on various Persian NLP tasks. Several other transformer variants, including RoBERTa-fa, ALBERT-fa, and DistilBERT-fa, have also been adapted for Persian. In parallel, multilingual encoders such as E5 [41] and GTE [22] support Persian representations through contrastive pretraining on large multilingual corpora.

Despite the success of these models, they remain dependent on the availability of large labeled datasets for fine-tuning. Even with powerful pretrained representations, performance on downstream tasks such as sentiment analysis is constrained by the size and diversity of the training data [25]. For low-resource languages like Persian, this dependency is particularly problematic. While transfer learning and cross-lingual approaches offer partial solutions, they do not address the fundamental need for more labeled data. To date, systematic investigations into how synthetic data generation enhances Persian-specific models remain scarce. This research gap motivates exploring generative strategies as a practical complement to strengthen model fine-tuning in Persian sentiment analysis.

2.3 Text augmentation for low-resource languages

Data augmentation is a widely adopted strategy to address data scarcity in NLP. Traditional text augmentation techniques include synonym replacement, random insertion or deletion, and back-translation [7]. Back-translation, in particular, has been widely used for low-resource languages [13]. In this procedure, a sentence is translated into a pivot language (e.g., English) and subsequently translated back to the original language, producing paraphrased variants that preserve semantic meaning. For Persian sentiment analysis, Taheri et al. [37] applied back-translation to enhance aspect-based sentiment analysis, demonstrating performance improvements over non-augmented baselines. Similarly, the DeepSentiPers corpus [35] introduced a balanced variant through controlled class rebalancing and a back-translated variant, effectively doubling the dataset size. These studies suggest that traditional augmentation techniques can improve Persian sentiment classification. A systematic evaluation of text augmentation techniques for low-resource languages was conducted by Sahin [33], who compared syntax-level, token-level, and character-level methods across multiple languages and sequence tagging tasks. The study found that character-level methods were the most consistent performers, and that augmentation was particularly effective for morphologically rich languages. Persian, with its rich morphology, falls into this category, suggesting that augmentation can be beneficial for Persian NLP tasks.

Despite their utility, traditional augmentation methods have inherent limitations that restrict their efficacy in deep learning models. Token-level manipulations, such as synonym replacement, often operate on surface features rather than deep semantic contexts. As a result, they frequently suffer from semantic drift, particularly when applied to polysemous words or morphologically rich structures in Persian [18]. Similarly, back-translation remains highly dependent on the quality and domain adaptation of the underlying translation systems, which can introduce artificial translation artifacts or systematically distort the subtle sentiment expressed in the text. Crucially, these heuristic methods merely transform or perturb existing sentences rather than producing entirely new instances. Consequently, they fail to substantially expand the diversity of the training distribution. This constraint has motivated a shift toward leveraging large language models (LLMs) to synthesize novel, label-consistent training examples from scratch.

Corrected Proof

2.4 LLM-based synthetic data augmentation

Recent advances in LLMs have broadened the scope of data augmentation in natural language processing. Unlike traditional methods that primarily perturb existing sentences through lexical substitution, paraphrasing, or translation, LLMs can generate new task-relevant samples with richer semantics and greater contextual flexibility. This is particularly valuable for sentiment classification, where preserving polarity, sentiment intensity, and label consistency is critical. Prior research on sentiment shifts in LLM-rephrased text further indicates that advanced language models can often preserve sentiment-related content under controlled rewriting, although this is not guaranteed in all cases [32].

A growing body of research has shown that LLM-based augmentation should not be viewed solely as a text generation mechanism. Rather, LLMs may serve several complementary functions in the augmentation pipeline. These roles include generating training instances, rewriting examples into label-consistent variants, enriching inputs with contextual cues, and validating candidate outputs before model training. This broader view is consistent with recent surveys and framework-oriented studies. For example, Ding et al. [12] organize LLM-based augmentation from multiple data and learning perspectives, while Chai et al. [6] provide a systematic overview of augmentation strategies, granularity levels, and evaluation practices. Similarly, Yousefi Jordehi et al. [18] highlight the importance of label-aware synthetic sample generation for classification settings, where the value of augmentation depends not only on diversity but also on preserving class-discriminative semantics. Furthermore, Bayer et al. [5] demonstrated the efficacy of LLM-based text generation in improving performance across various NLP tasks.

Despite these advantages, recent studies also emphasize that LLM-generated data may introduce new risks. Li et al. [21] show that synthetic examples can inherit and amplify biases present in the underlying model, which may subsequently affect downstream learning. Other studies report that rewriting-based augmentation can cause semantic drift, especially when domain-specific or sentiment-bearing expressions are replaced by more generic alternatives. For instance, comparative analyses by Pavlyshenko and Stasiuk [27] likewise indicate that although LLM-based augmentation is often effective, simpler methods may remain competitive when they are carefully aligned with the task and model architecture. These findings suggest that synthetic data quality should be assessed not only in terms of quantity or diversity, but also with respect to semantic fidelity, label preservation, and distributional compatibility with naturally occurring text.

Another important direction in the literature concerns low-resource and multilingual scenarios, where LLM-based augmentation has shown particular promise. Ubani et al. [38] demonstrated that instruction-based generation can produce useful synthetic data for low-resource text classification even in few-shot settings. In a similar vein, Wang et al. [42] discuss the growing role of data augmentation in low-resource and multilingual NLP settings, including the increasing use of LLM-based approaches for generating task-relevant data in under-resourced scenarios. However, the current literature remains heavily concentrated on high-resource languages, especially English, and systematic evidence for Persian is still limited. Therefore, although LLM-based augmentation has become an increasingly influential paradigm, its effectiveness for Persian sentiment classification remains insufficiently explored.

Overall, the literature indicates that the key strength of LLM-based augmentation is its ability to move beyond surface-level transformations and generate semantically richer, task-conditioned data. However, its effectiveness depends on carefully controlling generation, rewriting, and filtering to mitigate hallucination, semantic drift, bias inheritance, and distributional artifacts. Building on these obser-

Corrected Proof

vations, this study examines whether label-consistent samples generated by modern LLMs can improve the performance of transformer-based models for Persian sentiment classification.

From a methodological perspective, leveraging the internal, parametric knowledge of frozen, pre-trained LLMs offers a compelling alternative to traditional rule-based data generation and resource-intensive fine-tuning. Traditional linguistic augmentation often relies on rigid, hand-crafted rules or lexical databases, which fail to capture context-dependent semantic nuances and struggle with morphological variations in languages like Persian. In contrast, LLMs contain rich, multi-lingual linguistic knowledge that can be prompted to generate diverse, contextually appropriate, and label-consistent samples without requiring explicit rule definition or additional training parameter updates. Recent studies support this parameter-fixed paradigm. For instance, Zhang et al. [46] highlight that leveraging the internal knowledge of instruction-tuned LLMs to generate entirely new instances from scratch yields superior linguistic diversity compared to simple surface-level rewriting. Furthermore, in tasks requiring precise label preservation, Radliński et al. [31] highlighted that such models maintain high semantic fidelity, ensuring that the generated training data remains consistent with the target sentiment classes. Exploiting these models as zero-shot or few-shot knowledge sources avoids the computational overhead of fine-tuning. However, it also introduces quality-related challenges. To mitigate these risks, confidence-based filtering and aggregation methods have been proposed to secure the integrity of the generated labels [45]. By tapping into these rich parametric representations, we extract high-quality, linguistically varied training instances for Persian sentiment classification. This strategy provides targeted structural and lexical support across underrepresented categories without requiring manual rule design or heavy model retraining.

2.5 Summary and research gap

In summary, prior work has established several key foundations for Persian sentiment analysis, including annotated corpora such as SentiPers and DeepSentiPers alongside baseline architectures like LSTM and CNN. More recently, Persian-specific or adapted transformer models such as ParsBERT have further advanced the field. In addition, traditional augmentation and data-balancing techniques, particularly back-translation and class balancing, have been shown to improve performance in Persian sentiment classification. More recent studies on LLM-based augmentation have also demonstrated its potential, especially in high-resource language settings, while highlighting important methodological and evaluation-related considerations.

Nevertheless, several important gaps remain. Traditional augmentation methods typically generate variations of existing sentences. Consequently, they exhibit limited ability to produce fully novel and semantically diverse training instances. Cross-lingual transfer approaches provide another useful direction, but they require careful alignment across languages and may be affected by domain mismatch. Most importantly, to the best of our knowledge, no systematic study has examined LLM-based synthetic data generation specifically for Persian sentiment classification across multiple models and task settings.

Motivated by this gap, the present study provides a systematic evaluation of LLM-based synthetic data augmentation for Persian sentiment analysis. Specifically, we (i) generate new synthetic sentences using two recent LLMs, DeepSeek-V3-0324 and GPT-4o mini, (ii) evaluate their impact across nine transformer-based models under a unified fine-tuning framework, (iii) compare three non-LLM data configurations (Original, Balanced, and Translation) with their LLM-enhanced counterparts, and (iv)

Corrected Proof

examine both binary and five-class sentiment classification settings. The following sections describe our methodology and experimental results in detail.

3 Materials and methods

This section describes the methodology and experimental framework used to investigate and evaluate the impact of LLM-generated data on the performance of Persian sentiment classification.

3.1 Theoretical background of the transformer architecture

The Transformer [40] uses a self-attention mechanism to capture long-range dependencies in sequences. Formally, given an input sequence $X = [x_1, x_2, \ldots, x_n]$, each token is projected into query, key, and value vectors:

$$Q = XW_Q, \qquad K = XW_K, \qquad V = XW_V. \tag{1}$$

The scaled dot-product attention is then computed as:

$$\text{Attention}(Q,K,V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \tag{2}$$

where d_k is the key dimension. This operation allows each token to attend to all others in the sequence. Multi-head attention applies this in parallel across multiple heads:

$$\text{MultiHead}(Q,K,V) = \text{Concat}(h_1, \ldots, h_H)W^O. \tag{3}$$

For sequence classification, a fixed-size sentence representation must be derived. In this study, we adopt two strategies depending on the model family. For BERT-based models (ParsBERT, RoBERTa, ALBERT, DistilBERT), we use the final hidden state of the [CLS] token:

$$\hat{y} = \text{softmax}\left(W_c \cdot h_{[\text{CLS}]} + b_c\right). \tag{4}$$

For embedding-optimized models (E5, GTE), we use mean pooling over all non-padding token representations:

$$h_s = \frac{1}{N}\sum_{i=1}^{N} h_i, \qquad \hat{y} = \text{softmax}\left(W_c \cdot h_s + b_c\right). \tag{5}$$

This mathematical formulation underlies the Transformer-based encoders and the task-specific classification layers employed in our experiments. For a detailed exposition, we refer the reader to [40].

3.2 Dataset preparation and augmentation

Our methodology begins with systematic data preparation and expansion to construct multiple benchmark variants of Persian sentiment corpora. We specifically structure these configurations to exhibit controlled differences in class balance and linguistic diversity, enabling precise evaluation of the resulting downstream effects. The following subsections detail the baseline corpora, enhancement techniques, and final training configurations used for experimental validation.

Corrected Proof

3.2.1 Dataset overview

The datasets used in this research are derived from SentiPers, a comprehensive Persian sentiment corpus introduced by Hosseini et al. [16]. SentiPers was originally developed to facilitate sentiment and opinion mining tasks in Persian. It compiles user-generated reviews from Digikala, a major Iranian e-commerce platform. The corpus contains both sentence-level polarity labels and aspect-based annotations. Consequently, it represents one of the most detailed publicly available resources for Persian sentiment analysis. Its combination of informal user language, domain diversity, and polarity granularity enables reliable benchmarking of NLP models trained for sentiment classification in Persian.

All baseline datasets used in this study, described in the subsequent subsection, are systematically derived from SentiPers through balancing and back-translation processes. In addition, the augmented datasets introduced later in this work further extend these baselines through LLM-based generation, while preserving structural and labeling consistency with the original corpus.

3.2.2 Baseline datasets

The baseline datasets employed in this study are those introduced in DeepSentiPers [35], with the following properties:

1. Original dataset (*Original*): This version corresponds to the raw sentence-level corpus derived from SentiPers, comprising 5,561 annotated sentences across five sentiment classes.
2. Balanced dataset (*Balanced*): To mitigate the skewed class distribution observed in the original corpus, an extra data augmentation procedure was applied following the DeepSentiPers approach. This process increases the number of sentences in minority sentiment classes and slightly reduces those in majority classes, resulting in a more evenly distributed dataset of 7,023 labeled samples. The method can be viewed as a controlled data noising strategy [44] aimed at improving representativeness without altering polarity.
3. Translation-based dataset (*Translation*): Each sentence from the balanced version was translated into English and back-translated into Persian using the Google Translate API. This back-translation method [13] introduces paraphrastic and lexical variation while preserving the original sentiment polarity, effectively doubling the dataset size to 14,046 samples.

3.2.3 LLM-driven synthetic data generation

To tackle the persistent challenges of data scarcity and class imbalance in Persian sentiment analysis, we developed a text generation strategy using LLMs to create high-quality samples. We employed two state-of-the-art LLMs, DeepSeek-V3 [23] (specifically DeepSeek-V3-0324) and GPT-4o mini [1]. These models generated new sentences corresponding to our five sentiment classes, ranging from –2 (strongly negative) to +2 (strongly positive). Both models were accessed through their standard web interfaces without fine-tuning, operating strictly under default inference settings.

We selected approximately 200 source sentences strictly from the training split of the original dataset using an automated, label-aware procedure. The number of selected sentences from each sentiment class was determined approximately in proportion to its frequency in the full dataset, ensuring label coverage. Within each class, longer sentences were preferred to provide richer contextual prompts for the LLMs.

Corrected Proof

In our generation pipeline, each selected sentence and its sentiment label were formatted into a structured Persian prompt to maintain linguistic naturalness and avoid translation-induced artifacts. The prompts directed the models to preserve the source sentiment polarity and intensity while introducing lexical and syntactic diversity within the same domain. To exemplify this approach, below is a prompt used specifically for generating negative sentiment samples (label = –1).

Sample prompt (originally written in Persian and translated here for clarity):

"For a five-class sentiment classification task with labels ranging from –2 (strongly negative) to +2 (strongly positive), we aim to perform data augmentation by generating new sentences from existing labeled samples. As an expert in sentiment-aware Persian text generation, read the following sentence, which has a sentiment label of –1, and generate 5 new Persian sentences in the same domain that preserve both the sentiment polarity and its intensity, while maintaining coherence and natural language quality."

In the first generation stage, the ∼200 selected seed sentences were submitted to DeepSeek-V3-0324. Five candidate sentences were generated per seed, yielding 1,000 synthetic sentences. These samples were added to the *Original* dataset to form *Original*(+1000). To further improve the class distribution, we subsequently identified underrepresented sentiment classes requiring additional samples. For each such class, a random subset of seed sentences with the corresponding label was selected from the same 200-sentence pool. Each selected seed was queried to generate five additional candidates. Because some outputs exhibited high structural overlap with earlier generations, we did not retain all five candidates. Instead, two to three distinct sentences were selected per source through review and consensus among three research team members. This step provided 146 additional instances, forming *Original*(+1146). In the subsequent scaling stage, an additional 500 sentences were generated using GPT-4o mini as a complementary generator, relying on the same restricted pool of ∼200 training seed sentences. This shared, compact seed-pool design enabled us to examine the effect of scaling augmentation volume across different accessible models under a deliberately constrained low-resource setting. Adding these 500 sentences to *Original*(+1146) produced *Original*(+1646). Finally, we applied the full set of 1,646 synthetic samples to the other two baseline corpora, producing *Balanced*(+1646) and *Translation*(+1646).

The held-out test set remained fully isolated throughout the augmentation process and was not used at any stage of sample generation or selection. To further mitigate the risk of data leakage, we performed an exact-match overlap audit between the generated texts and the unseen test partition. This analysis revealed no verbatim overlaps, confirming that the augmented corpus contains no leaked evaluation instances.

To assess whether the LLM-generated sentences preserved the intended sentiment labels of their source sentences, we manually reviewed 300 synthetic sentences sampled from the 1,000 generated outputs while approximately preserving the distribution of the five sentiment classes. The selected sentences were shuffled before annotation, and annotators were not shown the source sentences or their labels. Outputs originating from the same source sentence were also separated to prevent them from appearing consecutively. Three research team members independently assigned one of the five sentiment labels to each sentence. When at least two annotators agreed, their majority label was adopted. In cases of complete disagreement, the final label was established through discussion and consensus. Notably, the final consensus labels matched the targeted seed polarities in over 90% of the sampled cases, confirming that the LLM pipeline reliably adhered to sentiment intensity instructions without semantic

Corrected Proof

drift. Annotators also assigned each sentence an overall quality label (good, medium, or poor) based on fluency, grammatical acceptability, and semantic coherence. All evaluated instances satisfied basic coherence criteria, with zero samples rated as *poor*. Furthermore, the vast majority were unanimously rated as *good*, demonstrating that the generated samples introduce valid linguistic variability without artificial noise.

It is worth noting that all datasets shared the same test set to ensure consistent evaluation across experiments. A comprehensive summary of all datasets, including the baseline versions discussed in Section 3.2.2 and the augmented variants, is presented in Table 1. This overview provides detailed class distributions for both five-class and binary configurations, serving as a reference for all subsequent analyses. Moreover, for the binary sentiment classification task, neutral samples are excluded, and the non-neutral polarities are collapsed into two broader classes, negative (labels $\{-1,-2\}$ to 0) and positive (labels $\{+1,+2\}$ to 1). Consequently, the shared binary test set comprises exactly 1,111 samples, consisting of 196 negative (17.64%) and 915 positive (82.36%) instances.

Table 1: Comprehensive summary of all datasets used in this study

Dataset Name	Dataset Type	Total Samples	-2	-1	0	+1	+2
Original	Baseline	5,561	28	513	2,409	1,623	988
Balanced	Baseline	7,023	158	937	2,000	2,000	1,928
Translation	Baseline	14,046	316	1,874	4,000	4,000	3,856
Original(+1000)	Augmented	6,561	33	618	2,844	1,958	1,108
Original(+1146)	Augmented	6,707	34	619	2,905	1,958	1,191
Original(+1646)	Augmented	7,207	36	665	3,123	2,103	1,280
Balanced(+1646)	Augmented	8,669	166	1,089	2,714	2,480	2,220
Translation(+1646)	Augmented	15,692	324	2,026	4,714	4,480	4,148
Test	Test	1,854	12	184	743	561	354

To ensure a fair, consistent, and reproducible evaluation process, all baseline and LLM-augmented datasets are ultimately fed into a unified fine-tuning framework. This framework is designed to handle different dataset variants and sentiment classification settings (binary and five-class) within a single pipeline, while maintaining identical preprocessing, tokenization, and training procedures across models. In practice, the framework first applies the appropriate label processing strategy, then performs model initialization and fine-tuning, followed by validation-driven hyperparameter selection and final testing. A schematic representation of this framework is illustrated in Figure 1, which summarizes the overall workflow. Further methodological details and experimental analysis related to this framework are discussed in the subsequent sections.

3.2.4 Intrinsic quality and diversity analysis of synthetic data

To characterize the intrinsic linguistic properties of the synthetic corpus and assess potential redundancy, we conducted an intrinsic lexical audit across the entire synthetic corpus (N = 1,646, 27,017 total tokens). Standard text preprocessing revealed 1,622 strictly unique sentences, yielding an exact duplicate rate of merely 1.46%. Pairwise word-level Jaccard similarity analysis ($\approx 1.35 \times 10^6$ pairs) demonstrated near-zero lexical redundancy. Only 4 sentence pairs (0.0003%) exhibited a Jaccard similarity exceeding 0.80, and only 9 pairs (0.0007%) exceeded 0.70. The mean pairwise Jaccard distance reached 0.9555, indicating substantial inter-sentence variance. Furthermore, the synthetic text achieved a Type-Token Ratio (Distinct-1) of 0.1556 (4,205 unique types) and a Distinct-2 score of 0.6252, con-

Corrected Proof

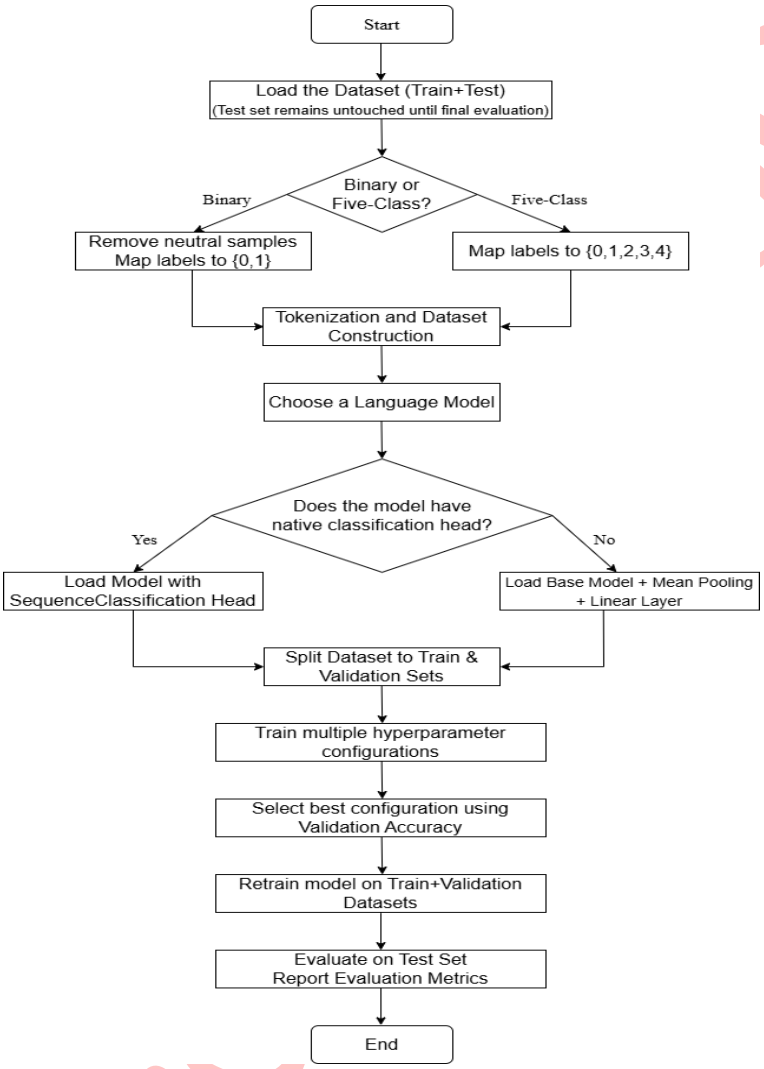


Figure 1: The schematic illustration of the unified fine-tuning framework with validation-based hyperparameter selection and full-data retraining

firming substantial syntactic and compositional richness. Complementing these automated metrics, our blinded evaluation of 300 stratified samples (representing 18.2% of the corpus, 95% confidence level, $\pm 5.1\%$ margin of error) demonstrated high sentiment fidelity and syntactic coherence.

3.3 Models

Recent progress in natural language processing has positioned transformer-based architectures at the core of modern text understanding systems. Their ability to model long-range dependencies through self-attention and to build rich contextualized representations makes them particularly effective for sentiment analysis, especially in low-resource languages such as Persian, where subtle emotional and syn-

tactic variations must be captured accurately. In this research, we employ a diverse set of transformer models that are either specifically designed for Persian or pretrained on large multilingual corpora. This model variety allows us to evaluate how LLM-based augmentation affects sentiment classification performance across different architectural paradigms by comparing baseline versus augmented conditions within each model. The subsequent subsections describe the selected models and their adaptation to the Persian sentiment domain.

3.3.1 Overview of selected transformer models

ParsBERT is a Persian-specific transformer built upon the BERT-base architecture. It was pretrained on a large-scale Persian text corpus, including Persian Wikipedia, online news, and social media data. This extensive exposure enables ParsBERT to capture morphological and syntactic structures unique to Persian, making it an appropriate benchmark for Persian NLP tasks [14]. ParsBERT v2.0 follows the same BERT-base architecture as its predecessor but was further pretrained on an expanded and more diverse Persian corpus comprising over 3.9 million documents, 73 million sentences, and 1.3 billion tokens. In this updated release, the vocabulary was reconstructed to better capture modern Persian morphology and domain variety. This extended pretraining allows a systematic examination of how corpus scale and lexical coverage during pretraining influence downstream sentiment classification performance [14].

ALBERT-fa is a parameter-efficient adaptation of the ALBERT architecture, which introduces two core optimizations: factorized embedding parameterization and cross-layer parameter sharing. These design modifications substantially reduce the model size without causing notable performance degradation [20], thereby enabling faster fine-tuning while preserving rich contextual understanding in Persian text.

DistilBERT-fa is a Persian adaptation of the DistilBERT architecture introduced by Sanh et al [34]. It employs knowledge distillation to transfer the linguistic capabilities of a larger BERT model into a smaller and more efficient network. By retaining approximately two-thirds of BERT's layers, DistilBERT-fa preserves most of the representational capacity while significantly reducing computational cost, making it particularly suitable for fine-tuning on low-resource Persian sentiment analysis tasks.

RoBERTa-fa builds upon the RoBERTa framework introduced by Liu et al. [24], which refined the original BERT training methodology by eliminating the next-sentence prediction objective, employing dynamic masking, and extending the pretraining duration. The Persian adaptation leverages these optimizations to achieve more stable and context-aware representations, particularly beneficial for modeling long and syntactically rich review sentences often found in Persian sentiment data.

The multilingual version of E5-small aligned-sentiment model is derived from the E5 family of text encoders [41], originally designed for embedding and retrieval applications through contrastive pretraining. This variant offers a compact architecture with multilingual capabilities and fine-tuning alignment toward sentiment understanding, making it well-suited for low-resource sentiment analysis tasks in Persian.

E5-General-v2 is a large-scale encoder model derived from the E5 family [41], originally trained through weakly supervised contrastive pretraining for general-purpose text embeddings. This variant has been further fine-tuned on sentiment-oriented data across diverse domains, enhancing its ability to capture affective semantics and providing a robust comparative baseline for evaluating the influence of

Corrected Proof

LLM-generated synthetic data.

GTE-multilingual-base is a large-scale encoder designed for universal text embedding generation [22]. It builds upon the contrastive learning paradigm introduced in E5, extending it through multilingual and multi-domain pretraining to enable robust cross-lingual semantic alignment. The model provides contextually rich representations suitable for sentiment and semantic inference tasks. GTE-multilingual-reranker-base is a ranking-oriented extension of the GTE family, trained using pairwise contrastive objectives to enhance semantic relevance modeling [22]. Although originally optimized for reranking applications, its multilingual pretraining makes it transferable to sentiment and other text understanding tasks.

This diverse set of nine transformer models, encompassing both Persian-specific and multilingual architectures, provides a balanced experimental foundation. Our intentional selection covers a wide spectrum of design philosophies, from compact efficient encoders to large-scale multilingual transformers. This enables a systematic investigation of how synthetic data influences performance across different representation paradigms and yields a fair and generalizable assessment of augmentation effects for Persian sentiment analysis. A summary of the key specifications for each model is provided in Table 2.

Table 2: Architectural specifications and input-length settings of the Transformer models evaluated in this study

Model	Parameters (M)	Tokenizer Type	Maximum Supported Length	Length Used in Experiments
ParsBERT	∼163	WordPiece	512	128
ParsBERT v2.0	∼163	WordPiece	512	128
RoBERTa-fa	∼118	BPE	512	128
ALBERT-fa	∼12	SentencePiece	512	128
DistilBERT-fa	∼66	WordPiece	512	128
E5-Small	∼146	XLM-R tokenizer	512	128
E5-General	∼560	XLM-R tokenizer	512	128
GTE	∼305	XLM-R tokenizer	8192	128
GTE-Reranker	∼305	XLM-R tokenizer	8192	128

3.3.2 Model adaptation for sentiment classification

To ensure a consistent and comparable evaluation across all dataset configurations, we implemented a unified fine-tuning framework that adapts transformer-based encoders for supervised sentiment classification. For models originally designed as embedding generators, such as E5 and GTE, a lightweight linear classification layer was appended and optimized using cross-entropy loss. The cross-entropy objective is a standard choice for multi-class classification, defined as:

$$\mathscr{L}_{CE} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} y_{i,c}\log(\hat{y}_{i,c}), \tag{6}$$

where N denotes the number of samples, C is the number of sentiment classes, $y_{i,c}$ is the true one-hot label, and $\hat{y}_{i,c}$ represents the predicted probability for class c. This formulation penalizes incorrect predictions proportionally to their confidence, encouraging stable convergence across imbalanced datasets.

This extension transforms general-purpose sentence encoders into task-specific classifiers while preserving their pretrained semantic representations. At the sentence level, token embeddings from the

Corrected Proof

final encoder layer were aggregated through mean pooling, a strategy that provides robust representations for variable-length Persian sentences. The entire pipeline emphasizes reproducibility and fairness rather than architectural innovation. All models followed an identical training protocol, including the same optimizer, the same validation-based hyperparameter selection procedure, and the same number of training epochs. The final hyperparameter values (learning rate and batch size) were selected independently for each model from a candidate grid based on validation accuracy. This controlled setup ensures that observed performance differences stem from dataset variations and augmentation effects rather than model-specific tuning, forming a reliable experimental benchmark for analyzing the influence of LLM-generated data on Persian sentiment modeling.

In summary, all transformer models follow a common pipeline consisting of encoding input text into contextualized representations via self-attention, deriving fixed-size sentence vectors ([CLS] or mean pooling), and applying a linear softmax classifier for binary or five-class tasks. This architecture leverages pretrained contextual information to effectively capture fine-grained sentiment distinctions.

4 Results and discussion

This section empirically evaluates LLM-based data augmentation across nine transformer architectures for Persian sentiment analysis. The findings demonstrate that while this strategy effectively enhances performance, the magnitude of the gains depends heavily on task granularity, model architecture, and sample volume. The remainder of this section is organized as follows: Section 4.1 and Section 4.2 detail the experimental framework and evaluation metrics. Section 4.3 and Section 4.4 present paired baseline comparisons and peak performance ceiling evaluations. Section 4.5 investigates task granularity and augmentation sensitivity, Section 4.6 analyzes the non-monotonic impact of augmentation size. Section 4.7 provides an architectural and mechanistic discussion of the observed behaviors, and Section 4.8 formally quantifies the statistical significance and effect size of our results.

4.1 Implementation details and hyperparameters settings

The mathematical formulation of the Transformer architecture and the classification layers, described in section 3.1, directly supports the steps of Algorithm 1. Specifically, the self-attention and multi-head attention mechanisms (Eq. 1–3) correspond to the encoding of input text in the model forward pass, while the softmax classification layers (Eq. 4 and Eq. 5) correspond to the prediction step. The cross-entropy loss function (Eq. 6) is used as the training objective in the fine-tuning step.

All experiments were conducted under a unified and controlled training framework to ensure fair comparison across datasets and transformer architectures. As shown in Figure 1 and formally defined in Algorithm 1, all models follow the same fine-tuning and evaluation pipeline. The implementation was carried out using the Hugging Face Transformers library with PyTorch backend, executed on an NVIDIA Tesla T4 GPU with mixed-precision (fp16) training to improve computational efficiency. All experiments were repeated three times with different random seeds (42, 123, 2026) to ensure robustness.

Corrected Proof

Algorithm 1: The Unified fine-tuning with validation-based hyperparameter selection and full-data retraining

Require: Train/Test files, pretrained model M, candidate hyperparameter set $\mathscr{H}$, max length L
Ensure: Final Test Accuracy, Weighted F1-score, Macro F1-score, Confusion Matrix.

Load training and test datasets.
Determine task type:
if binary classification **then**
    Remove neutral samples and map labels to $\{0,1\}$.
else
    Map labels $\{-2,-1,0,1,2\}$ to $\{0,1,2,3,4\}$.
end if
Split training data into training and validation subsets (e.g., 85% / 15%).
Tokenize all texts using the pretrained tokenizer with maximum sequence length L.
Initialize *best_score* $\leftarrow 0$, *best_hparams* $\leftarrow$ *null*.
for each hyperparameter configuration $h \in \mathscr{H}$ **do**
    Initialize model (with native head or custom classifier depending on M).
    Fine-tune on the training split using h.
    Evaluate on the validation split and compute accuracy.
    if validation accuracy $>$ *best_score* **then**
        best_score $\leftarrow$ accuracy
        best_hparams $\leftarrow h$
    end if
end for
Reinitialize the model.
Retrain using *best_hparams* on the combined Training + Validation data.
Evaluate the final model on the held-out Test set.
return Final Test Accuracy, Weighted F1-score, Macro F1-score, and Confusion Matrix.

For each dataset and model, training followed a validation-driven selection strategy. After dataset loading, text labels were mapped according to the target task, namely binary sentiment classification or five-class polarity prediction. The training file was then divided into training and validation subsets using an 85%/15% ratio and stratified sampling based on sentiment labels. The test set remained in a separate held-out file and was completely isolated and untouched until the final evaluation. All text inputs were tokenized using model-specific tokenizers with truncation and padding to a fixed maximum length $L = 128$.

To reduce tuning bias and improve reproducibility, model selection was restricted to a predefined set of candidate hyperparameter configurations, $\mathscr{H}$, rather than unrestricted optimization. Each configuration was fine-tuned on the training split and evaluated on the validation split, with the best-performing configuration selected based on validation accuracy. This search was performed once on the first run (seed 42), and the selected hyperparameters were kept fixed across subsequent runs to isolate the effect of random initialization. The model was then reinitialized and retrained with the selected hyperparameters on the combined training and validation data to fully utilize the available supervision. Final performance was reported on the held-out test set.

Corrected Proof

Two fine-tuning strategies were employed, depending on model architecture. For transformer models with native classification heads, training was conducted using the Hugging Face Trainer API. For embedding-oriented architectures, custom PyTorch training loops were implemented, incorporating mean-pooling over token embeddings followed by a linear classification layer. All models were optimized using the AdamW optimizer. A discrete hyperparameter search was performed over three learning-rate candidates $2\times10^{-5}, 3\times10^{-5}, 5\times10^{-5}$. For the six smaller models, three batch sizes $\{8, 16, 32\}$ were included in the search space. For the three larger models (E5-General, GTE, and GTE-Reranker), the batch size 32 was excluded due to GPU memory constraints on Google Colab, resulting in a search space of $\{8, 16\}$. The number of training epochs was fixed to 3 across all experiments. These values were selected based on widely adopted practices for fine-tuning transformer-based models, while also respecting GPU memory constraints.

This standardized experimental design ensures that observed performance differences primarily reflect dataset composition and the effects of LLM-based augmentation, rather than uncontrolled hyperparameter tuning, while also supporting stability, fairness, and reproducibility across all evaluated configurations. To ensure full experimental transparency, the complete implementation details, including precise model identifiers from the Hugging Face, generation prompts and hyperparameters, random seeds, and library requirements, are curated and publicly accessible at the project repository.

4.2 Evaluation metrics

To quantitatively evaluate model performance on both five-class and binary sentiment classification tasks, three complementary metrics were adopted: Accuracy, Weighted F1-score, and Macro F1-score. These metrics jointly provide a comprehensive view of model behavior by capturing global correctness, equal sensitivity across all sentiment classes, and support-weighted robustness under class imbalance.

Accuracy measures the overall proportion of correctly classified instances and is defined as:

$$\text{Accuracy} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{1}(\hat{y}_i = y_i), \tag{7}$$

where N denotes the total number of test samples, y_i is the true label, $\hat{y}_i$ represents the predicted label, and $\mathbf{1}(\hat{y}_i = y_i)$ is an indicator function equal to 1 when the prediction is correct and 0 otherwise. This metric provides a straightforward measure of the overall correctness of the model's predictions.

Since the sentiment datasets exhibit class imbalance, accuracy alone may not reflect true model performance. Therefore, the weighted F1-score was also employed. The F1-score for class c is defined as the harmonic mean of precision and recall for that class:

$$\text{F1}_c = 2\times\frac{\text{Precision}_c\times\text{Recall}_c}{\text{Precision}_c+\text{Recall}_c} \tag{8}$$

where for each class c, precision and recall are calculated as:

$$\text{Precision}_c = \frac{TP_c}{TP_c+FP_c}, \tag{9}$$

https://github.com/Persian-NLP-Research/Persian-Sentiment-LLM-Augmentation.

Corrected Proof

$$\text{Recall}_c = \frac{TP_c}{TP_c + FN_c}, \tag{10}$$

with TP_c, FP_c, and FN_c denoting true positives, false positives, and false negatives for class c, respectively. The weighted F1-score then aggregates these class-specific F1-scores, weighting each by its class support:

$$\text{F1}_{\text{weighted}} = \sum_{c=1}^{C} \frac{n_c}{N} \cdot \text{F1}_c, \tag{11}$$

where n_c denotes the number of samples in class c, and N is the total number of samples.

To assess model performance across all classes equally, the Macro F1-score is computed as the unweighted arithmetic mean of the per-class F1-scores:

$$\text{F1}_{\text{macro}} = \frac{1}{C} \sum_{c=1}^{C} \text{F1}_c, \tag{12}$$

where C denotes the total number of classes ($C = 5$ for the fine-grained task and $C = 2$ for binary classification).

To complement aggregate metrics with granular, class-level error diagnostics, row-normalized confusion matrices were constructed. For any true class i and predicted class j, the normalized matrix entry $\bar{M}_{i,j}$ represents the conditional probability of predicting class j given true class i:

$$\bar{M}_{i,j} = \frac{M_{i,j}}{\sum_{k=1}^{C} M_{i,k}}, \tag{13}$$

where $M_{i,j}$ denotes the raw count of instances with true label i predicted as j. Row-normalization decouples the visual inspection from the underlying class imbalance, directly mapping diagonal entries to per-class recall ($\bar{M}_{i,i} = \text{Recall}_i$) and off-diagonal entries to directional misclassification dynamics.

All metrics were computed using scikit-learn [28], with definitions consistent with standard evaluation literature [29].

4.3 Paired performance analysis

To isolate the genuine contribution of LLM-generated samples from the inherent distributional traits of the base datasets, this section conducts a strictly controlled paired difference analysis. Rather than contrasting disparate configurations across dataset families, each baseline variant, namely *Original*, *Balanced*, and *Translation*, is directly compared against its corresponding augmented counterpart. Tables 9 and 10 present the performance deltas ($\Delta = Augmented - Baseline$) for the five-class and binary sentiment classification tasks across nine transformer-based architectures. Detailed performance metrics across both baseline and augmented configurations are reported in Tables 3–8 for five-class and binary tasks.

4.3.1 Five-class sentiment classification

As summarized in Table 9, introducing LLM-synthesized data yields substantial gains across the majority of evaluated configurations, with the most pronounced improvements observed in Macro F1-score

Corrected Proof

(ΔM-F1). Enhancing the unadjusted raw data with LLM samples yields uniformly positive accuracy gains (ΔAcc ranging from +0.04 to +1.62 percentage points) and weighted F1-score. Crucially, substantial surges in Macro F1-score emerge, reaching +9.04 percentage points for GTE, +6.57 for GTE-Reranker, and +5.30 for RoBERTa-fa. Because the five-class raw corpus is constrained by label skewness and subtle affective boundaries, the synthetic additions appear to populate sparse regions of the feature space, markedly bolstering decision margins for minority and neutral categories.

Imposing prior class balancing already stabilizes baseline representations. Nonetheless, further augmenting this set with 1,646 LLM samples produces complementary gains in the majority of embedding and lightweight architectures. Notably, E5-Small and GTE demonstrate steady improvements across all metrics (Δ *Acc* of +1.95 and +1.39, respectively). Minor contractions in accuracy for ParsBERT and AlBERT-fa suggest that when label parity is enforced heuristically, excessive synthetic density can occasionally perturb the decision boundaries of dense representations. Augmenting the cross-lingual pseudo-labeled corpus improves accuracy and weighted F1 across all nine architectures, peaking at +1.97 and +1.92 percentage points for AlBERT-fa. While Macro F1 exhibits marginal trade-offs in three native Persian encoders (ParsBERT, RoBERTa-fa, and DistilBERT-fa), embedding-focused backbones such as GTE (ΔM-F1 = +3.24) and AlBERT-fa (ΔM-F1 = +3.53) exhibit substantial synergistic improvements.

4.3.2 Binary sentiment classification

The binary classification results in Table 10 reaffirm the viability of LLM augmentation, though the narrower baseline error rates naturally moderate the magnitude of performance shifts. In the *Original* vs. *Augmented* comparison, eight of the nine models achieve positive deltas across all three metrics. DistilBERT-fa benefits the most (ΔAcc = +1.26, ΔM-F1 = +2.52), closely followed by GTE and GTE-Reranker. This reinforces the hypothesis that compact models gain robust semantic regularization from high-fidelity LLM prompts. AlBERT-fa remains the sole exception, displaying a minor contraction across metrics.

Under the *Balanced* condition, LLM augmentation yields notable recoveries and peaks, highlighted by AlBERT-fa (ΔAcc = +1.92, ΔM-F1 = +2.98) and GTE-Reranker (ΔAcc = +1.23, ΔM-F1 = +1.94). For the *Translation* family, positive deltas are sustained in five architectures, with DistilBERT-fa and E5-Small showing marked gains (ΔAcc of +0.75 and +0.57, respectively). Slight degradations observed in GTE and RoBERTa-fa point to potential stylistic dissonance between machine-translated artifacts and generative LLM syntax when operating near the empirical ceiling.

4.3.3 Contextualizing against prior benchmarks

Positioning these controlled paired gains within the broader Persian NLP landscape highlights the cumulative progress achieved. On the SentiPers benchmark, the established baseline reported by DeepSentiPers achieved maximum F1-scores of 91.98% (binary) and 69.33% (five-class) utilizing a Bi-LSTM network coupled with translation-based augmentation. By coupling contemporary transformer backbones with targeted LLM generation, our peak configurations achieve mean weighted F1-scores of 96.94% ($\pm$0.20) on *Original*($+1146$) and 79.16% ($\pm$0.42) on *Original*($+1000$) for the binary and five-class tasks, respectively (both established by E5-General). These absolute performance levels, cou-

Corrected Proof

Table 3: Paired comparison between the *Original* dataset and its LLM-augmented variants, in five-class sentiment classification (Mean Macro F1 ± standard deviation over three runs)

Models	*Original*	*Original*(+1000)	*Original*(+1146)	*Original*(+1646)
ParsBERT	57.83 ± 1.25	60.53 ± 3.42	**60.73 ± 3.73**	58.39 ± 0.32
ParsBERT v2.0	56.77 ± 0.74	57.87 ± 0.29	58.42 ± 1.45	**58.93 ± 2.17**
RoBERTa-fa	54.26 ± 0.55	55.37 ± 0.56	**59.56 ± 1.76**	56.71 ± 1.80
DistilBERT-fa	54.23 ± 0.70	**57.36 ± 2.27**	56.05 ± 1.42	56.00 ± 1.61
AlBERT-fa	51.08 ± 1.88	**53.80 ± 1.31**	53.34 ± 0.52	52.35 ± 0.53
E5-Small	56.57 ± 0.66	57.53 ± 0.40	58.07 ± 0.38	**58.24 ± 0.34**
E5-General	72.26 ± 5.03	**75.37 ± 0.86**	71.61 ± 2.61	73.96 ± 2.41
GTE	59.45 ± 0.58	61.31 ± 1.56	60.34 ± 1.86	**68.49 ± 2.57**
GTE-Reranker	59.93 ± 1.91	60.87 ± 3.74	**66.50 ± 2.59**	58.63 ± 1.10

Table 4: Paired comparison between the *Original* dataset and its LLM-augmented variants, in Binary sentiment classification (Mean Macro F1 ± standard deviation over three runs)

Models	*Original*	*Original*(+1000)	*Original*(+1146)	*Original*(+1646)
ParsBERT	87.20 ± 0.84	**88.33 ± 0.77**	88.13 ± 0.62	87.89 ± 0.95
ParsBERT v2.0	**88.03 ± 0.79**	87.33 ± 1.00	87.02 ± 0.37	**87.96 ± 0.93**
RoBERTa-fa	81.96 ± 1.49	82.95 ± 0.81	**83.51 ± 0.71**	82.25 ± 0.71
DistilBERT-fa	80.83 ± 1.56	81.93 ± 1.05	**83.35 ± 0.78**	82.70 ± 0.38
AlBERT-fa	**79.62 ± 1.28**	77.65 ± 0.80	78.56 ± 0.87	77.84 ± 1.04
E5-Small	87.50 ± 0.09	87.16 ± 1.04	87.72 ± 0.58	**88.49 ± 0.60**
E5-General	93.90 ± 0.85	93.90 ± 0.27	**94.67 ± 0.34**	93.27 ± 0.09
GTE	88.99 ± 0.95	89.58 ± 0.58	89.41 ± 0.39	**90.35 ± 1.10**
GTE-Reranker	87.73 ± 0.87	88.52 ± 1.01	**89.21 ± 1.37**	88.08 ± 0.72

Table 5: Paired comparison between the *Balanced* dataset and its LLM-augmented variant, in five-class sentiment classification (Mean ± standard deviation over three runs)

	Balanced		*Balanced*(+1646)	
Models	Weighted F1	Macro F1	Weighted F1	Macro F1
ParsBERT	**73.14 ± 0.53**	63.95 ± 1.18	72.87 ± 0.70	**65.90 ± 1.10**
ParsBERT v2.0	**73.27 ± 0.37**	**62.45 ± 1.07**	72.61 ± 0.54	60.49 ± 4.05
RoBERTa-fa	70.32 ± 0.28	58.25 ± 2.18	**70.70 ± 0.35**	**60.81 ± 1.45**
DistilBERT-fa	71.33 ± 0.66	62.12 ± 1.37	**72.01 ± 0.13**	**62.79 ± 1.44**
AlBERT-fa	**69.22 ± 0.35**	55.71 ± 0.88	69.14 ± 0.37	**58.41 ± 1.49**
E5-Small	70.28 ± 0.98	55.95 ± 0.71	**72.32 ± 0.44**	**58.51 ± 1.41**
E5-General	75.76 ± 0.73	71.03 ± 0.95	**76.93 ± 0.79**	**73.31 ± 1.55**
GTE	72.55 ± 0.08	66.23 ± 0.28	**73.98 ± 0.36**	**67.44 ± 0.70**
GTE-Reranker	72.85 ± 0.68	66.17 ± 0.48	72.88 ± 0.67	66.15 ± 0.92

pled with the rigorous paired improvements documented above, confirm that LLM-driven augmentation serves as an effective mechanism for mitigating lexical scarcity in Persian sentiment processing. It is worth noting that complete metric breakdowns are provided in the project repository.

4.4 Peak performance analysis and practical ceiling evaluation

While Section 4.3 presented a strictly controlled paired ablation to isolate the marginal contribution of LLM-generated samples across corresponding dataset variants, real-world deployment typically prioritizes identifying the highest attainable performance ceiling for each architecture. In practice, a practitioner seeks to determine whether integrating synthetic data, regardless of the specific underlying

Corrected Proof

Table 6: Paired comparison between the *Balanced* dataset and its LLM-augmented variant, in binary sentiment classification

	Balanced		*Balanced*(+1646)	
Models	Weighted F1	Macro F1	Weighted F1	Macro F1
ParsBERT	93.72 ± 0.20	89.44 ± 0.30	93.81 ± 0.50	89.49 ± 0.89
ParsBERT v2.0	**94.43 ± 0.15**	**90.53 ± 0.30**	94.26 ± 0.09	90.22 ± 0.17
RoBERTa-fa	91.50 ± 0.12	85.89 ± 0.18	**92.04 ± 0.44**	**86.61 ± 0.66**
DistilBERT-fa	91.95 ± 0.34	86.50 ± 0.50	**92.29 ± 0.03**	**87.04 ± 0.05**
AlBERT-fa	89.50 ± 0.96	82.49 ± 1.36	**91.33 ± 0.33**	**85.47 ± 0.49**
E5-Small	94.58 ± 0.36	90.75 ± 0.64	94.64 ± 0.39	90.85 ± 0.68
E5-General	**96.77 ± 0.32**	**94.45 ± 0.55**	96.34 ± 0.18	93.76 ± 0.31
GTE	94.33 ± 0.63	90.36 ± 1.09	**94.76 ± 0.31**	**91.07 ± 0.51**
GTE-Reranker	93.81 ± 0.33	89.55 ± 0.57	**94.99 ± 0.32**	**91.49 ± 0.57**

Table 7: Paired comparison between the *Translation* dataset and its LLM-augmented variant, in five-class sentiment classification

	Translation		*Translation*(+1646)	
Models	Weighted F1	Macro F1	Weighted F1	Macro F1
ParsBERT	72.92 ± 0.30	**66.04 ± 1.94**	**73.33 ± 0.96**	65.09 ± 0.77
ParsBERT v2.0	73.03 ± 0.68	64.85 ± 1.04	**73.52 ± 0.67**	64.86 ± 1.15
RoBERTa-fa	70.57 ± 0.40	**62.56 ± 0.66**	**71.09 ± 0.63**	61.86 ± 0.85
DistilBERT-fa	71.50 ± 0.17	**62.15 ± 0.52**	**72.06 ± 0.38**	61.56 ± 0.22
AlBERT-fa	69.32 ± 0.97	59.70 ± 0.15	**71.24 ± 0.92**	**63.23 ± 2.26**
E5-Small	70.61 ± 0.24	59.47 ± 1.12	**71.47 ± 0.79**	**60.76 ± 2.16**
E5-General	75.53 ± 0.75	70.08 ± 0.61	**76.09 ± 0.21**	70.08 ± 0.40
GTE	72.93 ± 1.29	64.22 ± 1.37	**74.49 ± 0.27**	**67.46 ± 0.25**
GTE-Reranker	73.19 ± 0.88	65.30 ± 1.47	**73.72 ± 1.59**	**66.11 ± 2.20**

Table 8: Paired comparison between the *Translation* dataset and its LLM-augmented variant, in binary sentiment classification

	Translation		*Translation*(+1646)	
Models	Weighted F1	Macro F1	Weighted F1	Macro F1
ParsBERT	93.83 ± 0.37	89.50 ± 0.66	93.93 ± 0.25	**89.69 ± 0.48**
ParsBERT v2.0	93.88 ± 0.13	89.48 ± 0.12	**94.04 ± 0.16**	**89.86 ± 0.24**
RoBERTa-fa	**92.57 ± 0.30**	**87.31 ± 0.51**	92.45 ± 0.51	87.10 ± 0.87
DistilBERT-fa	92.56 ± 0.68	87.36 ± 1.06	**93.30 ± 0.28**	**88.62 ± 0.46**
AlBERT-fa	91.00 ± 0.46	84.76 ± 0.76	**91.28 ± 0.45**	**85.34 ± 0.72**
E5-Small	94.55 ± 0.42	90.76 ± 0.65	**95.08 ± 0.15**	**91.59 ± 0.27**
E5-General	96.55 ± 0.36	94.07 ± 0.59	96.60 ± 0.01	94.17 ± 0.01
GTE	**94.66 ± 0.51**	**90.91 ± 0.83**	94.20 ± 0.37	90.18 ± 0.64
GTE-Reranker	94.40 ± 0.40	90.42 ± 0.65	**94.72 ± 0.16**	**90.01 ± 0.24**

augmentation approach, enables a model to surpass its best possible performance achieved solely on unaugmented data.

To establish this empirical performance envelope, this section benchmarks the absolute peak performance achieved by each model under any baseline configuration against its best result obtained under LLM-augmented conditions (derived from the optimal values across Tables 3–8). Following the experimental protocol detailed in Section 3, all reported metrics represent the empirical mean and standard deviation across three independent runs with distinct random seeds, evaluating both overall Accuracy and Macro-averaged F1 score to capture minority-class stability. The following subsections detail this

Corrected Proof

Table 9: Paired performance differences (Δ = Augmented − Baseline) across models in the five-class sentiment classification (percentage points). ΔAcc, ΔW-F1, and ΔM-F1 denote changes in Accuracy, Weighted and Macro F1-scores, respectively. Positive values indicate improvements yielded by LLM-based augmentation, while negative ones reflect performance degradation.

	Original vs +*Aug*			*Balanced* vs +*Aug*			*Translation* vs +*Aug*		
Models	ΔAcc	ΔW-F1	ΔM-F1	ΔAcc	ΔW-F1	ΔM-F1	ΔAcc	ΔW-F1	ΔM-F1
ParsBERT	+0.92	+0.94	+2.90	−0.33	−0.27	+1.95	+0.29	+0.41	−0.95
ParsBERT v2.0	+1.07	+1.09	+2.16	−0.64	−0.66	−1.96	+0.43	+0.50	+0.01
RoBERTa-fa	+1.06	+1.08	+5.30	+0.38	+0.38	+2.53	+0.52	+0.53	−0.88
DistilBERT-fa	+0.80	+1.04	+3.13	+0.55	+0.68	+0.67	+0.49	+0.56	−0.59
AlBERT-fa	+1.56	+1.62	+2.72	−0.20	−0.08	+2.70	+1.97	+1.92	+3.53
E5-Small	+1.62	+1.65	+1.67	+1.95	+2.04	+2.56	+0.77	+0.86	+1.29
E5-General	+0.04	+0.06	+3.11	+1.18	+1.17	+2.28	+0.59	+0.56	0.00
GTE	+0.78	+1.01	+9.04	+1.39	+1.43	+1.21	+1.70	+1.56	+3.24
GTE-Reranker	+0.27	+0.45	+6.57	+0.04	+0.03	−0.02	+0.58	+0.53	+0.81

Table 10: Paired performance differences (Δ = Augmented − Baseline) across models in the binary sentiment classification (percentage points). ΔAcc, ΔW-F1, and ΔM-F1 denote changes in Accuracy, Weighted and Macro F1-scores, respectively. Positive values indicate improvements yielded by LLM-based augmentation, while negative ones reflect performance degradation.

	Original vs +*Aug*			*Balanced* vs +*Aug*			*Translation* vs +*Aug*		
Models	ΔAcc	ΔW-F1	ΔM-F1	ΔAcc	ΔW-F1	ΔM-F1	ΔAcc	ΔW-F1	ΔM-F1
ParsBERT	+0.78	+0.72	+1.13	+0.15	+0.09	+0.05	+0.09	+0.10	+0.19
ParsBERT v2.0	+0.03	−0.01	−0.07	−0.15	−0.17	−0.31	+0.09	+0.16	+0.38
RoBERTa-fa	+0.72	+0.81	+1.55	+0.66	+0.54	+0.72	−0.12	−0.12	−0.21
DistilBERT-fa	+1.26	+1.37	+2.52	+0.36	+0.34	+0.54	+0.75	+0.74	+1.26
AlBERT-fa	−0.39	−0.50	−1.06	+1.92	+1.83	+2.98	+0.24	+0.28	+0.58
E5-Small	+0.48	+0.54	+0.99	+0.06	+0.06	+0.10	+0.57	+0.53	+0.83
E5-General	+0.45	+0.45	+0.77	−0.45	−0.43	−0.69	+0.03	+0.05	+0.10
GTE	+0.90	+0.84	+1.36	+0.45	+0.43	+0.71	−0.48	−0.46	−0.73
GTE-Reranker	+0.81	+0.84	+1.48	+1.23	+1.18	+1.94	+0.30	+0.32	+0.59

upper-bound comparative analysis across the five-class and binary classification settings.

4.4.1 Five-class classification task

Figure 2 illustrates the ceiling performance of the nine evaluated architectures under baseline versus LLM-augmented regimes in the five-class setup. In this analysis, peak Accuracy and peak Macro-F1 are selected independently for each architecture, meaning the best configuration under each metric may differ.

Across all architectures, peak augmented configurations matched or surpassed baseline ceilings in Accuracy. The highest overall performance was achieved by E5-General, which established the strongest empirical benchmark, reaching 79.24% (±0.40) Accuracy and a notable 75.37% (±0.86) Macro-F1, compared to its baseline ceilings of 79.20% (±1.20) and 72.26% (±5.03), respectively. Notably, while the gain in Accuracy reflects an empirical ceiling effect for this high-capacity model, the substantial increase in Macro-F1 coupled with a marked reduction in its standard deviation (from ±5.03 to ±0.86) highlights significantly enhanced cross-class separability and training stability under synthetic augmentation.

Corrected Proof

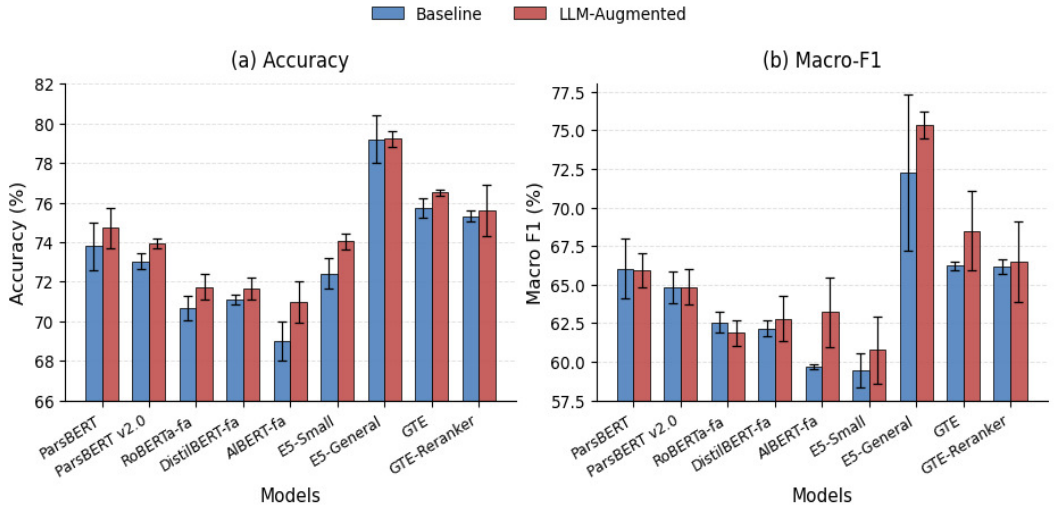


Figure 2: Performance ceilings for the five-class sentiment classification. The panels display the peak Accuracy (left) and peak Macro-F1 (right) achieved by each architecture under baseline and LLM-augmented configurations. Peak values are identified independently for each metric to demonstrate the maximum empirical capacity of each architecture.

A notable performance gain is observed in compact and parameter-efficient architectures. Specifically, AlBERT-fa exhibited the largest relative jump under generative expansion. Peak Accuracy increased from 69.00% (±0.98) to 70.97% (±1.03), while Macro-F1 improved by +3.53 points, from 59.70% (±0.15) to 63.23% (±2.26). A comparable trend is visible in E5-Small, which progressed from 72.42% (±0.79) to 74.04% (±0.39) in Accuracy and from 59.47% (±1.12) to 60.76% (±2.16) in Macro-F1. This pattern suggests that smaller-capacity models may benefit disproportionately from the additional linguistic variability introduced by LLM-generated samples.

Among standard-sized BERT variants, ParsBERT and ParsBERT v2.0 reached peak Accuracy values of 74.73% (±1.02) and 73.93% (±0.24), respectively, yielding modest improvements over their baseline ceilings (73.81% and 73.03%). For these models, Macro-F1 remained effectively unchanged (65.90% vs. 66.04% for ParsBERT, 64.86% vs. 64.85% for ParsBERT v2.0), indicating that while LLM augmentation can improve overall separability, fine-grained five-class sentiment distinctions may impose a shared structural bottleneck on classic encoder models. Conversely, retrieval-oriented encoders such as GTE achieved a notable increase in peak Macro-F1 from 66.23% (±0.28) to 68.49% (±2.57), highlighting potential synergy between contrastive-style representations and enriched contextual diversity.

4.4.2 Binary classification task

Figure 3 illustrates the performance ceilings across the nine evaluated architectures under the baseline and LLM-augmented regimes in the binary sentiment classification setup. Similar to the five-class evaluation, peak Accuracy and peak Macro-F1 are identified independently for each model architecture to reflect its maximum empirical potential under each metric.

Corrected

Consistent with the patterns observed in the multi-class setup, synthetic augmentation either preserved or elevated the performance ceiling across the majority of architectures. E5-General again established the empirical upper bound of the benchmark, reaching a peak Accuracy of 96.97% (±0.21) and a Macro-F1 of 94.67% (±0.34), improving upon its already strong baseline ceilings of 96.76% (±0.32) and 94.45% (±0.55), respectively.

A pronounced synthetic enhancement is again evident in compact and sentence-embedding models. Most notably, E5-Small achieved substantial gains under augmentation, advancing its peak Accuracy from 94.54% (±0.34) to 95.05% (±0.16) and its Macro-F1 from 90.76% (±0.65) to 91.59% (±0.27). Importantly, both metrics exhibited lower standard deviations, underscoring that this expansion contributed to greater run-to-run stability. Similarly, DistilBERT-fa demonstrated solid ceiling improvements, with peak Accuracy increasing from 92.47% (±0.75) to 93.22% (±0.29) and Macro-F1 rising from 87.36% (±1.06) to 88.62% (±0.46).

For the classic BERT variants, ParsBERT recorded slight ceiling increments in both Accuracy (93.85% vs. 93.76%) and Macro-F1 (89.69% vs. 89.50%), while ParsBERT v2.0 maintained a high, virtually saturated ceiling across both regimes (94.21% vs. 94.36% in Accuracy, 90.22% vs. 90.53% in Macro-F1). This plateau effect in binary ParsBERT models reinforces the hypothesis that binary sentiment polarities in Persian are largely well-separated at approximately 94% accuracy, leaving limited headroom for further augmentation-driven gains. Finally, the retrieval-oriented GTE-Reranker leveraged synthetic diversification effectively, lifting its peak Accuracy from 94.36% (±0.42) to 94.93% (±0.32) and its peak Macro-F1 from 90.42% (±0.65) to 91.49% (±0.57).

Overall, the binary ceiling analysis confirms that LLM-based augmentation yields clear headroom expansions primarily for lightweight or contrastively pre-trained encoders, while standard-capacity models operate near the inherent informational limit of the binary dataset.

4.5 Task granularity and augmentation sensitivity

While binary and multi-class sentiment classification are inherently governed by distinct output spaces and chance levels, a cross-task examination reveals systematic differences in how transformer architectures respond to dataset variants and synthetic enrichment.

A cross-setting overview demonstrates that binary classification operates within a relatively constrained optimization band across all architectures, with model-averaged accuracies ranging from 92.17% on the Original baseline to 93.72% on the Translation regime, alongside Macro-F1 scores spanning 86.20% to 89.40%. In contrast, the five-class task exhibits a substantially wider performance spread and a pronounced trade-off between metrics. While baseline-level accuracy averages 73.24% on the Original partition, Macro-F1 begins at a much lower baseline average of 58.04%, rising to 62.43% under Balanced sampling and peaking at 63.82% in the Translation approach.

This divergent behavior highlights the distinct operational challenges posed by classification granularity. In the binary setup, dominant representations (e.g., E5-General achieving over 96% Accuracy and 94% Macro-F1) operate close to empirical saturation. Because coarse positive-negative polarities are already well-separated in the latent space of pre-trained encoders, the marginal utility of additional generated samples remains bounded. Performance gains in binary configurations are largely driven by structural distribution alignment (such as balanced class representation) rather than massive sample scaling.

Corrected Proof

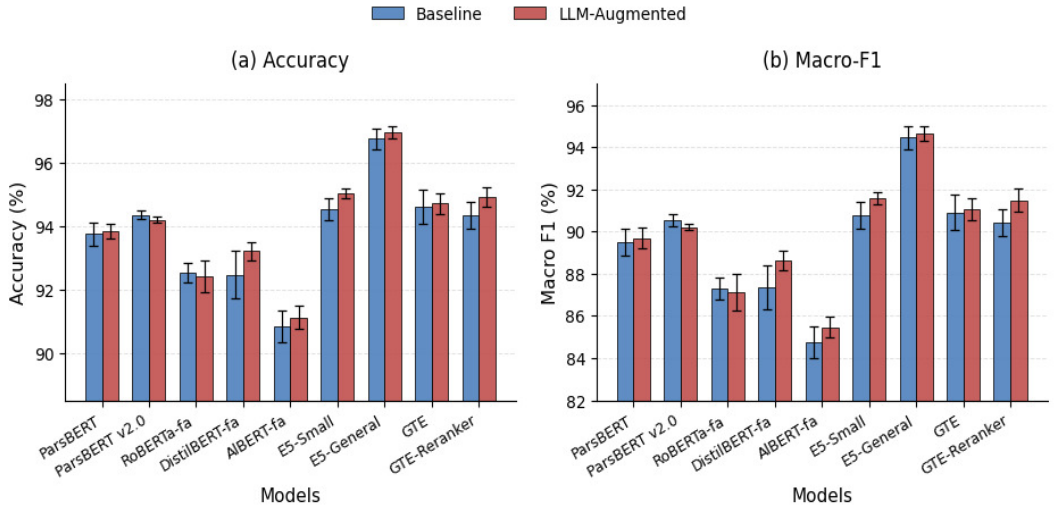


Figure 3: Performance ceilings for the binary sentiment classification. The panels display the peak Accuracy (left) and peak Macro-F1 (right) achieved by each architecture under baseline and LLM-augmented configurations. Similar to the multi-class analysis, peaks are identified independently per metric to represent the maximum empirical capacity of each architecture.

In the five-class setting, separating subtle adjacent categories (e.g., distinguishing neutral from mildly positive sentiments) creates considerable decision boundary overlap. Consequently, baseline models trained on unaugmented data often achieve high overall accuracy by favoring majority classes at the expense of class-averaged balance. Here, synthetic expansion and dataset balancing strategies exert their strongest relative impact. For 8 out of the 9 evaluated architectures, moving from the Original baseline to enriched configurations (Balanced and Translation) yielded substantial Macro-F1 gains (averaging a +5.78 percentage point increase on Translation), effectively stabilizing minority and borderline class representations.

Despite the large differences in task difficulty, architectural hierarchy remains largely preserved across both granularities. Dual-encoder and contrastively pre-trained architectures, most notably E5-General, strictly define the upper empirical boundary in both settings, while lightweight encoders (such as AlBERT-fa and DistilBERT-fa) demonstrate the highest relative responsiveness to generated sample injection when fine-grained boundaries must be resolved.

In summary, classification granularity acts as a primary determinant of LLM-driven data utility. While binary classification provides limited headroom for augmentation-driven expansion due to inherent separability, fine-grained multi-class tasks heavily leverage synthetic linguistic variability to resolve ambiguous decision boundaries and restore metric equilibrium.

4.6 Effect of augmentation size

While the primary objective of this study was not to analyze augmentation size systematically, the Original dataset family provides a natural setting for an exploratory analysis across augmentation scales (+1000, +1146, and +1646 samples). It should be noted that these configurations do not represent an

Corrected

isolated sample-size progression, as the +1146 and +1646 variants also alter the underlying class composition (e.g., targeting specific class imbalances). Consequently, observed performance shifts reflect the combined influence of generated sample volume and targeted class distribution. Figures 4 and 5 show Weighted F1 across these augmentation configurations for all nine models in the five-class and binary settings, respectively. The corresponding Macro F1 results are reported in Tables 3 and 4.

Figure 4 indicates that increasing the training set with augmentation sizes around 1000-1146 samples often improves Weighted F1, although the trend is not strictly monotonic. For several encoder-based classifiers (e.g., ParsBERT, DistilBERT-fa, and RoBERTa-fa), the best Weighted F1 is achieved at moderate augmentation levels (+1000 or +1146), followed by a plateau or a small drop at +1646. The Macro F1 results in Table 3 show a clear model-dependent optimum. ParsBERT and RoBERTa-fa peak at +1146 (60.73 and 59.56, respectively), whereas DistilBERT-fa and AlBERT-fa peak at +1000 (57.36 and 53.80). In contrast, embedding-based models exhibit clearer gains at larger sizes. E5-Small reaches its best performance at +1646 in both Weighted and Macro F1, and GTE shows a marked improvement at the largest augmentation level, with its Macro F1 increasing from 59.45 (*Original*) to 68.49 (*Original*(+1646)). Overall, the lower augmentation levels produced the most reliable gains across a broader range of models, while continued improvement at larger volumes was concentrated in a smaller subset of embedding-based models.

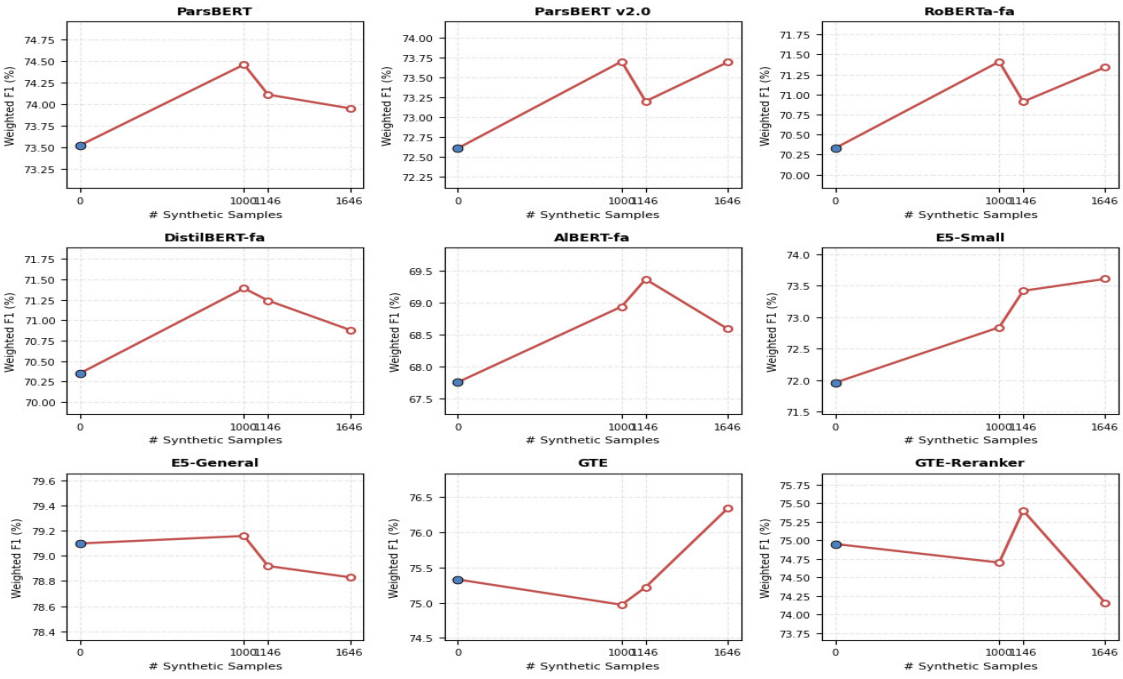


Figure 4: Effect of augmentation size on Weighted F1 for the *Original* dataset family in five-class sentiment classification. Each subplot corresponds to one model, with the x-axis showing the number of generated samples and the y-axis showing Weighted F1 (adjusted per model to highlight within-model trends).

As shown in Figure 5 and Table 4, the binary task exhibits a similar non-monotonic, model-dependent scaling behavior. ParsBERT performs best at +1000 (88.33 Macro F1), and multiple models peak at +1146, including RoBERTa-fa (83.51), DistilBERT-fa (83.35), E5-General (94.67), and GTE-Reranker

Proof

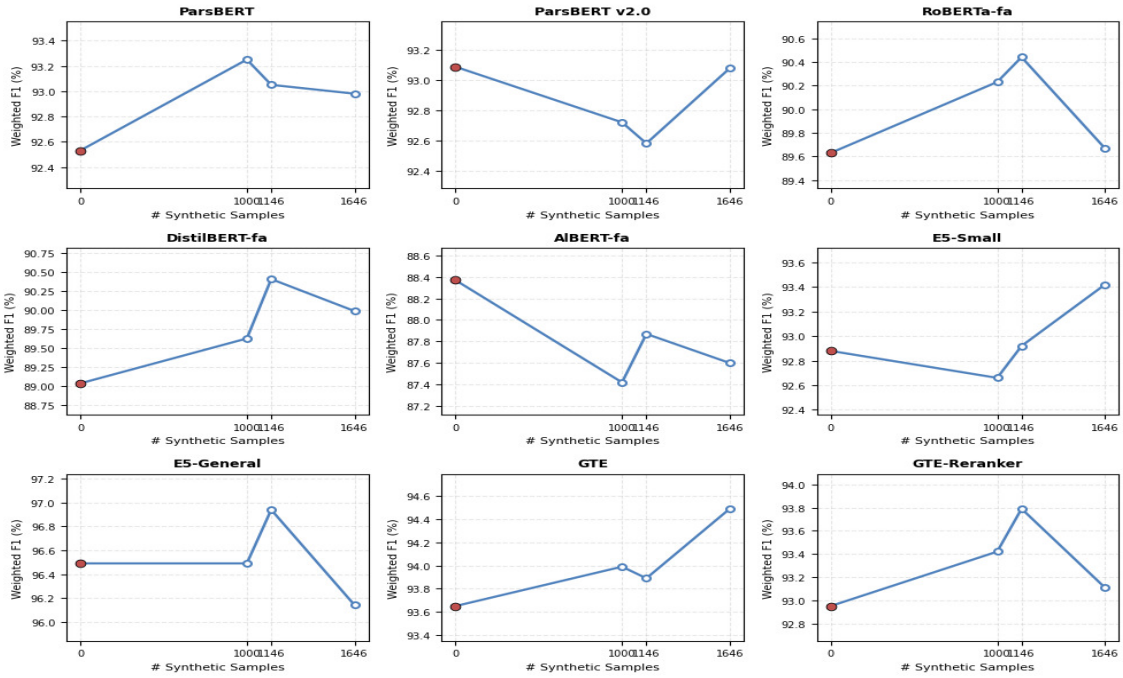


Figure 5: Effect of augmentation size on Weighted F1 for the *Original* dataset family in binary sentiment classification. Each subplot corresponds to one model, with the x-axis showing the number of generated samples and the y-axis showing Weighted F1 (adjusted per model to highlight within-model trends).

(89.21). Increasing the synthetic set to +1646 does not consistently improve these models further. Conversely, E5-Small and GTE show continued improvements up to +1646, reaching 88.49 and 90.35 Macro F1, respectively. Notably, for AlBERT-fa and ParsBERT v2.0, the highest Macro F1 is observed on the unaugmented baseline, suggesting that the net benefit of synthetic augmentation in the binary setting is sensitive to model capacity and the match between synthetic and real data distributions.

Collectively, in this exploratory setting, performance does not increase monotonically across augmentation scales. Adding approximately 1,000 and 1,146 generated samples provides the most reliable gains across tasks and metrics, whereas the largest size (+1646) mainly benefits a subset of embedding-based models (particularly GTE and E5-Small). We emphasize that this analysis remains exploratory, as it is based on three discrete augmentation regimes where sample volume and class composition covary, and is restricted to a single dataset family. Attributing observed variations solely to sample size would be an oversimplification; rather, they stem from the joint interaction between generated volume and class distribution. A purely controlled, single-factor scaling study with additional datasets and finer-grained enhancement levels is left for future work.

4.7 Architectural mechanisms and synthetic data synergy

The empirical findings across both classification tasks highlight distinct interactions between model architectures, pre-training paradigms, and the utility of LLM-generated data. These observations yield three primary mechanistic insights. First, the superior performance and ceiling stability observed in

dense retrieval models (*e.g.*, E5-General and GTE variants) can be attributed to their contrastive pre-training objectives. Unlike standard Masked Language Models (MLMs) such as ParsBERT and RoBERTa-fa, which optimize token-level reconstruction, contrastively trained encoders are explicitly optimized to map sentence-level semantic similarities into a smooth and isotropic embedding space. Consequently, when fine-grained synthetic variations are injected into the training distribution, contrastive architectures are inherently better equipped to separate nuanced class boundaries in challenging five-class sentiment topologies without collapsing neighboring representations.

Second, lightweight and distilled architectures, notably DistilBERT-fa and AlBERT-fa, exhibited substantial relative performance gains under augmented five-class regimes. With fewer parameters, compact models often suffer from representational bottlenecks and are prone to overfitting sparse boundary regions in low-resource or imbalanced Persian benchmarks. LLM-generated samples act as an effective semantic regularizer for these models, densifying the representations around ambiguous boundary classes and mitigating structural bias without requiring architectural expansion.

Finally, the marginal improvements or slight performance degradations observed when augmenting binary datasets, particularly for over-parameterized models, point to an information saturation threshold. In binary sentiment classification, where the latent semantic manifold is linearly or near-linearly separable, standard pre-trained representations already capture sufficient discriminative signals. Under such saturated conditions, injecting generated data can introduce subtle linguistic artifacts or distribution shifts (semantic drift) that act as label noise rather than informative signal, leading to diminishing returns.

To complement the aggregate performance results, Figures 6,7 and 8 present the row-normalized confusion matrices averaged across the three random seeds. These three architectures were selected as representative from our model spectrum, a high-capacity contrastive encoder (GTE), a parameter-efficient compact model (DistilBERT-fa), and a standard pre-trained MLM (ParsBERT v2.0). We evaluate them by comparing the unaugmented *Original* dataset against the *Original*$(+1646)$ configuration to capture the maximal observable impact of LLM augmentation on class decision boundaries. These matrices provide a class-level view of how augmentation changes recall and the distribution of errors.

For GTE (Figure 6), augmentation was associated with a marked increase in recall for the minority *Strong Neg* class, from 0.0% to 30.6%. The most notable accompanying change was a reduction in the proportion of *Strong Neg* instances predicted as *Neg*, from 77.8% to 47.2%. Recall also increased modestly for *Neg* (from 66.5% to 68.5%), *Neutral* (from 80.8% to 81.0%), and *Pos* (from 66.5% to 68.7%). In contrast, recall for *Strong Pos* decreased from 87.0% to 85.2%. Thus, the augmented model recovered a larger share of the *Strong Neg* instances, while the change was not uniform across all classes.

For DistilBERT-fa (Figure 7), the recall of *Neg* increased from 49.1% to 54.2%, and that of *Pos* increased from 60.8% to 62.6%. Recall for *Strong Neg*, previously 0.0%, reached 2.8%. The error distribution for this class also shifted. The proportion of *Strong Neg* instances assigned to the two positive classes (*Pos* and *Strong Pos*) decreased from 50.0% to 27.7%. These gains were accompanied by small decreases in recall for *Neutral* (79.3% to 78.3%) and *Strong Pos* (82.6% to 80.9%), indicating that augmentation did not improve every class equally.

The third matrix (Figure 8) shows a different pattern. The recall of *Strong Neg* increased from 0.0% to 2.8%, however, most instances of this class were still assigned to *Neg*, with that error increasing from 69.4% to 72.2%. At the same time, recall increased for *Neutral* (78.0% to 79.5%) and *Pos* (65.1% to 68.0%), while it decreased for *Neg* (60.5% to 60.1%) and *Strong Pos* (83.2% to 81.2%). The share of

Corrected Proof

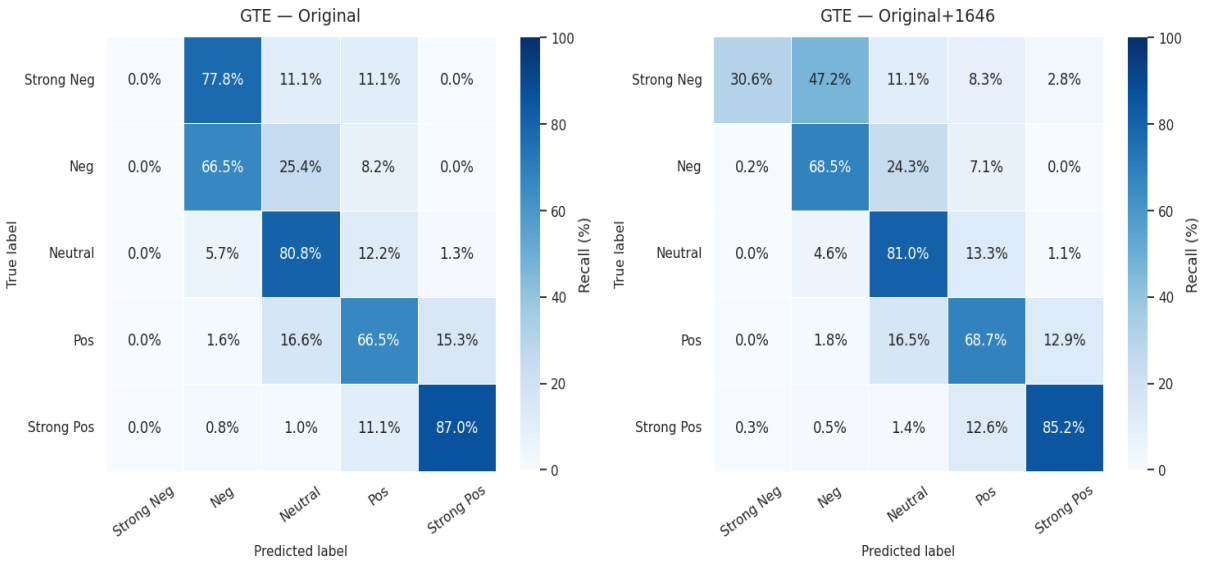


Figure 6: Row-normalized confusion matrices averaged across three random seeds for GTE

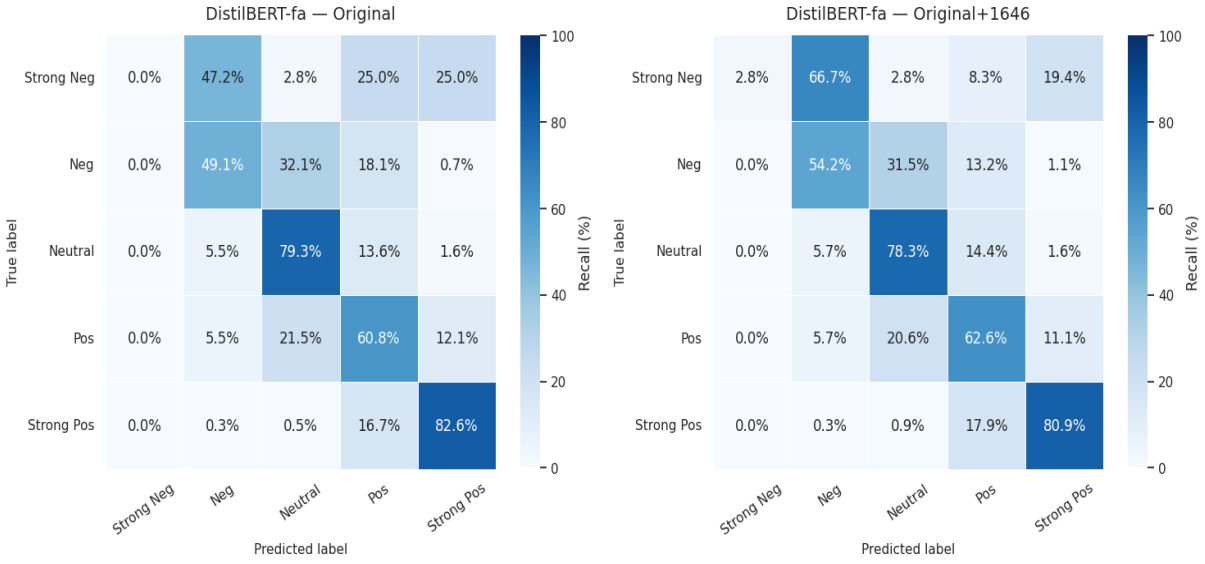


Figure 7: Row-normalized confusion matrices averaged across three random seeds for DistilBERT-fa

Strong Neg instances assigned to the positive classes decreased from 22.2% to 16.6%, but the persistent confusion with *Neg* shows that recognition of this minority class remained limited.

Overall, the matrices indicate that the effect of LLM augmentation is class-dependent. It can improve recall for selected classes and alter specific error patterns, but it does not yield uniform gains across the label set. For completeness, the per-seed confusion matrices and full behavioral profiles for all evaluated architectures are available in the project repository.

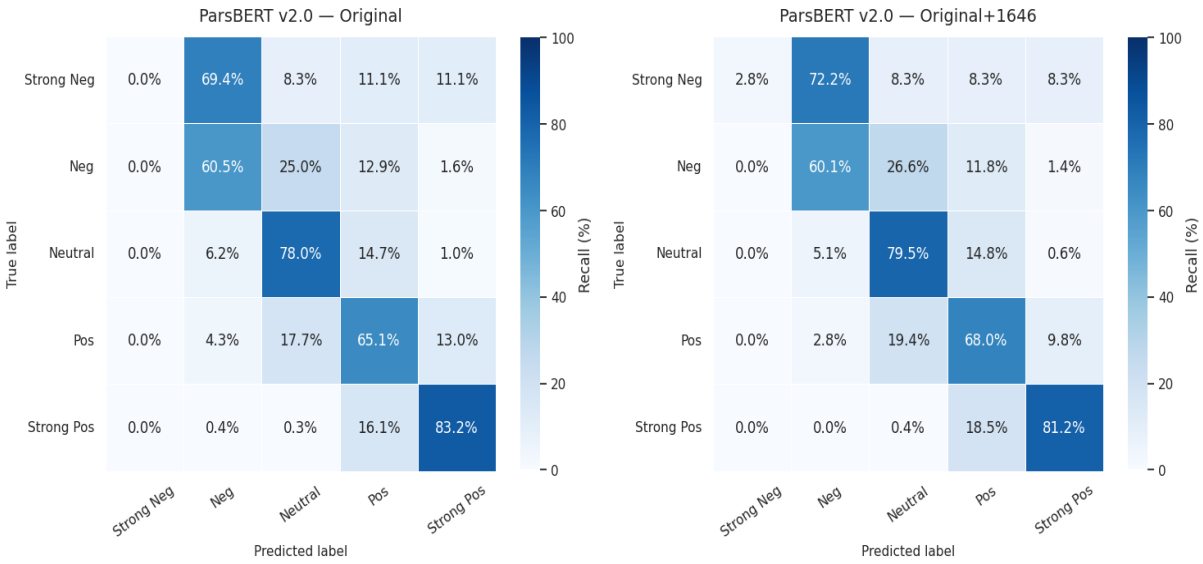


Figure 8: Row-normalized confusion matrices averaged across three random seeds for ParsBERT v2.0

4.8 Statistical significance and effect size analysis

To empirically evaluate whether the performance gains yielded by LLM-generated data were statistically meaningful rather than artifacts of stochastic variance, we conducted rigorous hypothesis testing across all nine model architectures ($N = 9$). Following Demšar [10], we first applied the Shapiro-Wilk test to assess the normality of paired performance differences across multiple benchmarks. If normality was violated, we utilized the non-parametric Wilcoxon signed-rank test as the primary criterion. For normally distributed cases, we used paired two-tailed t-tests instead. To safeguard against errors arising from multiple simultaneous hypotheses, raw p-values were adjusted using the Holm-Bonferroni step-down procedure. Furthermore, because statistical power is inherently constrained by sample size ($N = 9$) and because model architectures evaluated on the same benchmark do not represent strictly independent experimental replications, we treat cross-architecture win rates as descriptive observations. Accordingly, we also report Cohen's d alongside adjusted p-values to distinguish practical effect size magnitude from strict hypothesis-testing significance.

Table 11 summarizes the statistical evaluation and effect size metrics across the five augmentation regimes. Two distinct patterns emerge from the multi-scenario analysis. First, in the five-class classification task, synthetic data injection yields substantial practical improvements, particularly when targeted at class-imbalance mitigation. Under the *Original*(+1000) configuration, the Macro-F1 improvement (+1.96 percentage points) yields observed performance gains across all evaluated architectures (9/9 wins), achieving statistical significance after Holm-Bonferroni correction (p_{adj}=0.0234) alongside a very large effect size (d=2.05). Similarly, for the *Translation*(+1646) scenario, Accuracy and Weighted-F1 show descriptively positive shifts across all nine backbones (9/9 wins, p_{adj}=0.0234, $d \geq 1.37$). For *Original*(+1146) and *Balanced*(+1646), raw paired comparisons often indicate nominal gains ($p_{raw} \leq 0.05$). However, conservative family-wise error adjustments yield non-significant adjusted values. For instance, adjusted p-values reach $p_{adj} \approx 0.061$ for *Original*(+1146) and up to $p_{adj} = 0.442$

Corrected

for *Balanced*(+1646). Nevertheless, the observed effect sizes remain practically meaningful, ranging from moderate to very large across all evaluated metrics and scenarios.

Second, the binary classification task presents a contrasting behavior. Across all five scenarios, performance gains in binary sentiment classification remain modest ($\Delta \leq 0.67$ percentage points). Furthermore, none of the comparisons reach post-correction statistical significance ($p_{adj} \geq 0.30$). This empirical outcome directly mirrors the performance ceiling previously identified in Section 4.4. Baseline binary accuracies were already situated between 92.17% and 93.72%. Consequently, the narrow residual error margin left limited headroom for data augmentation to yield statistically detectable separation across the nine models.

Finally, a comparative view across augmentation scales reveals nuanced dynamics within the multi-class setting. Moving from +1000 to +1146 instances largely sustains positive gains across models. However, expanding further to the +1646 configuration introduces greater cross-model variance and moderates the effect size for Macro-F1 (from d=2.05 to d=0.76). This trend highlights that scaling generated data does not follow a strictly monotonic trajectory. Instead, it exhibits diminishing marginal returns as additional data is incorporated. Full per-model score distributions and intermediate test statistics are documented in the online repository.

5 Limitations

Despite the meaningful empirical findings, this study has several limitations that should be acknowledged and considered when interpreting the results.

First, although our comparative regimes evaluate incremental LLM-generated expansions (up to 1,646 samples), the broader scaling dynamics between augmented corpus size and downstream performance warrant more systematic exploration. Future investigations should conduct extensive data-scaling ablation studies to pinpoint the exact empirical threshold where additional samples cease to yield downstream gains and begin introducing semantic redundancy or label drift. Second, the data generation pipeline relied on two specific general-purpose LLMs (DeepSeek-V3 and GPT-4o mini) under direct zero-shot prompt formulations. Future research should evaluate more capable frontier reasoning models and explore structured, multi-stage generation workflows, such as diversity-aware seed selection, chain-of-thought elicitation, and automated post-generation quality filtering, to systematically maximize semantic entropy and sample efficiency. Third, the empirical experiments in this study focus exclusively on Persian. While the overall methodological framework is language-agnostic, its direct applicability to other low-resource or morphologically complex languages requires empirical validation through cross-lingual transfer studies. Fourth, the benchmark evaluations were conducted within a single specialized domain, namely e-commerce product reviews derived from SentiPers. Sentiment expressions in more informal or highly polarized contexts, such as conversational social media, user-generated comments on news platforms, or colloquial discourse, exhibit distinct linguistic idiosyncrasies, figurative expressions, and syntactic variability that were not captured in this study.

Finally, attributing performance gains solely to semantic enrichment remains challenging, as improvements in low-frequency classes are largely driven by reduced data skewness. Consequently, future research requires controlled ablation frameworks and representation probing to isolate the individual contributions of semantic novelty versus volumetric regularization.

Corrected Proof

Table 11: Cross-architecture statistical significance and effect size evaluation across five augmentation scenarios ($N = 9$ models). p-values correspond to paired two-tailed t-tests (^{t}) for normal differences and Wilcoxon signed-rank tests (^{w}) for non-normal distributions (Shapiro-Wilk $p < 0.05$). p_{adj} reflects Holm-Bonferroni correction within each scenario family ($m = 6$).

Augmentation Scenario	Task	Metric	Gain (Δ)	Wins	Cohen's d	p_{adj}	p_{raw}
Original vs. +1000	Five-class	Acc	+0.54	7/9	0.89	0.1953	0.0391^w
		W-F1	+0.63	7/9	1.01	0.1953	0.0391^w
		M-F1	+1.96	9/9	2.05	0.0234	0.0039^w
	Binary	Acc	+0.16	5/9	0.30	1.0000	0.3921^t
		W-F1	+0.13	5/9	0.23	1.0000	0.5013^t
		M-F1	+0.18	5/9	0.17	1.0000	0.6239^t
Original vs. +1146	Five-class	Acc	+0.64	7/9	1.04	0.0611	0.0140^t
		W-F1	+0.65	7/9	1.07	0.0611	0.0122^t
		M-F1	+2.47	8/9	1.11	0.0610	0.0102^t
	Binary	Acc	+0.35	6/9	0.62	0.3052	0.1017^t
		W-F1	+0.36	7/9	0.58	0.3052	0.1190^t
		M-F1	+0.65	7/9	0.55	0.3052	0.1367^t
Original vs. +1646	Five-class	Acc	+0.55	7/9	0.75	0.2569	0.0537^t
		W-F1	+0.61	7/9	0.82	0.2381	0.0397^t
		M-F1	+2.15	8/9	0.76	0.1172	0.0195^w
	Binary	Acc	+0.22	6/9	0.44	0.6559	0.2186^t
		W-F1	+0.21	6/9	0.37	0.6559	0.2988^t
		M-F1	+0.34	6/9	0.31	0.6559	0.3759^t
Balanced vs. +1646	Five-class	Acc	+0.48	6/9	0.55	0.4424	0.1366^t
		W-F1	+0.52	6/9	0.59	0.4424	0.1123^t
		M-F1	+1.33	7/9	0.86	0.2002	0.0334^t
	Binary	Acc	+0.47	7/9	0.65	0.4424	0.0885^t
		W-F1	+0.43	7/9	0.62	0.4424	0.1017^t
		M-F1	+0.67	7/9	0.59	0.4424	0.1168^t
Translation vs. +1646	Five-class	Acc	+0.82	9/9	1.37	0.0234	0.0039^w
		W-F1	+0.82	9/9	1.52	0.0234	0.0039^w
		M-F1	+0.74	5/9	0.44	0.5067	0.2199^t
	Binary	Acc	+0.16	7/9	0.45	0.5067	0.2145^t
		W-F1	+0.18	7/9	0.51	0.5067	0.1678^t
		M-F1	+0.33	7/9	0.57	0.5067	0.1267^t

6 Conclusion

This study provided a comprehensive empirical evaluation of LLM-generated data augmentation for Persian sentiment classification across nine transformer architectures and multiple training configurations. By leveraging the parametric knowledge of frozen, instruction-tuned LLMs via a prompt-driven paradigm, we demonstrated that high-quality text samples can be synthesized without expensive model fine-tuning or cumbersome rule-based methods. Crucially, our findings reveal that the downstream impact of generative augmentation is not uniform across tasks and models, but rather intrinsically tied to task granularity and baseline representation.

Across multi-seed evaluations, LLM augmentation substantially enhanced classification accuracy across the majority of architectures in the challenging five-class setting (up to ≈2.0% depending on model architecture), improving performance in up to 8 out of 9 evaluated models. Fine-grained confusion matrix analyses revealed that generated samples effectively mitigated boundary ambiguities within underrepresented extreme sentiment classes (e.g., strong negative and strong positive). In contrast, binary sentiment classification exhibited a distinct performance ceiling, where mature contextual repre-

Corrected Proof

sentations left limited headroom for additional data intervention, occasionally yielding negligible fluctuations or marginal declines. Rigorous paired hypothesis testing further contextualizes these observations, establishing that while high baseline scores in binary tasks dilute statistical divergence, the practical significance in multi-class configurations remains pronounced. In addition, intrinsic corpus audits revealed high lexical diversity without evidence of redundant memorization. Furthermore, generative augmentation acted as a practical complement to conventional strategies such as class-balancing and back-translation, demonstrating that targeted sample injection via frozen, tuning-free models can reinforce structured augmentation regimes in low-resource environments.

Future research will extend this framework toward design-science principles, specifically investigating diversity-aware seed selection algorithms to optimize sample efficiency without inflating dataset volume. In addition, assessing cross-domain portability across varied low-resource NLP benchmarks and developing automated quality-filtering pipelines will offer deeper mechanistic insights into curated data generation.

Conflicts of interest

The authors declare that there are no conflicts of interest.

References

[1] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F.L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, R. Avila, GPT-4 technical report, arXiv:2303.08774, 2023.

[2] S. Ahmadnia, A.Y. Jordehi, M.H.K. Heyran, S. Mirroshandel, O. Rambow, Opinion mining using pre-trained large language models: Identifying the type, polarity, intensity, expression, and source of private states, in: Proc. 2024 Joint Int. Conf. Comput. Linguist. Lang. Resour. Eval. (LREC-COLING 2024) (2024) 12481–12495.

[3] S. Ahmadnia, A.Y. Jordehi, M.H.K. Heyran, S.A. Mirroshandel, O. Rambow, C. Caragea, Active few-shot learning for text classification, in: Proc. 2025 Conf. Nations Am. Chapter Assoc. Comput. Linguist.: Hum. Lang. Technol. (Vol. 1: Long Papers) (2025) 6677–6694.

[4] A. Areshey, H. Mathkour, Exploring transformer models for sentiment classification: A comparison of BERT, RoBERTa, ALBERT, DistilBERT, and XLNet, Expert Syst. **41** (2024) e13701.

[5] M. Bayer, M.A. Kaufhold, B. Buchhold, M. Keller, J. Dallmeyer, C. Reuter, Data augmentation in natural language processing: a novel text generation approach for long and short text classifiers, Int. J. Mach. Learn. Cybern. **14** (2023) 135–150.

[6] Y. Chai, H. Xie, J.S. Qin, Text data augmentation for large language models: A comprehensive survey of methods, challenges, and opportunities, Artif. Intell. Rev. **59** (2026) 35.

[7] J. Chen, D. Tam, C. Raffel, M. Bansal, D. Yang, An empirical survey of data augmentation for limited data learning in NLP, Trans. Assoc. Comput. Linguist. **11** (2023) 191–211.

Corrected Proof

[8] R. Cheruku, K. Hussain, I. Kavati, A.M. Reddy, K.S. Reddy, Sentiment classification with modified RoBERTa and recurrent neural networks, Multimed. Tools Appl. **83** (2024) 29399–29417.

[9] O. Davar, G. Dar, F. Ghasemian, DeepSentiParsBERT: A deep learning model for Persian sentiment analysis using ParsBERT, in: Proc. 28th Int. Comput. Conf., Comput. Soc. Iran (CSICC), IEEE (2023) 1–5.

[10] J. Demšar, Statistical comparisons of classifiers over multiple data sets, J. Mach. Learn. Res. **7** (2006) 1–30.

[11] J. Devlin, M.W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: Proc. 2019 Conf. North Am. Chapter Assoc. Comput. Linguist.: Hum. Lang. Technol., Vol. 1 (Long and Short Papers) (2019) 4171–4186.

[12] B. Ding, C. Qin, R. Zhao, T. Luo, X. Li, G. Chen, W. Xia, J. Hu, A.T. Luu, S. Joty, Data augmentation using large language models: Data perspectives, learning paradigms and challenges, arXiv:2403.02990, 2024.

[13] M. Fadaee, A. Bisazza, C. Monz, Data augmentation for low-resource neural machine translation, arXiv:1705.00440, 2017.

[14] M. Farahani, M. Gharachorloo, M. Farahani, M. Manthouri, ParsBERT: Transformer-based model for Persian language understanding, Neural Process. Lett. **53** (2021) 3831–3847.

[15] R. Ghasemi, S.A. Ashrafi Asli, S. Momtazi, Deep Persian sentiment analysis: Cross-lingual training for low-resource languages, J. Inf. Sci. **48** (2022) 449–462.

[16] P. Hosseini, A.A. Ramaki, H. Maleki, M. Anvari, S.A. Mirroshandel, SentiPers: a sentiment analysis corpus for Persian, arXiv:1801.07737, 2018.

[17] A.Y. Jordehi, M.H.K. Heyran, S. Ahmadnia, S.A. Mirroshandel, O. Rambow, Improving opinion mining through automatic prompt construction, J. Inf. Syst. Telecommun. (JIST) **3** (2024) 216.

[18] A.Y. Jordehi, S.A. Mirroshandel, O. Rambow, Label-Aware Pseudo-Training Sample Generation for Text Classification, J. Artif. Intell. Res. **85** (2026).

[19] L. Lakshmi, A.B.M. Ali, K.D. Sree Devi, M. Rafiq, I. Shernazarov, N.A. Othman, M.I. Khan, Opinion mining in e-commerce: Evaluating machine learning approaches for sentiment analysis, Results Control Optim. **19** (2025) 100575.

[20] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, R. Soricut, ALBERT: A lite BERT for self-supervised learning of language representations, arXiv:1909.11942, 2019.

[21] M. Li, H. Chen, Y. Wang, T. Zhu, W. Zhang, K. Zhu, K.F. Wong, J. Wang, Understanding and mitigating the bias inheritance in LLM-based data augmentation on downstream tasks, arXiv:2502.04419, 2025.

[22] Z. Li, X. Zhang, Y. Zhang, D. Long, P. Xie, M. Zhang, Towards general text embeddings with multi-stage contrastive learning, arXiv:2308.03281, 2023.

Corrected Proof

[23] A. Liu, B. Feng, B. Xue, B. Wang, B. Wu, C. Lu, C. Zhao, C. Deng, C. Zhang, C. Ruan, C. Dai, D. Dai, Deepseek-v3 technical report, arXiv:2412.19437, 2024.

[24] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, RoBERTa: A robustly optimized BERT pretraining approach, arXiv:1907.11692, 2019.

[25] A. Magueresse, V. Carles, E. Heetderks, Low-resource languages: A review of past work and future challenges, arXiv:2006.07264, 2020.

[26] J. Panduro-Ramirez, Sentiment analysis in customer reviews for product recommendation in e-commerce using machine learning, in: Proc. 2024 Int. Conf. Adv. Comput. Commun. Appl. Inform. (ACCAI), IEEE (2024) 1–5.

[27] B. Pavlyshenko, M. Stasiuk, Using Large Language Models for Data Augmentation in Text Classification Models, Int. J. Comput. **24** (2025) 148–154.

[28] F. Pedregosa, K. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, Scikit-learn: Machine learning in Python, J. Mach. Learn. Res. **12** (2011) 2825–2830.

[29] D.M. Powers, Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation, arXiv:2010.16061, 2020.

[30] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, Improving language understanding by generative pre-training, 2018, https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf.

[31] Ł. Radliński, M. Guściora, J. Kocoń, Backtranslation and paraphrasing in the LLM era? comparing data augmentation methods for emotion classification, in: Proc. Int. Conf. Comput. Sci., Springer Nature Switzerland, Cham (2025) 3–17.

[32] F.L. Richardson, A. Villavicencio, R. Menezes, The rephrased reality: Analysing sentiment shifts in LLM-rephrased text, in: Proc. 40th ACM/SIGAPP Symp. Appl. Comput. (2025) 920–927.

[33] G.G. Şahin, To augment or not to augment? A comparative study on text augmentation techniques for low-resource NLP, Comput. Linguist. **48** (2022) 5–42.

[34] V. Sanh, L. Debut, J. Chaumond, T. Wolf, DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter, arXiv:1910.01108, 2019.

[35] J.P.R. Sharami, P.A. Sarabestani, S.A. Mirroshandel, DeepSentiPers: Novel deep learning models trained over proposed augmented Persian sentiment corpus, arXiv:2004.05328, 2020.

[36] F. Soleimanzadeh, M.J. Shayegan, Sentiment analysis in Persian language using long short-term memory and deep learning, in: Proc. 11th Int. Conf. Web Res. (ICWR), IEEE (2025) 597–603.

[37] A. Taheri, A. Zamanifar, A. Farhadi, Enhancing aspect-based sentiment analysis using data augmentation based on back-translation, Int. J. Data Sci. Anal. **19** (2025) 491–516.

Corrected Proof

[38] S. Ubani, S.O. Polat, R. Nielsen, Zeroshotdataaug: Generating and augmenting training data with chatgpt, arXiv:2304.14334, 2023.

[39] Y.Z. Vakili, A. Fallah, S. Zakeri, Enhancing sentiment analysis of Persian tweets: A transformer-based approach, in: Proc. 10th Int. Conf. Web Res. (ICWR), IEEE (2024) 226–230.

[40] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Adv. Neural Inf. Process. Syst. **30** (2017).

[41] L. Wang, N. Yang, X. Huang, B. Jiao, L. Yang, D. Jiang, R. Majumder, F. Wei, Text embeddings by weakly-supervised contrastive pre-training, arXiv:2212.03533, 2022.

[42] Z. Wang, P. Wang, K. Liu, P. Wang, Y. Fu, C.T. Lu, C.C. Aggarwal, C. Pei, Y. Zhou, A comprehensive survey on data augmentation, arXiv:2405.09591, 2024.

[43] M. Wankhade, A.C.S. Rao, C. Kulkarni, A survey on sentiment analysis methods, applications, and challenges, Artif. Intell. Rev. **55** (2022) 5731–5780.

[44] Z. Xie, S.I. Wang, J. Li, D. Lévy, A. Nie, D. Jurafsky, A.Y. Ng, Data noising as smoothing in neural network language models, arXiv:1703.02573, 2017.

[45] V. Yadav, Z. Tang, V. Srinivasan, PAG-LLM: Paraphrase and aggregate with large language models for minimizing intent classification errors, in: Proc. 47th Int. ACM SIGIR Conf. Res. Dev. Inf. Retr. (2024) 2569–2573.

[46] S. Zhang, L. Dong, X. Li, S. Zhang, X. Sun, S. Wang, J. Li, R. Hu, T. Zhang, G. Wang, F. Wu, Instruction tuning for large language models: A survey, ACM Comput. Surv. **58(7)** (2026) 1–36.

Corrected Proof