DIAG-Sapienza at GSI:detect: Joint Detection and Classification of Gender Stereotypes with Structured Prompting and Fine-Tuning

Olga E. Sorokoletova[1], Emanuele Musumeci[1] and Daniele Nardi[1]

[1]*Sapienza University of Rome, Piazzale Aldo Moro, 5, Roma, 00185, Italy*

Abstract

This report presents the contribution from the DIAG-Sapienza team for the GSI:detect task at EVALITA 2026. We address both the main regression task and the classification sub-task in a joint formulation, motivated by the hypothesis that category-level information can support more accurate estimation of gender stereotype intensity, especially in low-data settings. Our contribution features three systems corresponding to the challenge tracks: 1) a zero-shot approach that used GPT-5 as an LLM-as-a-Judge with structured prompting and explanation generation to elicit implicit reasoning, 2) a few-shot approach that augments prompting with randomly sampled and retrieved in-context examples, and 3) an encoder-only fine-tuned RoBERTa-based model that integrates LLM-generated reasoning sentences as additional input. Leaderboard results on the official test set show that our zero-shot and few-shot systems rank first and second, on the main regression task, outperforming all competing submissions. Overall, the results suggest that jointly predicting gender stereotype values and categories benefits the regression task, and that eliciting concise explanations can improve prediction quality.

Keywords

Gender Stereotype Detection, Bias Detection, Large Language Models, LLM-as-a-Judge, Zero-Shot Learning, Few-Shot Learning

1. Introduction

In this report, we address the GSI:detect challenge [1] of the EVALITA 2026 campaign [2], which focuses on evaluating systems' ability to detect and classify gender stereotypes (GSs) across diverse types of short texts in Italian. The dataset includes manually collected texts drawn from social media and informative websites, spanning both formal and informal language, which makes the task representative of real-world usage. The challenge comprises a compulsory main task on *Gender Stereotype Detection* and an optional sub-task on *Gender Stereotype Classification.* The main task is formulated as a regression problem, requiring the assignment of a numerical score in the range $[0, 1]$ that quantifies the degree to which a text contains or refers to a gender stereotype. The sub-task is a multi-class classification problem, in which the system must assign one of six predefined Gender Stereotype categories: Role, Personality, Competence, Physical, Sexual, and Relational, when a stereotype is present, or None when no stereotype is detected.

In our approach, we address both tasks jointly rather than treating them as independent challenges. While a modular formulation is possible, we generate both outputs within the same system call from the same input. This design choice is motivated by observations in the literature that Large Language Models (LLMs) often struggle with purely numerical outputs and tend to perform less reliably on regression tasks than on text-based tasks [3]. We therefore hypothesize that explicitly incorporating information from the classification task into the context provided to the LLM, including the definitions of stereotype categories, can support the estimation of gender stereotype values and improve regression performance in zero-shot and few-shot settings.

This issue does not arise in fine-tuning scenarios, where the model is explicitly adapted to the target

EVALITA 2026: 9th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian, Feb 26 – 27, Bari, IT

sorokoletova@diag.uniroma1.it (O. E. Sorokoletova); musumeci@diag.uniroma1.it (E. Musumeci); nardi@diag.uniroma1.it (D. Nardi)

0009-0005-4356-2649 (O. E. Sorokoletova); 0009-0004-2359-5032 (E. Musumeci); 0000-0001-6606-200X (D. Nardi)

© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

objectives rather than being prompted to perform them as a by-product of next-word prediction, as in general-purpose LLMs. However, fine-tuned models are highly sensitive to architecture choices. In this setting, it is straightforward to implement a parallelized architectures that jointly handle regression and classification while sharing intermediate representations, for example, a shared encoder with task-specific heads. This design enables a controlled assessment of whether joint modeling yields measurable gains, and it allows us to test the extent to which shared representations can transfer information between the two tasks under supervised training.

These modeling choices must ultimately be grounded in a clear operationalization of what constitutes a stereotype in the first place. Within the GSI:detect challenge, a stereotype is defined as a pre-constituted, generalized, and simplistic opinion that is not based on the evaluation of individual cases but is mechanically repeated about people, events, or situations [4]. Gender stereotypes, in particular, are frequently found in misogynistic hate speech but also appear in non-hateful communication, where they may be unconscious and used with positive meaning. The greatest challenge lies in detecting such stereotypes when they are latent in both the form and the implied content of the analyzed text.

For this reason, in our zero-shot and few-shot experiments, we explicitly feature this definition in the task explanation provided to the LLM. This step is crucial, as stereotypes and bias are inherently ambiguous and subjective concepts. As reflected in the dataset annotation scheme, even human annotators may disagree on the presence and intensity of a gender stereotype: depending on personal background, values, and experiences, the same expression may be seen as clearly stereotypical, neutral, or even positive [2, 5]. LLMs, in turn, are highly sensitive to prompting strategies, a property that is also exploited in adversarial prompting scenarios [6]. Without a clear operational definition, an LLM acting as a generalizer over its training data may therefore struggle to align with the intended task interpretation. By explicitly grounding the model in the task-specific definition, we steer it toward the intended linguistic and conceptual framing of gender stereotypes, promoting more consistent and accurate judgments.

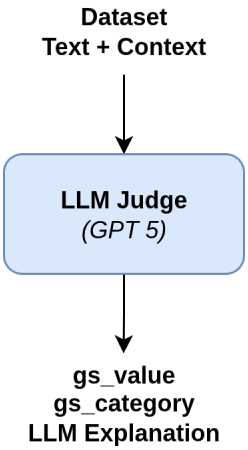
To address the challenge comprehensively, we develop three systems corresponding to the three challenge tracks: a zero-shot system, a few-shot system, and an encoder-only fine-tuned system. All three systems are applied to the jointly treated problem of Gender Stereotype Detection and Classification. Section 2 describes the architecture and design choices of each proposed system. Section 3 analyzes the experimental results, first reporting the leaderboard outcomes on the test set based on our submissions and then presenting the internal experimentation conducted to obtain those submissions. Finally, Section 4 concludes the report by summarizing our observations and discussing implications and potential directions for improvement.

2. Description of the System

This section describes the systems developed for the GSI:detect challenge and the design choices underlying their implementation. In particular, we detail three systems corresponding to the zero-shot, few-shot, and encoder-only fine-tuned settings, highlighting their shared components as well as the differences introduced by prompting strategies, in-context examples, retrieval mechanisms, and supervised training. The goal of this section is to provide a clear and systematic account of how each system operationalizes the task definitions and how these design decisions support the experimental analysis presented in the subsequent section.

2.1. Zero-Shot System

Zero-shot learning [7] is typically defined as the ability to generalize to previously unseen classes by transferring knowledge from seen classes, often through auxiliary semantic information. In the context of prompting Large Language Models, the notion of *zero-shot* is used in a more operational sense: the model is queried without task-specific in-context examples of solved instances (shots), which would instead constitute *few-shot* prompting [8], and without any parameter updates, and is expected to perform the task based solely on natural-language instructions. Under this constraint, the



System Prompt: You are an impartial judge for detecting and classifying gender stereotypes in short Italian texts. Given a text:
- Assign a GS score in $[0, 1]$ indicating how strongly it reflects or refers to a gender stereotype;
- Round the score to the closest among: 0.00, 0.25, 0.50, 0.75, and 1.00;
- Assign one stereotype category from: *Role*, *Personality*, *Competence*, *Physical*, *Sexual*, *Relational*;
- If multiple categories appear, select the one that appears first in the text.

Definitions:
A stereotype is a generalized and simplistic opinion, mechanically repeated and not based on individual evaluation.

- *Role*: expectations about what men/women should do or be;
- *Personality*: emotional or behavioral traits assigned by gender;
- *Competence*: judgments about abilities or skills based on gender;
- *Physical*: expectations about appearance or self-care;
- *Sexual*: stereotypes or insults related to sexuality;
- *Relational*: expectations about behavior in interpersonal or romantic relations.

Provide a one-sentence explanation in English for your score and category.

Assistant Prompt: `# repeated_definition_of_a_stereotype`. Gender stereotypes, in particular, are often found in misogynistic hate speech, but they also appear in non-hateful communication; unconscious stereotypes can be used with positive meaning as well. `# repeated_definitions_of_categories`.

User Prompt: You are given a short Italian text (and context when available). Produce a single prediction with these fields:

- `"gs_value"`: one of 0.00, 0.25, 0.50, 0.75, or 1.00
- `"gs_category"`: one of *Role*, *Personality*, *Competence*, *Physical*, *Sexual*, *Relational* (or *no* if score is 0.00)
- `"explanation"`: one concise sentence in English

Output exactly one JSON object and nothing else.

Text:
`<text_wo_context>`.
`# if_contextual_information_is_provided:`
Context (use only to interpret the text, not to replace it):
`<context>`.
Now provide your prediction in strict JSON format (one JSON object only).

Figure 1: Architecture of the zero-shot system and prompt template with dynamically replaced placeholders.

prompt becomes the main mechanism for steering the model toward the intended behavior. As a result, performance can vary substantially with the formulation of the instructions, the degree of specificity, and the linguistic patterns used to express task definitions and output constraints.

The architecture of our zero-shot system is linear, as illustrated in Figure 1. In both zero-shot and few-shot experiments, we employ GPT-5 as an LLM-as-a-Judge to evaluate the input texts, leveraging

its strong instruction-following and reasoning capabilities as the most recent model in the GPT family. The system is controlled through a three-level prompting scheme consisting of a system prompt, an assistant prompt, and a user prompt.

The system prompt assigns the role of an impartial judge, specifies the task requirements, and introduces the stereotype categories. The assistant prompt then provides the operational definition of gender stereotypes and reiterates the category definitions to reinforce the shared task framing (reiterations are omitted in Figure 1). Finally, the user prompt supplies the instance-level input and constraints the expected output format, requiring the fields `gs_value`, `gs_category`, and `explanation`, defined as "a one-sentence explanation in English" and denoted as "LLM Explanation" in the scheme in Figure 1. It provides the input text, separated from an optional description of its surrounding context when available. The `context` field is used exclusively to separate contextual information from the text: the model is first presented with the text alone and is then explicitly instructed to interpret it within the given context, using the formulation: "Context (use only to interpret the text, not to replace it)". A lightweight preprocessing step is applied beforehand to separate the text from its context.

Requiring the model to generate a one-sentence explanation is intended to encourage more accurate responses by implicitly eliciting reasoning during inference [9]. Since explanations are not part of the official submission format, this field is discarded after generation.[1] As the two tasks are treated jointly, both `gs_value` and `gs_category` are predicted within a single model call.

2.2. Few-Shot System

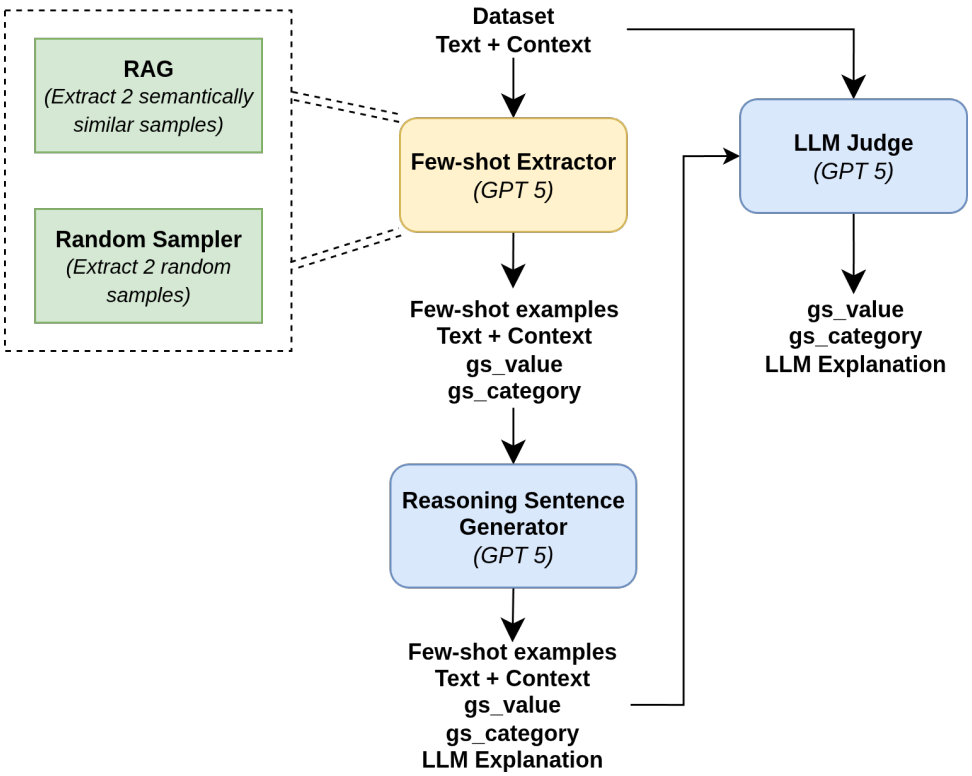


Figure 2: Architecture of the few-shot system.

In-context learning can often yield surprisingly strong results, motivating us to explore a few-shot variant of our approach, shown in Figure 2.

In this setting, requesting an `explanation` in the user prompt while providing few-shot examples that omit the `explanation` field would create a mismatch between the demonstrated output template

[1]Code and supplementary materials, including the model-generated explanations, are available at: GSI-detect_DIAG_Sapienza.

and the required output format. Since in-context learning relies on pattern induction from the provided examples [8], such structural inconsistencies can reduce the model's ability to reliably follow instructions and reproduce the intended schema. To maintain alignment between demonstrations and the target output format, we therefore ensure that they contain the same fields.

For experiments that include explanations, we adopt a two-step procedure. First, in a separate GPT-5 call, we generate one-sentence explanations for the ground-truth examples by providing the `text`, `context`, and the gold `gs_value` and `gs_category`, and prompting the model to justify these gold labels. We then augment the few-shot demonstration JSON objects with the resulting `explanation` field before using them for few-shot prompting. This setup uses two GPT-5 calls: one to generate explanations for demonstrations and one to act as a judge at inference time, as in the zero-shot setting.

To support a more informative selection of few-shot examples, we implement a Retrieval-Augmented Generation (RAG) that selects part of the examples based on textual similarity to the input instance and complements them with additional examples sampled at random. This design exposes the model to both similar and diverse patterns and has been shown to improve few-shot performance [10]. We use LanceDB[2] to build the retrieval database. In LanceDB, we configure an embedding function from the registry and include it in the table schema. The embedding function automatically generates embeddings when data is inserted into the table and similarly embeds the queried text at retrieval time. We then retrieve the most similar examples by comparing the query embedding to the stored text embeddings in the database. LanceDB supports the most popular embedding providers. For `sentence-transformers`, the default model is `all-MiniLM-L6-v2`.

2.3. Encoder-Only Fine-Tuned System

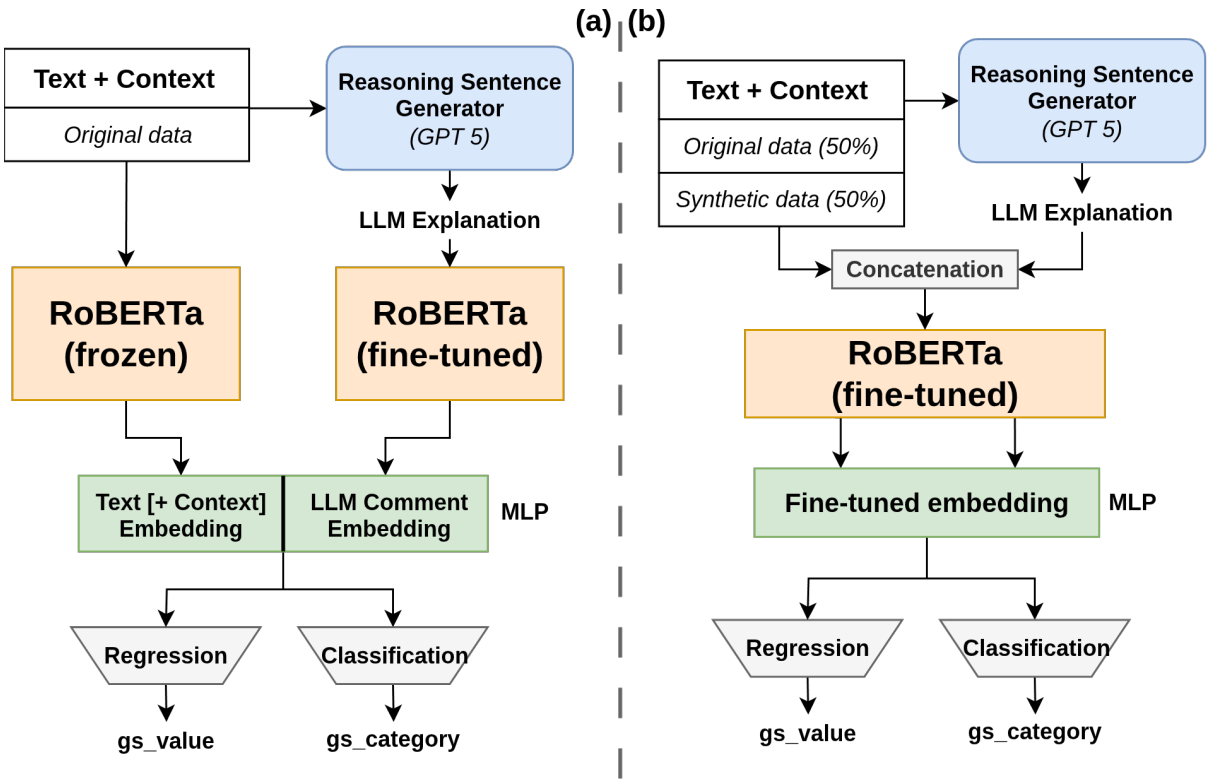


Figure 3: Architecture of the encoder-only fine-tuned system. Panel (a) illustrates a preliminary dual encoder configuration evaluated during ablation studies (Section 3.2.2) that yielded suboptimal performance. Panel (b) shows the final submitted single encoder architecture adopted for the encoder-only track.

[2]https://lancedb.com/

Before the advent of the LLM-as-a-Judge paradigm, the dominant approach for evaluating language systems relied on supervised fine-tuning over task-specific datasets [11]. Nowadays, few-shot or zero-shot inference can yield competitive performance, while in other cases, fine-tuning remains more effective for eliciting specialized behaviors from pretrained models. The relative effectiveness of zero-shot prompting, few-shot in-context learning, and supervised fine-tuning is highly task-dependent and remains an open question, particularly for nuanced phenomena such as gender bias and stereotype detection. For this reason, in addition to zero-shot and few-shot approaches, we also explore the third submission track of the GSI:detect challenge and develop a fine-tuned system.

The architecture of our fine-tuned system belongs to the class of encoder-only models, which focus on learning discriminative representations. The system, illustrated in Figure 3, takes as input the original Italian texts from the dataset, with optional `context` concatenated to the textual sample when available. In parallel, reasoning sentences corresponding to the `explanation` field are generated through a separate call to a GPT-5-mini model. In this preliminary step, the LLM is prompted to assess the presence of gender stereotypes and to perform the joint regression and classification task. As a result, the overall architecture comprises two components: a GPT-5-mini-based Reasoning Sentence Generator and a RoBERTa-based judge model that is fine-tuned to perform the final predictions.

2.3.1. RoBERTa-Based Judge

The RoBERTa-based judge [12] employs a single encoder that takes as input the concatenation of the `text`, the (optional) `context`, and the LLM-generated `explanation`. Fine-tuning is performed directly on this encoder, producing a feature representation in the model's embedding space. The architecture is completed by two task-specific heads, one for regression and one for classification, which are trained jointly using a weighted loss function to balance the two objectives.

2.3.2. Data Augmentation

Before fine-tuning, data augmentation is applied to mitigate overfitting, which can arise due to the limited size of the original dataset. Additional samples are generated through a separate GPT-5 call, in which the model is tasked with producing new Italian texts that express the same gender stereotype category and value as the original samples. For each instance in the original dataset, a new `text` is generated with the same `gs_value` and `gs_category` but with different wording, structure, and contextual framing.

The augmentation process mirrors the prompting strategy used in the zero-shot system and relies on a system prompt, an assistant prompt, and a user prompt. In the system prompt, GPT-5 is assigned the role of an expert linguist specializing in the detection and classification of gender stereotypes in Italian and is instructed to generate realistic and grammatically correct texts. The prompt specifies several constraints: the generated texts must preserve the stereotype intensity level, expressed as `gs_value` (`0.00=no stereotype, 0.25=mild, 0.50=moderate, 0.75=strong, and 1.00=explicit/severe`), maintain the original `gs_category`, and remain natural and plausible while differing substantially from the source text. The assistant prompt reiterates the definitions of gender stereotypes and their categories to reinforce adherence to these constraints. Finally, the user prompt provides the original `text`, `gs_value`, and `gs_category` and requests the generation of new text matching these attributes. The full prompt template is reported in Appendix A.1.

To prevent the inclusion of incoherent outputs, hallucinations, or violations of the imposed constraints, all generated samples undergo manual curation and filtering. As a result, 195 data samples are added to the development dataset, substantially increasing its diversity. An example of an original development sample and its generated counterpart is shown below; the English translation is provided in Appendix A.2.

Original GSI Dev Sample (ID: GS0458)

`text_wo_context`: "IO, mammta e tu, non ne posso più ecc ecc ecc. Dobbiamo anche dire, che i matrimoni duravano una vita e non si sentivano tutti i giorni come oggi di femminicidi, si scherza MOLTO sui sentimenti delle persone, perche' la torta viene mangiata prima della festa e datosi che non c'è più vergogna di NIENTE, molto facilmente si dice che non era buona e non c'è più desiderio, ma sé si mangia un poco alla volta, la torta è desiderata e buona, poche parole a buon intenditore!"
`context`: "Commento a una vecchia foto di un matrimonio in cui sia gli sposi, sia gli invitati sono tutti molto seri, con la dicitura: *Un matrimonio d'altri tempi.*"
`gs_value`: 0.25
`gs_category`: sexual

Augmented Counterpart (Synthetic Sample)

`text`: "Quando ascolto mia nonna parlare dei matrimoni di una volta, ripete che le ragazze perbene sapevano aspettare e non si concedevano subito; forse è anche per questo che le storie duravano."
`gs_value`: 0.25
`gs_category`: sexual

3. Results

In this section, we report both the leaderboard results obtained with the official GSI:detect evaluator on the test set and the findings from our internal experimentation conducted before submission.

3.1. Leaderboard Results

Our submission consists of six evaluation runs on the official test set: one run for the zero-shot track, one for the few-shot track, and four runs for the encoder-only fine-tuning track. For the encoder-only fine-tuned system, the four runs correspond to two different experimental configurations. Specifically, Runs 1 and 2 use only the original dataset texts and generated reasoning sentences, whereas Runs 3 and 4 use the augmented version of the dataset. Each configuration includes two runs because we consider two alternative formats for the regression output. In Run 1 (and analogously Run 4), the model predicts a continuous `gs_value` in the full range $[0, 1]$. In Run 2 (and analogously Run 3), the predicted `gs_value` is rounded to the nearest value in the discrete set $\{0.00, 0.25, 0.50, 0.75, 1.00\}$, reflecting the granularity of the ground-truth annotation scheme.

Table 1 and Table 2 summarize the official leaderboard results for the main task and the sub-task, respectively. We first discuss the main task of *Gender Stereotype Detection*. Here, our zero-shot and few-shot systems achieve the first and second positions on the leaderboard, respectively, outperforming all other submitted systems. The official evaluated metric for this task is: $1 + \frac{1}{NMSE}$, where $NMSE$ denotes the Normalized Mean Square Error, with higher values indicating better performance. In contrast, the encoder-only fine-tuned system yields substantially lower scores, suggesting that further refinement of this approach is required.

Table 2 reports the results for the sub-task of *Gender Stereotype Classification*. In this setting, the few-shot system achieves the highest ranking among our submissions, placing 14th overall. The primary evaluation metric is the Micro-averaged F1 score, with higher values corresponding to better performance. Compared to the results on the main task, these findings support our hypothesis that the joint treatment of detection and classification is particularly beneficial for the regression task, while further improvements are necessary to strengthen performance on the classification sub-task.

Table 1
Main Task (Regression) results achieved on the test dataset.

Track	Run	Ranking	1+1/NMSE	NMSE	MSE
zero-shot	best	1	0.700	0.4292	0.0766
few-shot	best	2	0.685	0.4594	0.0820
encoder-only	1	45	0.470	1.1260	0.3752
encoder-only	2	46	0.462	1.1644	0.3691
encoder-only	4	47	0.453	1.2072	0.3562
encoder-only	3	48	0.446	1.2409	0.3601

Table 2
Sub-Task (Classification) results achieved on the test dataset.

Track	Run	Ranking	F1 Micro	F1 Macro
few-shot	best	14	0.606	0.55
zero-shot	best	19	0.581	0.52
encoder-only	1	32	0.375	0.29
encoder-only	2	33	0.357	0.29
encoder-only	3	34	0.349	0.29
encoder-only	4	35	0.349	0.29

Table 3
Ablation study of the few-shot system from Figure 2. Evaluation results achieved during experimentation on the development dataset.

System	Architecture	MSE	Accuracy	F1 Micro
few-shot	full	0.0765	0.5243	**0.6699**
few-shot	w/o reasoning sentence	0.0781	0.56	0.6250
few-shot	w/o RAG database	**0.070**	**0.600**	0.6350

3.2. Internal Experimentation

Our internal experimentation is structured into two ablation studies, focusing respectively on the few-shot system and the encoder-only fine-tuned system.

3.2.1. Ablation Study of the Few-Shot System

In few-shot experiments, we conduct two ablation studies: 1) with vs. without reasoning sentences and 2) with vs. without the RAG pipeline. In the first ablation, we disable the Reasoning Sentence Generator in Figure 2: we neither generate reasoning sentences for the few-shot examples nor request them in the user prompt. In the second ablation, we remove retrieval and sample all four few-shot examples uniformly at random from the development set. To avoid overlap between examples and the query instance, we apply a partitioning of the development data at each iteration and sample demonstrations only from the partition that excludes the query text.

Table 3 presents evaluation results on the development dataset, where gold labels are available. Since our approach treats the two tasks jointly, we report both Mean Squared Error (MSE) for the regression task (non-normalized; lower is better) and F1 Micro for the classification task. For the regression task, we additionally calculate accuracy.

Based on these results, we select the configuration without RAG to produce the test-set submission, denoted as the best run in Table 1 and Table 2. This configuration yields stronger performance on the regression task, whereas the full architecture achieves the best classification performance. However, the retrieval of RAG examples takes time. Finally, across settings, including reasoning sentences consistently improves performance, even though the gains are modest.

3.2.2. Ablation Study of the Encoder-Only Fine-Tuned System

Model selection was performed by training the model on 70% of the original dataset and evaluating predictions on the remaining 30%.

To address the limited size of the training data and mitigate the risk of model collapse, we adopted an encoder-only architecture with a single-encoder concatenation. In this design, both the `text` and the LLM-generated `explanation` are processed by a single encoder instance, resulting in approximately $125M$ trainable parameters and reducing overfitting risk. The model features a shared Multi-Layer Perceptron (MLP) with a hidden dimension of 768, which feeds into a classification head with hidden dimension 768 and a regression head with hidden dimension 512.

We employed a robust training strategy to stabilize learning on the small training dataset (322 samples). The dataset exhibited a strong class imbalance: to mitigate this problem, we used inverse-frequency class weights in the loss function. Training stability was further improved through gradient clipping (`max_norm=1.0`) and label smoothing with a factor of 0.1. A dropout rate of 0.3 was applied throughout the network: after each ReLU activation in the shared MLP and both before the hidden layer and before the output layer of each task-specific head. The classification head was initialized using Xavier uniform weights (gain 0.1) to ensure a neutral starting point. Optimization was performed with AdamW using a learning rate of $2e^{-5}$, a batch size of 4, and weight decay of 0.01. Training was run for up to 150 epochs with early stopping (patience 15).

Our experimentation followed an iterative ablation process, starting from a simple baseline and progressively increasing the architecture complexity until reaching the most promising configuration. The evolution of this process is detailed below and summarized in Table 4.

Baseline: Single Encoder We initially established a baseline using a single encoder architecture, in which the input text and the corresponding LLM-generated reasoning sentence are concatenated and processed by a RoBERTa-based encoder ($\sim 125M$ parameters). We applied a dropout rate of 0.5 to reduce overfitting over the training set. Due to the simplicity of the task, training was performed for up to 50 epochs, without data augmentation. Despite avoiding complete collapse, the model suffered from slow convergence and suboptimal performance, achieving a best validation Micro F1 score of ≈ 0.13 on the classification sub-task. This result indicates that the architecture struggled to effectively exploit the additional contextual information provided by the reasoning sentences.

Dual Encoder To improve upon the baseline, we hypothesized that a dual encoder architecture could better capture the distinct features of the original texts and the explanations. This setup employed two independent RoBERTa encoders ($\sim 250M$ parameters in total), whose outputs were concatenated and processed by a shared MLP. The first encoder, receiving the `text` and `context`, was kept frozen, while the second encoder, processing the reasoning sentence, was fine-tuned for up to 50 epochs. A dropout rate of 0.5 was applied. However, this configuration resulted in model collapse: the significant increase in parameters (+100%), combined with the low sample count of the training data, led to immediate overfitting. The model consistently collapsed to predicting a single class and achieving a low best F1 score of ≈ 0.04), regardless of hyperparameter adjustments. Supposedly, the reason behind the model collapse was the excessive model complexity with respect to the available dataset size.

Optimized Single Encoder (Final Model) The final delivered architecture reverts to the single encoder architecture and focuses on optimization and regularization to maximize stability and performance. Compared to the baseline, dropout was reduced, gradient clipping and label smoothing were introduced. Crucially, the training set was augmented in two steps with synthetic data, generated as previously described. A first experiment was conducted with a training set with a 20% of synthetic samples. A second experiment was conducted with 50% synthetic samples. Training was extended to 150 epochs with early stopping. This configuration yielded the best results, achieving high stability and improved performance, with no signs of model collapse. No further experimentation was conducted.

Table 4
Ablation study of model architectures and configurations. The dual encoder scaling attempt failed due to data scarcity, while the optimized single encoder with synthetic data augmentation achieved the best performance.

Experiment	Architecture	Params	Augmentation	Dropout	Epochs	Best Val F1
Baseline	Single Enc.	~125M	None	0.5	50	0.1257
Scaling Attempt	**Dual Enc.**	**~250M**	**None**	**0.5**	**50**	**0.0408**
Optimized	Single Enc.	~125M	Synthetic (20%)	0.3	100	0.8293
Final	**Single Enc.**	**~125M**	**Synthetic (50%)**	**0.3**	**150**	**0.8834**

In our experimentation, we adopted a set of focused design choices aimed at validating a coherent modeling hypothesis within the constraints of the shared task setting. While these choices proved effective, they also naturally delimit the scope of the experimental analysis and suggest several directions for further investigation.

4. Discussion and Conclusions

In our experimentation, we adopted a set of focused design choices. While these choices proved mostly effective, they naturally suggest several directions for further investigation.

First, we exclusively considered a joint formulation of *Gender Stereotype Detection* and *Gender Stereotype Classification*. Although treating the two tasks independently could offer additional insights, our results indicate that the joint approach is particularly effective for the regression task. A systematic comparison with fully independent task formulations would nonetheless be valuable to quantify the performance margin introduced by joint modeling and to further characterize task interactions.

Our systems rely on GPT-5 as the core evaluator model. While this choice is justified by its strong reasoning capabilities and overall performance, we did not conduct a comparative evaluation across different LLM families or providers. A broader cross-model analysis could help determine whether similar gains can be achieved with alternative architectures or whether the observed performance is tied to specific properties of GPT-5.

Another deliberate design choice concerns prompt structuring. We extended the standard system-prompt configuration by introducing an assistant prompt that reiterates key definitions. This is motivated by the hypothesis that reinforcing background knowledge can improve model alignment with the task definition. A controlled ablation removing the assistant prompt would allow for an assessment of its individual contribution.

Regarding the zero-shot system, we did not perform a dedicated ablation study to isolate the effect of enforcing explanation generation. Such an analysis could further clarify the role of implicit reasoning in improving prediction quality.

In the few-shot setting, we expected the integration of Retrieval-Augmented Generation to yield improvements, but did not observe it. A possible explanation is the limited semantic overlap between development and test samples, which reduces the benefit of retrieving highly similar examples. In addition, the remaining two examples are sampled randomly rather than selected to maximize contrast, and the small number of demonstrations limits coverage of all stereotype categories in the classification task. Exploring alternative retrieval strategies, varying the number of examples, or enforcing class-balanced selection could help address these factors. Although the few-shot architecture with RAG achieved stronger classification performance on the development set, we ultimately submitted the configuration without RAG due to its better regression performance and lower computational overhead. As a result, the potential leaderboard impact of the RAG-enhanced few-shot system on the classification task remains an open question.

Finally, it is noteworthy that zero-shot and few-shot systems outperform the fine-tuned model on the leaderboard. While this outcome may appear counterintuitive, fine-tuning performance is highly sensitive to architectural choices, backbone selection, and data availability. Our ablation study indicates

that encoder-only fine-tuning remains promising, but is strongly constrained by the limited size and imbalance of the training data. Further refinements, alternative architectures, and training on larger and more balanced datasets could substantially improve performance. Additionally, more systematic experimentation with different levels of data augmentation would help clarify its impact.

Overall, our zero-shot and few-shot systems demonstrate strong performance and robustness in low-data settings, validating the effectiveness of joint modeling and structured prompting. The encoder-only fine-tuned system represents an avenue for future improvement and extension.

5. Acknowledgments

This work has been carried out while Olga Sorokoletova and Emanuele Musumeci were enrolled in the Italian National Doctorate on Artificial Intelligence run by Sapienza University of Rome. We acknowledge partial financial support from PNRR MUR project PE0000013-FAIR.

Declaration on Generative AI

During the preparation of this work, the author(s) used GPT-5.2 and the Grammarly plugin in order to: Paraphrase and reword; Improve writing style; and Grammar and spelling check. After using these tool(s)/service(s), the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

References

[1] G. Comandini, M. Speranza, S. Brenna, D. Testa, S. Cavagnoli, B. Magnini, Gsi:detect at evalita 2026: Overview of the task on detecting gender stereotypes in italian, in: Proceedings of the Ninth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2026), CEUR.org, Bari, Italy, 2026.

[2] F. Cutugno, A. Miaschi, A. P. Aprosio, G. Rambelli, L. Siciliani, M. A. Stranisci, Evalita 2026: Overview of the 9th evaluation campaign of natural language processing and speech tools for italian, in: Proceedings of the Ninth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2026), CEUR.org, Bari, Italy, 2026.

[3] A. Testolin, Can neural networks do arithmetic? a survey on the elementary numerical skills of state-of-the-art deep learning models, Applied Sciences 14 (2024) 744.

[4] S. Cavagnoli, F. Dragotto, et al., Sessismo, Mondadori Education, 2021.

[5] M. Sanguinetti, F. Poletto, C. Bosco, V. Patti, M. A. Stranisci, An italian twitter corpus of hate speech against immigrants, in: Proceedings of the eleventh international conference on language resources and evaluation (LREC 2018), 2018.

[6] Y. Zeng, et al., How johnny can persuade llms to jailbreak them: Rethinking persuasion to challenge ai safety by humanizing llms, 2024. URL: https://arxiv.org/abs/2401.06373. arXiv:2401.06373.

[7] Y. Xian, B. Schiele, Z. Akata, Zero-shot learning-the good, the bad and the ugly, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 4582–4591.

[8] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al., Language models are few-shot learners, Advances in neural information processing systems 33 (2020) 1877–1901.

[9] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou, et al., Chain-of-thought prompting elicits reasoning in large language models, Advances in neural information processing systems 35 (2022) 24824–24837.

[10] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, et al., Retrieval-augmented generation for knowledge-intensive nlp tasks, Advances in neural information processing systems 33 (2020) 9459–9474.

[11] J. Howard, S. Ruder, Universal language model fine-tuning for text classification, arXiv preprint arXiv:1801.06146 (2018).

[12] 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, 2019. URL: https://arxiv.org/abs/1907.11692. arXiv:1907.11692.

A. Appendix

A.1. Prompt Template for Data Augmentation

Below, we report the prompt template adopted during the data augmentation procedure. Placeholders enclosed in angle brackets (e.g., `<text_wo_context>`) are dynamically replaced with instance-specific values.

System Prompt: You are an expert linguist specializing in detecting and classifying gender stereotypes in Italian texts. Your task is to augment a dataset by generating new realistic Italian texts that contain gender stereotypes with the SAME stereotype score and category as the original. The generated text must:
- Be natural, grammatically correct Italian;
- Preserve the stereotype intensity level (score) of the original;
- Use different wording and context while maintaining the stereotype category;
- Sound authentic and plausible.

Stereotype Definitions:

- *Role*: expectations about what men/women should do or be;
- *Personality*: emotional or behavioral traits assigned by gender;
- *Competence*: judgments about abilities or skills based on gender;
- *Physical*: expectations about appearance or self-care;
- *Sexual*: stereotypes or insults related to sexuality;
- *Relational*: expectations about behavior in interpersonal or romantic relations.

Score intensity:
0.00=no stereotype; 0.25=mild; 0.50=moderate; 0.75=strong; 1.00=explicit/severe.

Assistant Prompt: `# repeated_definition_of_a_stereotype`. Gender stereotypes, in particular, are often found in misogynistic hate speech, but they also appear in non-hateful communication; unconscious stereotypes can be used with positive meaning as well. `# repeated_definitions_of_categories`.

User Prompt: Generate a NEW Italian text with the SAME gender stereotype as the original.

IMPORTANT CONSTRAINTS:
- Create a completely different sentence (different wording, structure, scenario)
- Preserve the exact stereotype category and intensity score
- Output ONLY a valid JSON object with NO additional text

Original text:
`<text_wo_context>`

Stereotype category:
`<gs_category>`

Stereotype intensity ($0.00 - 1.00$):
`<gs_value>`

`# if_contextual_information_is_provided:`
Context (for interpretation only):
`<context>`

Respond with ONLY the following JSON structure:
{
`"text"`: ”new Italian sentence here”,
`"gs_value"`: `<gs_value>`,
`"gs_category"`: `<gs_category>`
}

A.2. English Translation of Data Augmentation Example (ID: GS0458)

The translations are provided for illustrative purposes only and do not affect the evaluation, which is conducted on the Italian texts.

Original GSI Dev Sample (ID: GS0458)

`text_wo_context`: "Me, mom, and you, I can't take it anymore etc. etc. etc. We also have to say that marriages used to last a lifetime, and you didn't hear every day like today about femicides. People joke A LOT about other people's feelings, because the cake is eaten before the party, and since there is no longer shame about ANYTHING, it is very easy to say it wasn't good and there is no longer desire. But if you eat it little by little, the cake is desired and good, a few words to the wise!"
`context`: "Comment on an old wedding photo in which both the bride and groom and the guests appear very serious, with the caption: *A wedding from another era.*"
`gs_value`: 0.25
`gs_category`: sexual

Augmented Counterpart (Synthetic Sample)

`text`: "When I hear my grandmother talk about weddings in the past, she repeats that respectable girls knew how to wait and did not give themselves away immediately; maybe that is also why relationships lasted."
`gs_value`: 0.25
`gs_category`: sexual