Artificial Conversations, Real Results: Fostering Language Detection with Synthetic Data

Fatemeh Mohammadi[1], Tommaso Romano[1], Samira Maghool[1] and Paolo Ceravolo[1,*]

[1]*University of Milan, Italy*

Abstract

Collecting high-quality training data is essential for fine-tuning Large Language Models (LLMs). However, acquiring such data is often costly and time-consuming, especially for non-English languages such as Italian. Recently, researchers have begun to explore the use of LLMs to generate synthetic datasets as a viable alternative. This study proposes a pipeline for generating synthetic data and a comprehensive approach for investigating the factors that influence the validity of synthetic data generated by LLMs by examining how model performance is affected by metrics such as prompt strategy, text length and target position in a specific task, i.e. inclusive language detection in Italian job advertisements. Our results show that, in most cases and across different metrics, the fine-tuned models trained on synthetic data consistently outperformed other models on both seed and synthetic test datasets. The study discusses the practical implications and limitations of using synthetic data for language detection tasks with LLMs.

Keywords

Large Language Models, synthetic data, inference, finetuning

1. Introduction

In recent years, Large Language Models (LLMs) have received considerable attention in language recognition tasks. However, the effectiveness of these models largely depends on appropriate fine-tuning procedures, which require access to large, diverse, and high-quality datasets for training and evaluation. Obtaining such datasets is challenging due to issues such as data scarcity, privacy constraints, class imbalance, an absence of edge cases, and the high costs of collection and annotation [1].

Synthetic data has been proposed as a potential solution to address certain challenges associated with language detection tasks, including limited data availability [2] and the psychological impact on annotators [3]. Synthetic data helps to overcome these limitations by offering greater control over data properties, allowing tailored augmentation for specific tasks, such as testing models under different conditions or rare scenarios [4].

It also accelerates the iterative model development process by providing readily available and customizable datasets [5]. In addition, synthetic data helps mitigate biases present in real-world datasets [6] and improves model generalization by exposing algorithms to a wider range of input variations [7]. This approach is particularly valuable in areas such as healthcare [8], complex systems [9], and natural language processing [10], where obtaining high-quality labeled data is often difficult or resource-intensive.

The capabilities of language models for most language detection tasks have been extensively discussed [11, 12]. However, an investigation of the utility of LLMs for detecting inclusive language is still lacking. In particular, an in-depth evaluation of the capabilities of such models, e.g. their ability to achieve high performance even when fine-tuned using synthetic data. Therefore, this paper aims to address the challenges associated with acquiring high-quality training data for fine-tuning LLMs, especially in low-resource language settings, such as many non-English languages. In this paper, we address these challenges through the following contributions: (1) Proposing a synthetic data generation pipeline to

TRUST-AI: The European Workshop on Trustworthy AI. Organized as part of the European Conference of Artificial Intelligence - ECAI 2025. October 2025, Bologna, Italy.

[*] Corresponding author.

fatemeh.mohammadi@unimi.it (F. Mohammadi); tommaso.romano@unimi.it (T. Romano); samira.maghool@unimi.it (S. Maghool); paolo.ceravolo@unimi.it (P. Ceravolo)

0000-0002-3100-7569 (F. Mohammadi); 0000-0001-8310-2050 (S. Maghool); 0000-0002-4519-0173 (P. Ceravolo)

address data scarcity in low-resource language settings. (2) Outline a workflow that involves fine-tuning an LLM on synthetic training data, followed by inference with fine-tuned and pre-trained models on synthetic test data to evaluate the effectiveness of synthetic data. (3) Focusing on inclusive language detection, an under-researched and challenging task, especially in gendered languages such as Italian. (4) Demonstrate the potential of synthetic data as a cost-effective, scalable solution by showing that fine-tuned models trained on synthetic data outperform other models on both seed and synthetic test data.

2. Background and Related Works

2.1. Synthetic data generation

Synthetic data generation has become an essential approach to mitigate challenges related to data scarcity, privacy, and the need for diverse datasets when training machine learning models [13]. Different techniques, including Generative Adversarial Networks (GANs), Variational Autoencoders (VAEs), and LLMs, offer different capabilities tailored to specific data types. GANs are particularly effective for generating tabular data, while VAEs are widely used for generating synthetic images and tabular datasets [14].

For synthetic text data, LLMs are the most suitable choice. These models produce coherent, contextually accurate text that is virtually indistinguishable from human-written content, making them ideal for natural language processing tasks [15]. By providing well-crafted prompts or instructions, LLMs can generate diverse and realistic textual data, such as dialogues, narratives, and domain-specific content [14]. Such synthetic data can be used for a variety of purposes, including fine-tuning LLMs, increasing the diversity of datasets, and simulating scenarios for testing and development.

2.2. Using synthetic data for language detection tasks

Several studies have explored using synthetic data for language detection tasks. Casula et al. [16] showed that instruction-based rewriting of original English texts can produce synthetic data effective enough to train classifiers that perform as well as, or better than, those trained on original data. Maheshwari et al. [17] demonstrated that synthetic data properties can predict performance in intent detection tasks. Khullar et al. [18] generated synthetic data for hate speech detection in low-resource languages, such as Hindi and Vietnamese, using machine translation and contextual entity substitution, finding that models trained on synthetic data can match or exceed the performance of those trained on available target-domain examples.

Despite existing research on the use of synthetic data for tasks such as detecting hate speech and abusive language, there is a notable lack of focus on inclusive language. Specifically, in the context of generating synthetic data for job advertisements, to our knowledge, only one study has been developed [19]. This study presented *SkillSkape*, an open-source synthetic dataset of job advertisements designed for skill-matching tasks rather than comprehensive language detection. Due to the gendered nature of the Italian language and the lack of research on this topic, our study is highly novel and fills an important gap in the field.

2.3. Fine-tuning LLMs using synthetic data

The use of LLMs to generate synthetic data is growing; however, there is limited research on fine-tuning LLMs specifically using synthetic data. One notable study [20] introduced Generalised Instruction Tuning (GLAN), a method for generating synthetic data tailored to instruction tuning tasks. Extensive experiments on the Mistral model showed that training with GLAN enabled the model to excel in several domains, including mathematical reasoning, coding, academic exams, logical reasoning, and general instruction. Remarkably, this was achieved without the use of task-specific training data for these applications. The study most closely related to our methodology is [21], which fine-tuned a model using a combination of seed and synthetic data. However, their focus differs, as their synthetic data relates to therapy sessions. Their experimental results showed that the hybrid model consistently outperformed others in specific vertical applications, achieving superior performance across all metrics. Additional tests confirmed the hybrid model's enhanced adaptability and contextual understanding in different scenarios. These results highlight the potential of combining seed and synthetic data to improve LLMs' robustness and contextual sensitivity, particularly for domain-specific and specialized applications.

3. Methodology

This section outlines our proposed approach for developing a framework driven by LLMs, designed to generate synthetic data and evaluate it using different prompt strategies across different language models. In this study, we used this framework to detect non-inclusive language in Italian job advertisements. Our approach consists of four main parts: (i) creation of a synthetic dataset by combining real and generated data, (ii) study of different prompting techniques and their impact on the performance of pre-trained models, (iii) fine-tuning of a model on the synthetic data, (iv) and finally inference using our fine-tuned model and other pre-trained models, on both synthetic and seed test datasets. Figure 1 shows an overview of the entire framework.

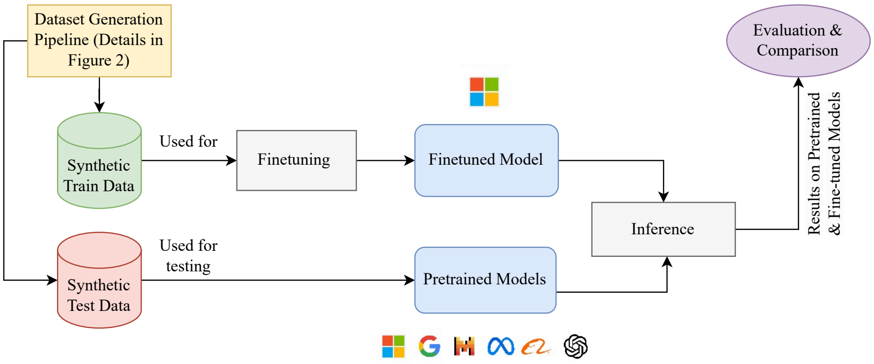


Figure 1: An overview of the entire framework of using synthetic data in inclusive language detection.

3.1. Synthetic data generation

Synthetic data generation is a key part of our methodology, enabling the creation of large datasets for both fine-tuning and evaluation purposes. A major challenge in this process is the risk of repetition within the dataset, where variation is limited to minor changes in the words that replace the placeholders. To mitigate this, we implemented a novel strategy: pre-splitting the templates into separate training and test sets before data generation. By separating the templates at the outset, we ensure that the test set remains sufficiently distinct from the training set, reducing the overlap of similar sentence structures. This approach minimizes overfitting and improves the overall robustness and performance of our fine-tuned model.

As shown in Figure 2, we allocate 70% of the dataset for training and the remaining 30% for evaluation. By using this partitioning strategy, we achieve a balance between optimizing the model's learning process and validating its generalization capabilities, thereby reducing the risk of data leakage. This ensures a robust and reliable evaluation of the model's performance on previously unseen data.

The process begins with a seed dataset composed of 99 real-world job descriptions, manually collected and annotated by domain experts in law and linguistics, which are then deconstructed into individual sentences (No. 1). Sentences containing words that can be masked are identified and reused as templates for dataset construction. Each sentence is given a binary label: TODO for sentences with maskable words that require further processing, and INCLUSIVE for sentences that are inherently neutral and cannot be discriminated. For example, a sentence like "Sarai [VERB] per un colloquio conoscitivo" is labeled TODO, while a neutral sentence like "Descrizione del ruolo:" is categorized as INCLUSIVE (No. 2).

Research has shown that text length plays a crucial role in shaping the performance and behavior of LLMs, affecting aspects such as coherence, contextual understanding, and generation quality [22]. A notable advantage of our synthetic data generation pipeline is the inclusion of a *chunk merger* module (No. 3), which allows precise control over text length. This feature allows the creation of synthetic data sets of different lengths, facilitating in-depth analysis of how text length affects LLM performance. We create templates containing placeholders for *job titles*, *work-related adjectives*, and *verbs*, categorized *by gender*.

In the next step (No. 4), we replace the placeholders in the annotated dataset with a corresponding word from a substitution list, where each vocabulary is labeled as `neutral`, `masculine`, or `feminine`. This list is validated by human native annotators to check if the genders are correctly assigned. This substitution process generates a large number of possible text combinations (*Chunk Merger* module). The generated dataset can be found here.

After generating synthetic test data, the next step is to generate labeled training data, which will be used for fine-tuning the LLM. To achieve this, the *Response Maker* module (No. 5) is used to assign labels to each sentence generated by the *Data Generator* module. Sentences containing only one label `masculine` or `feminine` are classified as NONINCLUSIVE, while those containing only `neutral` elements are classified as INCLUSIVE. For example, if the placeholder `[JOB]` in the template sentence "`[JOB]` will play a key role (In Italian: svolgerà un ruolo chiave)" is replaced by `insegnante` (teacher), the sentence is marked as INCLUSIVE, because `insegnante` is a gender-neutral term in Italian. Conversely, replacing `[JOB]` with `infermiere` (nurse) results in a label of NONINCLUSIVE, as `infermiere` is a masculine term that excludes female candidates. This systematic labeling approach, where labels are directly tied to gender references (i.e., masculine/feminine forms as non-inclusive and neutral forms as inclusive) and validated by human annotators, ensures the accurate identification of inclusive language within the dataset.

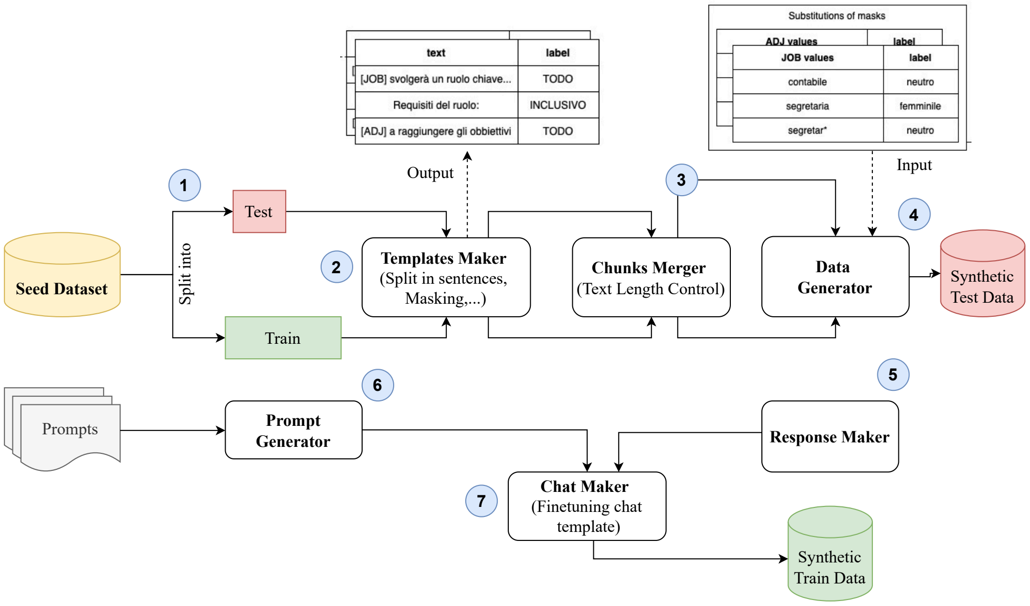
To fine-tune the LLM using chat template data, we use the *Prompt Generator* module along with the labeled sentences from the previous step. By feeding these two inputs into the *Chat Maker* module (No. 7), we generate 10,424 rows of data, which will serve as the training data set for fine-tuning the LLM. The details of the *Prompt Generator* module will be discussed in the next section.

3.2. Prompts generation based on different approaches

One of the primary objectives of this study is to examine the impact of various prompting methods on the responses generated by LLMs, with the ultimate goal of enhancing response quality. Recent research has shown that even small variations in prompting - such as rephrasing or changing the structure - can have a significant impact on LLM performance [23]. For example, strategies such as encouraging step-by-step reasoning or rephrasing objectives can lead to markedly different outcomes [24]. To streamline this exploration, we are implementing an automated system for designing and managing prompts using a modular prompting framework. This framework includes methods such as *zero-shot learning* (ZSL), *few-shot learning* (FSL), and ZSL with *chain-of-thought* strategy, which we call (ZSLCOT) prompting [25]. Using these prompting methods, we generated four different prompts: two for ZSL and one for each of the other strategies. The results provide valuable insights into how to design prompts to maximize LLM performance and response quality effectively.

3.3. LLM fine-tuning

We fine-tuned a pre-trained language model using the *Unsloth* library, known for its powerful fine-tuning capabilities and accelerated processing speed [26]. For this study, we chose the *Phi3-mini* model, a compact and cost-effective architecture with 29,884,416 trainable parameters. To further optimize efficiency, we used a 4-bit quantized version of the pre-trained model, which significantly reduced computational requirements and processing time without compromising performance. The fine-tuning



text	label
[JOB] svolgerà un ruolo chiave...	TODO
Requisiti del ruolo:	INCLUSIVO
[ADJ] a raggiungere gli obbiettivi	TODO

Substitutions of masks

ADJ values	label

JOB values	label
contabile	neutro
segretaria	femminile
segretar*	neutro

Figure 2: Synthetic data generation workflow- The sentences in the output table of the **Template Maker** module represent examples of job descriptions and maskings used for data generation, while the text in the input table of the **Data Generator** module consists of Italian job titles and adjectives annotated with their corresponding genders.

dataset consisted of 5,712 synthesized samples formatted as chat data containing questions, text, and responses. These samples were tokenized using a custom chat template designed specifically for the *Phi-3* architecture to ensure compatibility and maximize the effectiveness of the fine-tuning process.

The tuning used *Parameter-Efficient Fine-Tuning* (PEFT) with a LoFTQ configuration [27], which allows tuning without changing all model parameters, making the process more resource efficient. Using *SFTTrainer* from the Hugging Face library [28], a single epoch of 360 training steps was run on a Tesla T4 GPU with 14.748 GB of RAM, achieving a speed of 0.28 iterations per second and completing in 26.55 minutes. The resulting fine-tuned model was then uploaded to a model hub and made available for further use.

3.4. Inference using fine-tuned and pre-trained models

The inference process defined in this study refers to the procedure by which both pre-trained and fine-tuned language models generate responses based on input data. This process is applied consistently to both the synthetic test dataset and the manually annotated real-world dataset, ensuring a consistent evaluation framework. This dual evaluation approach minimizes the risk of overfitting by validating model performance on different data sources. Answers are generated using the automatically generated prompts, as described in Section 3.2.

We compared the results obtained by our fine-tuned model with five different LLMs: *LLaMA* 3 7B from Meta, *Phi-3-mini* 3.8B from Microsoft, *Mistral* 7B, *Qwen* 2 7B from Alibaba and *Gemma* 2 9B from Google. The *Phi-3-mini* used in the comparison is different from the *Phi-3-mini* we have fine-tuned. Our data collection yielded a substantial dataset of 10, 424 responses, for a total of 62, 544 data points. This large dataset enables a thorough evaluation of each model, prompt, and inferred label for the examples. The code developed in this study for prompt construction, data generation, and model finetuning is available in this GitHub repository.

3.5. Comprehensive comparative analysis

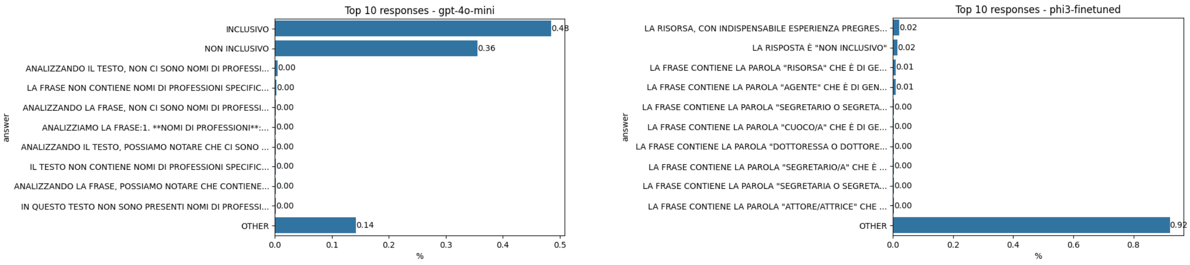
In this paper, the detection of non-inclusive language is framed as a binary classification problem, and we evaluate the models from several perspectives. Our evaluation includes the following key aspects. (i) The *structure of the top responses* generated by different LLMs, highlighting how well the output matches the provided prompts. (ii) A comparison of the *accuracy of different prompt strategies* applied to both synthetic and seed datasets, identifying the most effective models and prompts for each. (iii) Having identified the best-performing prompt for each model, we evaluate *model performance* using these optimized prompts. We use *precision*, *recall*, *specificity*, *F1 score*, and *accuracy* as the primary metrics. However, due to the issue of data imbalance, often cited in the literature as a source of metric skewness and potential bias, we also include *balanced accuracy* (bACC) as a more reliable metric [29].

4. Results and discussion

4.1. Evaluation of the structure of the produced responses

After generating responses using LLMs, the first step is to pre-process the output to ensure that the responses are standardized. The primary objective is to extract the binary labels - `INCLUSIVE` and `NON-INCLUSIVE` - from these responses. This pre-processing step is essential because the generated output often contains extraneous information, such as reasoning and explanations, which must be removed to produce a clean set of responses for accurate analysis.

Figures 3a and 3b show the top 10 responses from *GPT-4o-mini* and fine-tuned *Phi3* models as examples evaluated on the same test dataset. While the pre-trained models such as *GPT-4o-mini* produce output in the correct format, the fine-tuned model often introduces additional noise, such as explanations or special characters. To handle the varying outputs, several preprocessing functions were implemented to extract the desired labels. As a result, 99.90% of the responses were successfully transformed into the desired format through automated preprocessing, with less than 0.1% of responses falling outside the expected format.



(a) Top 10 responses by pre-trained GPT-4o-mini.

(b) Top 10 responses by finetuned Phi-3.

Figure 3: Comparison of top responses produced by pre-trained vs. finetuned models.

Another key observation is the class imbalance in the responses generated. The distribution of responses generated by the LLMs for the synthetic dataset shows that `INCLUSIVE` labels appear twice as often as `NON INCLUSIVE` labels. This imbalance supports the choice of *balanced accuracy* (bACC) as a more appropriate evaluation metric.

4.2. Evaluation of the prompting strategies

As discussed in the methodology section, the quality of the prompt is crucial for obtaining optimal responses from both pre-trained and fine-tuned LLMs. To save time and computational resources, we decided to use a single prompt strategy during the inference process. Before starting, we automatically compared the different prompts generated by our pipeline based on their accuracy to select the best one. The base ZSL prompt employed was: "You are an assistant who reads and analyzes sentences from Italian job advertisements. Your task is to determine whether the text contains profession names and whether the sentence refers to both genders. Respond only with the label 'NON INCLUSIVE' or 'INCLUSIVE'." For other prompting strategies, such as FSL, this basic ZSL prompt was extended by incorporating illustrative examples or by specifying more detailed instructions. Table 1, shows the accuracy of our four tested prompts for the synthetic and seed datasets, respectively.

Table 1
Accuracy comparison between prompts for different models across Synthetic and Seed datasets.

Dataset	Model	FSL#0	ZSL#0	ZSL#1	ZSLCOT#0
Synthetic	phi3_finetuned	**0.973**	**0.976**	**0.921**	**0.991**
	gpt_40_mini	**0.888**	0.810	0.631	0.820
	phi3	**0.508**	0.504	0.475	0.499
	llama3	**0.563**	0.502	0.546	0.526
	mistral	0.503	0.510	0.502	**0.520**
	gemma2	**0.598**	0.553	0.431	0.537
	qwen2	0.511	**0.580**	0.471	0.579
Seed Data	phi3_finetuned	**0.642**	**0.702**	**0.647**	**0.677**
	gpt_40_mini	0.565	**0.647**	0.595	0.585
	phi3	**0.512**	0.500	0.502	0.500
	llama3	0.512	**0.525**	0.519	0.515
	mistral	0.500	**0.542**	0.490	0.501
	gemma2	**0.545**	0.444	0.516	0.535
	qwen2	0.523	0.525	**0.542**	0.537

The results show that the FSL strategy generally performs better on synthetic data, achieving the highest accuracy in five of the seven models tested. In particular, the fine-tuned model shows the best performance of all strategies on the synthetic test data. On the seed dataset, our fine-tuned model also shows superior accuracy compared to the others. Overall, both the FSL and ZSL methods are effective on this dataset, with FSL outperforming three models and ZSL outperforming four of the seven models tested.

4.3. Evaluation of the models' performance

Having identified the best-performing prompt for each model, we performed a comparative analysis of the key performance metrics. The results presented in Table 2 show that our fine-tuned model outperforms all metrics for the synthetic test data and three out of six metrics for the seed data. This shows that training LLMs with synthetic data can be highly effective on seed data, highlighting the potential of synthetic data for language detection tasks, even in complex contexts such as non-inclusive language. A notable strength of almost all models is recall, which measures how effectively the model identifies true positives - in this case, inclusive labels. In addition, Table 2 shows that *GPT-4o-mini* performs well in terms of specificity and precision on seed data, demonstrating the potential of this latest compact model from OpenAI for language detection tasks.

Table 2
Evaluation of LLMs using their best-performing prompts across Synthetic and Seed datasets.

Dataset	Model	Recall	Specificity	Accuracy	bACC	Precision	F1-score
Synthetic	phi3_finetuned_zslcot#0	**0.990**	**0.992**	**0.991**	**0.991**	**0.997**	**0.993**
	gpt_40_mini_fsl#0	0.797	0.979	0.851	0.888	**0.989**	0.883
	phi3_fsl#0	**0.972**	0.045	0.696	0.508	0.707	0.818
	llama3_fsl#0	**0.950**	0.177	0.720	0.563	0.732	0.827
	mistral_zslcot#0	**0.777**	0.262	0.626	0.520	0.718	0.746
	gemma2_fsl#0	0.526	0.671	0.569	0.598	**0.791**	0.632
	qwen2_zsl#1	**0.930**	0.230	0.722	0.580	0.741	0.824
Seed Data	phi3_finetuned_zslcot#0	0.824	0.581	**0.713**	**0.702**	0.700	**0.757**
	gpt_40_mini_fsl#0	0.549	**0.744**	0.638	0.647	**0.718**	0.622
	phi3_fsl#0	**1.000**	0.023	0.553	0.512	0.548	0.708
	llama3_fsl#0	**0.980**	0.070	0.564	0.525	0.556	0.709
	mistral_zslcot#0	**0.900**	0.184	0.591	0.542	0.592	0.714
	gemma2_fsl#0	**0.765**	0.326	0.564	0.545	0.574	0.655
	qwen2_zsl#1	**0.922**	0.163	0.574	0.542	0.566	0.701

5. Conclusion

In this paper, we propose a comprehensive pipeline for generating and evaluating synthetic data using LLMs. Our approach involves extracting sentences from seed data and systematically replacing job titles or adjectives with alternatives that have different grammatical endings while adhering to Italian language rules. The resulting synthetic dataset was used to fine-tune *Phi3*, a model from Microsoft. We evaluated the performance of this fine-tuned model against six other pre-trained models. The results demonstrate that the LLM fine-tuned on our synthetic data outperformed the others, achieving superior performance on both the synthetic and seed test datasets. These findings indicate that our approach has the potential to produce reliable synthetic data, a contribution that is particularly significant in the context of Trustworthy AI. The quality and validity of training data are foundational to ensuring fairness, transparency, and accountability in downstream applications. Future work will therefore extend this methodology beyond conventional performance metrics to include systematic evaluations of bias and fairness. In addition, we aim to fine-tune a broader set of LLMs, thereby strengthening the generalizability and ethical robustness of our approach.

Acknowledgments

The work reported in this paper has been partly funded by the European Union - NextGenerationEU, under the National Recovery and Resilience Plan (NRRP) Mission 4 Component 2 Investment Line 1.5: Strengthening of research structures and creation of R&D “innovation ecosystems”, set up of “territorial leaders in R&D”, within the project “MUSA - Multilayered Urban Sustainability Action” (contract no. ECS 00000037).

Declaration on Generative AI

During the preparation of this work, the authors used GPT-5 in order to: Grammar and spelling check. After using these services, the authors reviewed and edited the content as needed and take full responsibility for the publication’s content.

References

[1] R. Liu, J. Wei, F. Liu, C. Si, Y. Zhang, J. Rao, S. Zheng, D. Peng, D. Yang, D. Zhou, et al., Best practices and lessons learned on synthetic data, in: First Conference on Language Modeling, 2024.

[2] A. Founta, C. Djouvas, D. Chatzakou, I. Leontiadis, J. Blackburn, G. Stringhini, A. Vakali, M. Sirivianos, N. Kourtellis, Large scale crowdsourcing and characterization of twitter abusive behavior, in: Proceedings of the international AAAI conference on web and social media, volume 12, 2018.

[3] M. J. Riedl, G. M. Masullo, K. N. Whipple, The downsides of digital labor: Exploring the toll incivility takes on online comment moderators, Computers in Human Behavior 107 (2020) 106262.

[4] A. Prakash, S. Boochoon, M. Brophy, D. Acuna, E. Cameracci, G. State, O. Shapira, S. Birchfield, Structured domain randomization: Bridging the reality gap by context-aware synthetic data, in: 2019 International Conference on Robotics and Automation (ICRA), IEEE, 2019, pp. 7249–7255.

[5] H. Le, M. Nguyen, W. Q. Yan, Machine learning with synthetic data–a new way to learn and classify the pictorial augmented reality markers in real-time, in: 2020 35th International Conference on Image and Vision Computing New Zealand (IVCNZ), IEEE, 2020, pp. 1–6.

[6] S. Maghool, E. Casiraghi, P. Ceravolo, Enhancing fairness and accuracy in machine learning through similarity networks, in: International conference on cooperative information systems, Springer, 2023, pp. 3–20.

[7] Q. Miao, J. Yuan, S. Zhang, F. Wu, K. Kuang, Domaindiff: Boost out-of-distribution generalization with synthetic data, in: ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2024, pp. 5640–5644.

[8] V. Bellandi, P. Ceravolo, E. Damiani, S. Maghool, M. Cesari, I. Basdekis, E. Iliadou, M. D. Marzan, A methodology to engineering continuous monitoring of intrinsic capacity for elderly people, Complex & Intelligent Systems 8 (2022) 3953–3971.

[9] S. Maghool, N. Maleki-Jirsaraei, Epidemic spreading phenomena on a scale-free network with time-varying transmission rate due to social responses, International Journal of Modern Physics C 31 (2020) 2050148.

[10] P. Eigenschink, T. Reutterer, S. Vamosi, R. Vamosi, C. Sun, K. Kalcher, Deep generative models for synthetic data: A survey, IEEE Access 11 (2023) 47304–47320.

[11] T. Sen, A. Das, M. Sen, Hatetinyllm: Hate speech detection using tiny large language models, arXiv preprint arXiv:2405.01577 (2024).

[12] K. Guo, A. Hu, J. Mu, Z. Shi, Z. Zhao, N. Vishwamitra, H. Hu, An investigation of large language models for real-world hate speech detection, in: 2023 International Conference on Machine Learning and Applications (ICMLA), IEEE, 2023, pp. 1568–1573.

[13] Y. Lu, M. Shen, H. Wang, X. Wang, C. van Rechem, T. Fu, W. Wei, Machine learning for synthetic data generation: a review, arXiv preprint arXiv:2302.04062 (2023).

[14] M. Goyal, Q. H. Mahmoud, A systematic review of synthetic data generation techniques using generative ai, Electronics 13 (2024) 3509.

[15] Z. Li, H. Zhu, Z. Lu, M. Yin, Synthetic data generation with large language models for text classification: Potential and limitations, arXiv preprint arXiv:2310.07849 (2023).

[16] C. Casula, E. Leonardelli, S. Tonelli, Don't augment, rewrite? assessing abusive language detection with synthetic data, in: Findings of the Association for Computational Linguistics ACL 2024, 2024, pp. 11240–11247.

[17] G. Maheshwari, D. Ivanov, K. E. Haddad, Efficacy of synthetic data as a benchmark, arXiv preprint arXiv:2409.11968 (2024).

[18] A. Khullar, D. Nkemelu, V. C. Nguyen, M. L. Best, Hate speech detection in limited data contexts using synthetic data generation, ACM Journal on Computing and Sustainable Societies 2 (2024) 1–18.

[19] A. Magron, A. Dai, M. Zhang, S. Montariol, A. Bosselut, Jobskape: A framework for generating synthetic job postings to enhance skill matching, arXiv preprint arXiv:2402.03242 (2024).

[20] H. Li, Q. Dong, Z. Tang, C. Wang, X. Zhang, H. Huang, S. Huang, X. Huang, Z. Huang, D. Zhang, et al., Synthetic data (almost) from scratch: Generalized instruction tuning for language models, arXiv preprint arXiv:2402.13064 (2024).

[21] A. Zhezherau, A. Yanockin, Hybrid training approaches for llms: Leveraging real and synthetic data to enhance model performance in domain-specific applications, arXiv preprint arXiv:2410.09168 (2024).

[22] J.-T. Baillargeon, L. Lamontagne, Assessing the impact of sequence length learning on classification tasks for transformer encoder models, in: The International FLAIRS Conference Proceedings, volume 37, 2024.

[23] L. Beurer-Kellner, M. Fischer, M. Vechev, Prompting is programming: A query language for large language models, Proceedings of the ACM on Programming Languages 7 (2023) 1946–1969.

[24] L. Reynolds, K. McDonell, Prompt programming for large language models: Beyond the few-shot paradigm, in: Extended abstracts of the 2021 CHI conference on human factors in computing systems, 2021, pp. 1–7.

[25] G.-G. Lee, E. Latif, X. Wu, N. Liu, X. Zhai, Applying large language models and chain-of-thought for automatic scoring, Computers and Education: Artificial Intelligence 6 (2024) 100213.

[26] M. Labonne, Fine-tune llama 3.1 ultra-efficiently with unsloth, https://huggingface.co/blog/mlabonne/sft-llama3, 2024. Online; accessed 2024-12-06.

[27] Y. Li, Y. Yu, C. Liang, P. He, N. Karampatziakis, W. Chen, T. Zhao, Loftq: Lora-fine-tuning-aware quantization for large language models, arXiv preprint arXiv:2310.08659 (2023).

[28] S. Jain, Hugging face. in introduction to transformers for nlp: With the hugging face library and models to solve problems, 2022.

[29] K. H. Brodersen, C. S. Ong, K. E. Stephan, J. M. Buhmann, The balanced accuracy and its posterior

distribution, in: 2010 20th international conference on pattern recognition, IEEE, 2010, pp. 3121–3124.