Festa at GSI:detect: In-Context Learning vs. Fine-Tuning for Gender Stereotype Detection

Donato Festa

University of Bari Aldo Moro, Italy

Abstract

This paper describes the system developed for the GSI:detect task at EVALITA 2026. The goal of the task is to quantify the intensity of gender stereotypes in Italian text (Main Task) and classify them into predefined categories (Subtask). We propose a comparative study utilizing **Google Gemini 2.5 Flash**, contrasting In-Context Learning strategies (Zero-Shot and Few-Shot) against a Fine-Tuning approach based on **UmBERTo**. Our results reveal a significant trade-off between classification performance and intensity estimation. The Few-Shot approach, enhanced by an ensemble strategy, achieved better performance in stereotype categorization (Micro F1: 0.6708), demonstrating superior capability in detecting implicit biases, particularly in the "Relational" category. Conversely, the best Zero-Shot configuration proved more effective for intensity regression (NMSE Score: 0.6205). This paper details the prompt engineering strategies, the experimental setup, and provides an error analysis of the different configurations.

Keywords

Gender Stereotype Detection, Large Language Models, Few-Shot Learning, Prompt Engineering, Ensemble Learning

1. Introduction

Gender stereotyping in language is a widespread issue that reinforces social inequalities and harmful biases. The GSI:detect task [1], organized within the EVALITA 2026 campaign [2], addresses this challenge by proposing a comprehensive evaluation of stereotypes on Italian texts. The task involves estimating the intensity of stereotypes on a continuous scale (Main Task) and classifying them into predefined categories (Subtask).

In this work, we place a particular emphasis on the categorization of gender stereotypes (the Subtask), as the identification of specific bias types is critical for developing targeted mitigation strategies and understanding the semantic nuances of various stereotype forms. Understanding whether a sentence conveys a stereotype related to competence rather than physical appearance, for instance, provides a more granular and actionable analysis of gender bias in social media.

Stereotypes can range from explicit remarks (e.g., regarding physical appearance) to subtle and implicit associations (e.g., questioning professional competence based on gender), making automatic detection a complex semantic challenge.

In this work, we explore the capabilities of Large Language Models (LLMs), specifically **Google Gemini 2.5 Flash**, comparing In-Context Learning strategies (Zero-Shot and Few-Shot) against a standard Fine-Tuning approach based on the **UmBERTo** architecture. Our contribution can be summarized as follows:

1. We conduct a systematic comparison of three distinct approaches for Italian stereotype detection:
    - **Zero-shot** prompting, where the model is given only the task description without examples.
    - **Few-shot** In-Context Learning, where the model is provided with a few annotated examples to guide its predictions.
    - **Fine-Tuning**, where we adapt a pre-trained UmBERTo model to the specific task using the provided dataset.

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

d.festa5@studenti.uniba.it (D. Festa)

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

2. We investigate the impact of context availability by experimenting with different numbers of examples ($k \in \{6, 10, 12, 14\}$) in the Few-Shot setting to determine the optimal trade-off between prompt complexity and accuracy.
3. For each approach we propose (at least) an ensemble strategy that aggregates predictions from multiple runs to improve classification stability and reduce variance in intensity regression.

2. Task and Dataset

The GSI:detect challenge targets the identification of gender stereotypes in Italian texts. It is divided into two parts:

- **Main Task (Intensity Prediction):** A regression task where the system must predict a scalar value (‘gs_value‘) representing the degree of stereotyping, ranging from 0 (no stereotype) to 1 (extreme stereotype).
- **Subtask (Stereotype Categorization):** A multi-class classification task where the system must assign a specific label to a sentence. The categories include: *Physical*, *Sexual*, *Role*, *Relational*, *Competence*, and *Personality*. If no stereotype is present, the label is *No*.

2.1. Dataset

The dataset provided by the organizers consists of User-Generated Content (UGC) collected from online sources (e.g., social media comments), characterized by informal language, emojis, and non-standard grammar. The data is split into:

- A **Development Set** of **200 sentences**, which we utilized for training and validation (via Cross-Validation) in the Fine-Tuning approach, and as the source of examples for the Few-Shot prompting.
- A **Test Set** of **810 sentences**, used for the final evaluation.

2.2. Data Preprocessing and Augmentation

Given the limited size of the development set and the natural class imbalance it exhibits, we implemented a custom **Conservative Data Augmentation** pipeline to mitigate this during the Fine-Tuning phase. Unlike aggressive augmentation techniques that might alter the semantic meaning (e.g., back-translation), our approach utilizes three safe transformations:

1. **Synonym Replacement:** Substituting a single adjective or noun with a contextually appropriate synonym (e.g., *“donna”* → *“signora”*).
2. **Safe Insertion:** Injecting neutral adverbs (e.g., *“sicuramente”*, *“ovviamente”*) at the beginning or end of the sentence.
3. **Pattern Paraphrasing:** Rewriting common syntactic patterns (e.g., *“sono sempre”* → *“risultano sempre”*).

We oversampled the minority classes to reach a target of 60 samples per class. For In-Context Learning, no augmentation was applied.

2.3. Evaluation Metrics

The official evaluation metrics are:

- **NMSE Score:** The official ranking metric for the Main Task. It is derived from the Normalized Mean Squared Error as $1/(1 + NMSE)$, providing a score between 0 and 1 where higher is better.
- **Micro F1-Score:** The official ranking metric for the Subtask (Classification). It evaluates classification performance globally by aggregating true positives, false negatives, and false positives across all samples, effectively giving equal weight to each *instance*.

- **Macro F1-Score:** In addition to the official Micro F1-Score we also used Macro F1, which calculates the arithmetic mean of the per-class F1 scores. This metric treats all classes equally regardless of their frequency, highlighting the model's ability to detect minority stereotype categories.

3. System Description

To address the GSI:detect challenge, we developed a system that contrasts Fine-Tuning with In-Context Learning (ICL) strategies utilizing Large Language Models. We focused on Google Gemini 2.5 Flash as our generative engine due to its strong reasoning capabilities and efficiency.

3.1. Fine-Tuning Approach (UmBERTo)

We employed **UmBERTo** [3] (`Musixmatch/umberto-commoncrawl-cased-v1`), a RoBERTa-based model pre-trained on a large corpus of Italian CommonCrawl data. Instead of a simple linear probe, we designed a custom **Multi-Layer Perceptron (MLP) head** to process the '[CLS]' token embedding. The head consists of a Dropout layer ($p = 0.1$), a dense projection to 128 units with Tanh activation, a second Dropout, and a final linear layer.

We fine-tuned two separate models:

- **Regression Model (Main Task):** The final layer is followed by a **Sigmoid** activation function to constrain the output strictly within the $[0, 1]$ range. We optimized the model using *Mean Squared Error* (MSE) loss.
- **Classification Model (Subtask):** The final layer returns the raw logits for the 7 target classes. We optimized the model using *Cross-Entropy Loss* (weighted to handle residual imbalance).

Training was performed for 15 epochs (Batch Size=16, LR=$3e^{-5}$, AdamW optimizer) using Stratified K-Fold Cross-Validation ($k = 5$).

3.1.1. Experimental Configurations (Runs).

To explore the trade-offs between stability and specialization, we submitted 5 distinct variations based on the stratified k-fold models. Note that since the *No* label was not permitted in the final submission, all runs mapped non-stereotype predictions to valid categories, but with different strategies:

- **Run 1 (Standard Ensemble):** A weighted ensemble of all 5 folds where, for each sample, the system selects the **valid** category with the highest aggregated probability score.
- **Run 2 (Mean Ensemble):** An unweighted ensemble using simple arithmetic averaging (intensity) and majority voting (categories) among valid classes.
- **Run 3 (Best Single Model):** Inference using only the single model (Fold 2) that achieved the highest validation scores.
- **Run 4 (Single Fold):** A run using only the second best model (Fold 3).
- **Run 5 (Role Fallback):** A weighted ensemble (Run 1) with a deterministic heuristic: instead of selecting the next most probable class, whenever the model predicts *No*, the system **always** defaults to *Role* (the most frequent category).

3.2. In-Context Learning (Gemini)

We leveraged Gemini 2.5 Flash [4] via API with generation `temperature` set to 0 for reproducibility. We disabled safety filters (`BLOCK_NONE`) to process the potentially offensive content of the dataset.

3.2.1. Prompt Engineering Strategies

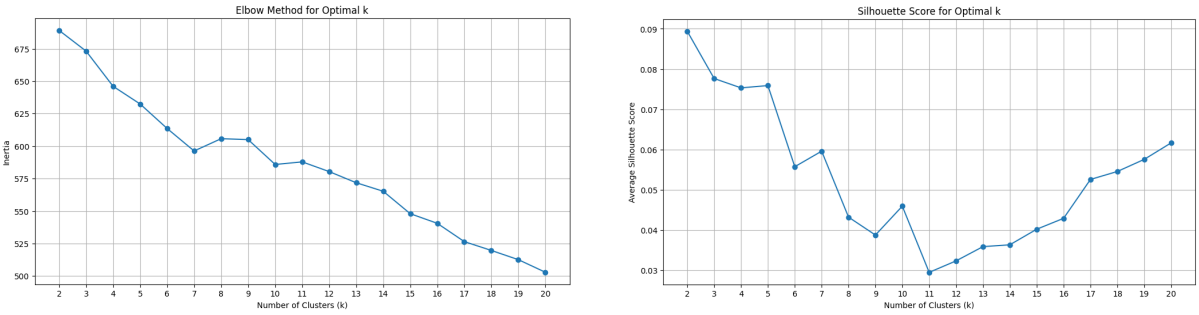
We adopted an iterative Prompt Engineering approach, testing distinct strategies for both Zero-Shot and Few-Shot settings.

Zero-Shot Iterations. Our experimentation began with a baseline template (V1). However, preliminary tests revealed suboptimal performance, characterized by frequent hallucinations and a lack of sensitivity. Consequently, we discarded V1 and evolved our approach into four distinct refined templates (V2–V5):

- **False Positive Reduction (V2):** An English prompt explicitly instructing the model to distinguish between factual topics (e.g., cooking) and stereotypes (e.g., gender obligations), aiming to improve precision.
- **Cultural Adaptation (V3):** A prompt entirely in **Italian**, defining categories with culturally specific examples to better align with the target language.
- **Chain-of-Thought (V4 & V5):** We implemented a *Chain-of-Thought* (CoT) mechanism by requiring the model to output a `"reasoning"` field in the JSON before the final label. This forces the model to articulate its logic (e.g., *"Is this an offensive generalization?"*) before committing to a score. We tested this strategy in both Italian (V4) and English (V5).

Few-Shot Strategies. Instead of randomly selecting examples, we implemented a data-driven selection process to construct the optimal few-shot prompt. We experimented with three selection strategies:

1. **Manual Selection ($k = 6$):** A baseline set of hand-picked examples covering the main categories.
2. **Semantic Clustering ($k = 10, 12$):** We generated sentence embeddings for the entire development set using `paraphrase-multilingual-mpnet-base-v2`, based on the Sentence-BERT architecture [5]. We then applied K-Means clustering to group semantically similar sentences and selected the samples closest to each cluster centroid. This ensures the model is exposed to the maximum semantic diversity of the dataset.
3. **Hybrid Strategy ($k = 14$):** A "Best-of-Both-Worlds" approach. We combined the optimal clustering set ($k = 12$) with two manually selected "corrective" examples, specifically chosen to address common failure cases (e.g., non-stereotypical sentences often misclassified as stereotypes).



(a) Elbow Method analysis.

(b) Silhouette Score analysis.

Figure 1: Clustering analysis on development set embeddings. The combined view of the Elbow curve and Silhouette scores guided the selection of $k \in \{10, 12\}$ to balance semantic granularity and cluster separation.

3.3. Ensemble Strategy

To mitigate the variance inherent in individual models and maximize robustness, we implemented two distinct ensemble strategies tailored to the nature of each approach.

3.3.1. LLM Ensemble (Zero-Shot & Few-Shot)

For the In-Context Learning approaches, we aggregated the outputs of the four distinct prompt variations (V2–V5 for Zero-Shot; $k \in \{6, 10, 12, 14\}$ for Few-Shot) using an unweighted voting mechanism:

- **Intensity (Main Task):** We computed the simple arithmetic mean of the predicted scores.
- **Categorization (Subtask):** We applied a standard Majority Voting. To resolve ties (e.g., two votes for *Role* and two for *Personality*), we implemented a deterministic tie-breaking rule: the system defaults to the prediction of the highest-performing single run (acting as the "leader" model).

Finally, to strictly adhere to the challenge submission format which disallowed the *No* label for the Subtask, we applied a post-processing step where any residual non-stereotype prediction was mapped to the *Role* category (the most frequent class in the training set).

3.3.2. Fine-Tuning Ensemble

For the fine-tuned UmBERTo models, we adopted a more aggressive aggregation strategy to boost sensitivity:

- **Weighted Voting:** Predictions from the 5 cross-validation folds were aggregated using a weighted voting scheme, where the weight of each model was proportional to its F1-score (for classification) or NMSE (for regression) on the validation set.
- **Forced Role Fallback:** To comply with the final challenge guidelines, which required the prediction of one of the six stereotype categories (excluding the *No* label), for the final run (Run 5) of this approach, we implemented a mandatory fallback mechanism. Whenever the weighted voting resulted in a *No* prediction, the system automatically defaulted to *Role* (the most frequent class in the training distribution) to ensure valid submission outputs.

4. Experimental Setup

To ensure reproducibility, we detail below the computational environment and the hyperparameters used for both the Fine-Tuning and In-Context Learning experiments.

4.1. Computational Environment

All experiments were conducted on a local workstation running **Windows 11**. The hardware configuration included an **Intel Core i7-14700HX** processor, **32GB of RAM**, and an **NVIDIA GeForce RTX 4070** GPU with 8GB of VRAM. We implemented the system using Python, leveraging the `PyTorch` framework and the Hugging Face `Transformers` library for the fine-tuning pipeline, and `scikit-learn` for the clustering operations.

4.2. Hyperparameters and Configuration

- **LLM Generation (Gemini):** For all prompt-based runs, we utilized the `google-generativeai` Python client. We set the generation `temperature` to 0 to ensure deterministic outputs and disabled safety filters (`HarmBlockThreshold.BLOCK_NONE`) to process the sensitive nature of the dataset without refusals.
- **Fine-Tuning (UmBERTo):** We trained the model for 15 epochs using the **AdamW** optimizer with a learning rate of $3e^{-5}$ and a linear scheduler (10% warmup). To optimize memory usage on the local GPU, we set the **training batch size to 16**. For the ensemble inference phase, we increased the batch size to 32 to maximize throughput.

5. Results

We evaluated the performance of our three approaches (Fine-Tuning, Zero-Shot, and Few-Shot) on the official test set. Table 1 summarizes the results for all 15 experimental runs, reporting the official ranking metric (Micro F1) alongside the internal Macro F1 and NMSE scores.

Table 1
Main results on the GSI:detect Test Set. The table compares the **Micro F1-Score** (Official Ranking Metric for the Subtask), **Macro F1-Score** (Internal Evaluation), and **NMSE Score** (Main Task) across all configurations. The best result for each metric is highlighted in **bold**.

Approach	Configuration (Run)	Micro F1	Macro F1	NMSE Score
Fine-Tuning	Run 3: Single Best Fold (Fold 2)	0.4789	0.4577	0.5409
	Run 4: Single Fold (Fold 3)	0.4525	0.4272	0.5527
	Run 2: Ensemble (Mean)	0.4894	0.4707	0.5604
	Run 5: Ensemble (Forced Role)	0.4683	0.4529	0.5604
	Run 1: Standard Ensemble	0.5211	0.5038	0.5604
Zero-Shot	V2: False Positive Red.	0.5898	0.5515	**0.6205**
	V3: Italian Cultural	0.5863	0.5427	0.6096
	V4: CoT (Italian)	0.5687	0.5227	0.5548
	V5: CoT (English)	0.6004	0.5605	0.5572
	Ensemble (Voting)	0.5968	0.5520	0.6177
Few-Shot	$k = 6$ (Manual Selection)	0.6356	0.6271	0.4996
	$k = 10$ (Clustering)	0.6496	0.6345	0.5605
	$k = 12$ (Clustering)	0.6567	0.6526	0.5650
	$k = 14$ **(Hybrid Strategy)**	**0.6708**	**0.6687**	0.5500
	Ensemble (Voting)	0.6655	0.6562	0.5717

5.1. Performance Analysis

The results highlight a significant performance gap between In-Context Learning and Fine-Tuning. The Few-Shot approach consistently outperformed all other methods, with the **Hybrid Strategy (**$k = 14$**)** achieving the best performance among the methods we tested in the official metric (**Micro F1: 0.6708**).
This validates the effectiveness of combining semantic clustering with manually selected corrective examples. Interestingly, while Zero-Shot models generally underperformed in classification, the English Chain-of-Thought configuration (V5) achieved a competitive Micro F1 of 0.6004, suggesting that advanced reasoning prompts can partially compensate for the lack of examples. Fine-Tuning proved to be the weakest approach (best Micro F1: 0.5211), likely due to the limited size of the training set (200 examples) which was insufficient for the model to generalize well on complex categories.

5.2. Error Analysis by Category

To understand the specific strengths and weaknesses of each approach, we compared the per-category F1-scores of the best performing run for each setting. The quantitative breakdown is reported in Table 2, while Figure 2 provides a visual comparison of the performance trends.

The "Relational" Breakthrough. The most striking improvement concerns the *Relational* category. The Zero-Shot model failed almost completely here (F1 0.14), likely because relational stereotypes (e.g., defining a woman solely as a wife) are subtle and structural. The Few-Shot Hybrid strategy, by including specific examples of this category, raised the score to 0.58, proving that In-Context Learning can effectively patch specific semantic blind spots.

Explicit vs. Implicit. All models performed relatively well on *Competence* and *Physical* stereotypes, which are often marked by explicit keywords.

Table 2
Category-wise F1-Scores for the best run of each approach.

Category *(Best Run)*	**Fine-Tuning** *(Ensemble)*	**Zero-Shot** *(V5 - CoT)*	**Few-Shot** *(Hybrid k14)*
Competence	0.62	**0.69**	0.67
Personality	0.49	0.56	**0.66**
Physical	0.58	0.74	**0.79**
Relational	0.49	0.14	**0.58**
Role	0.52	0.62	**0.65**
Sexual	0.32	0.61	**0.66**

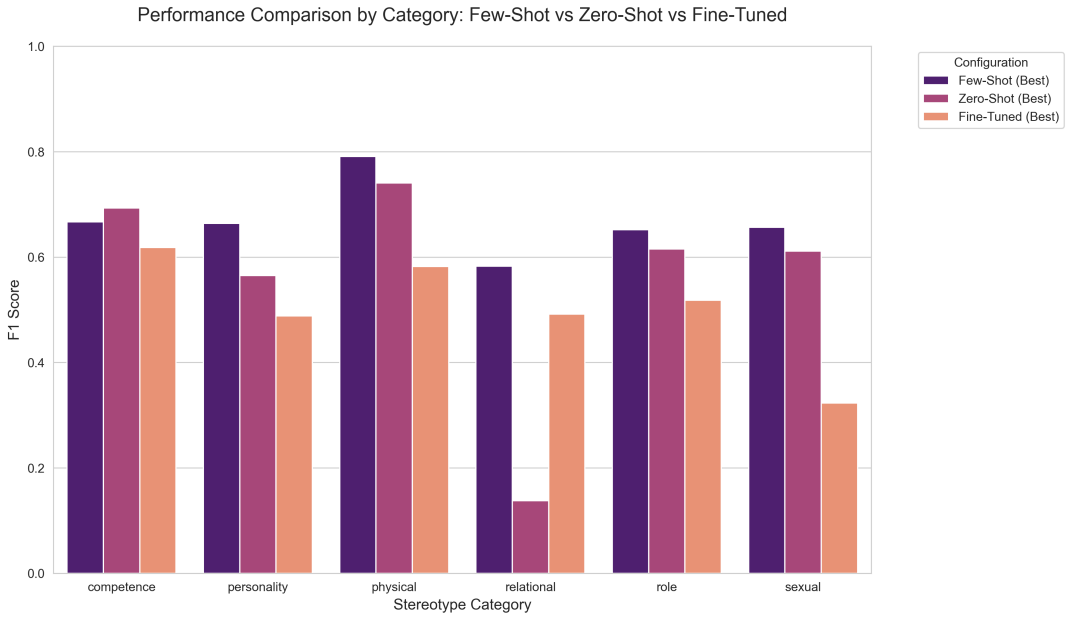


Figure 2: Performance comparison per stereotype category (F1-Score). The **Few-Shot Hybrid** strategy demonstrates superior robustness, particularly in the challenging *Relational* category where Zero-Shot approaches fail. Conversely, Fine-Tuning shows significant degradation in the *Sexual* category due to data scarcity.

Fine-Tuning Failure Cases. The Fine-Tuned model struggled severely with the *Sexual* category (F1 0.32). This suggests that the small training set did not contain enough variety of sexual stereotypes for the model to generalize, whereas the LLM (Gemini) could leverage its vast pre-training knowledge to detect them effectively even without many examples (Zero-Shot F1 0.61).

6. Conclusion

In this paper, we presented a comprehensive study on detecting gender stereotypes in Italian texts for the EVALITA 2026 GSI:detect task. By systematically comparing Fine-Tuning and In-Context Learning approaches, we highlighted the strengths and limitations of current NLP paradigms when applied to low-resource and nuanced tasks.

Our findings indicate that in scenarios with limited training data (200 examples), prompt-based learning with Large Language Models significantly outperforms traditional fine-tuning. Specifically, our **Few-Shot Hybrid Strategy** ($k = 14$) achieved the best performance among the methods we tested (Micro F1: 0.6708), demonstrating that the careful selection of semantic and corrective examples is more impactful than model retraining. Furthermore, our error analysis revealed that while explicit stereotypes (Physical, Competence) are easily detected, implicit biases (Relational) remain a challenge,

requiring targeted interventions like specialized few-shot examples, which successfully bridged the performance gap in our best run.

Declaration on Generative AI

During the development of this work, Generative AI tools were used as support. In particular, some system executions rely on the Google Gemini API (LLM). Additionally, AI-based tools were occasionally employed to help refine code snippets and to improve the clarity of the English writing. All content was carefully reviewed and revised by the author, who takes full responsibility for the correctness and originality of the final report.

Code Availability

The source code, experimental notebooks, and datasets used in this work are publicly available at: https://github.com/DonatoFe11/gsi-detect-evalita.

References

[1] Comandini, G., Speranza, M., Brenna, S., Testa, D., Cavagnoli, S., & Magnini, B. (2026). GSI:detect at EVALITA 2026: Overview of the Task on Detecting Gender Stereotypes in Italian. Proceedings of the Ninth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2026). Bari, Italy. CEUR.org.

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

[3] Parisi, L., Francia, S., & Magnani, M. (2020). Musixmatch at EVALITA 2020: UmBERTo for Sentiment Analysis. Proceedings of EVALITA 2020.

[4] Google DeepMind (2024). Gemini: A Family of Highly Capable Multimodal Models. arXiv preprint arXiv:2312.11805.

[5] Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. Proceedings of EMNLP-IJCNLP 2019.