CLiC at EXIST 2025: Combining Fine-tuning and Prompting with Learning with Disagreement for Sexism Detection

Notebook for the EXIST Lab at CLEF 2025

Pol Pastells[1,*,†], Mauro Vázquez[1,†], Mireia Farrús[1,2] and Mariona Taulé[1,2]

[1]Centre de Llenguatge i Computació (CLiC), Universitat de Barcelona
[2]Institute of Complex Systems (UBICS), Universitat de Barcelona

Abstract

We present the CLiC group's participation in the EXIST 2025 shared task, focusing on sexism detection in social media content. Our work addresses three subtasks: sexism identification (Task 1.1), source intention detection (Task 1.2), and sexism categorization (Task 1.3). We employed BERT [1] fine-tuning for Task 1.1 (binary sexism classification) and DSPy-based prompt optimization for Tasks 1.2 and 1.3, leveraging the initial classification outcomes. A key aspect of our approach is a Learning with Disagreement framework that utilizes annotator demographic information to model diverse perceptions of sexism. Our experimental design included three runs, exploring BERT-based methods for Task 1.1 and contrasting prompt-based methods, including variants with annotator information and Retrieval-Augmented Generation (RAG), for the subsequent tasks. Results demonstrate that BERT fine-tuning significantly surpassed prompt-based methods for Task 1.1, where our approach secured 9th place out of 67 participants in the soft label category. The integration of annotator information proved vital, leading to substantial performance gains across all tasks. The impact of RAG, however, remained inconclusive. These findings highlight the enduring effectiveness of fine-tuned models for core classification, while emphasizing the necessity of annotator-aware approaches for handling subjective concepts like sexism. Our code is available at https://github.com/clic-ub/EXIST_2025.

Keywords

Sexism identification, sexism categorization, learning with disagreement, prompting

1. Introduction

Sexism detection in social media has become increasingly important as online platforms struggle with harmful content moderation. The EXIST 2025 challenge [2, 3] addresses this need through multimodal evaluation, though our participation focused specifically on the textual components: Task 1.1 (sexism identification), Task 1.2 (source intention detection), and Task 1.3 (sexism categorization). While transformer-based fine-tuning has dominated recent EXIST editions, large language models (LLMs) have achieved state-of-the-art performance across numerous NLP tasks through prompt engineering. This creates an important methodological gap: shared tasks continue relying on fine-tuning approaches despite LLMs' broader success with prompt-based methods.

Motivated by this, our primary objective was to investigate the performance of prompt-based methods, specifically using DSPy [4] for systematic prompt optimization, in text classification problems within the EXIST framework. DSPy automatically generates and refines prompts through latent space exploration, offering a more principled comparison with traditional fine-tuning than manual prompt engineering.

We also employed a BERT fine-tuning approach for Task 1.1. This served as a well-tested baseline for classification tasks (see [5] for example) and provided a strong foundation of binary sexism classification upon which to build for Tasks 1.2 and 1.3. Comparing this fine-tuning approach with the prompting

CLEF 2025 Working Notes, 9 – 12 September 2025, Madrid, Spain
*Corresponding author.
†These authors contributed equally.
pol.pastells@ub.edu (P. Pastells); mauro.vazquez@ub.edu (M. Vázquez); mfarrus@ub.edu (M. Farrús); mtaule@ub.edu (M. Taulé)

techniques allowed us to evaluate the viability of relying solely on methods like few-shot tuning, example selection, and instruction optimization.

Beyond evaluating different modeling paradigms, we specifically aimed to address the importance of incorporating annotator information and retrieval-augmented generation (RAG) on model performance. Recognizing that sexism perception varies across demographic groups, our approach integrates annotator perspectives through a Learning with Disagreement (LeWiDi) framework. We systematically evaluated whether incorporating these annotator perspectives and RAG improves performance across the different modeling approaches tested.

To investigate these research questions, we designed three distinct runs for each task, summarized in Table 1. These runs allowed us to compare the BERT baseline, prompting with RAG, prompting with annotator information (AnI), and a combination of prompting, RAG, and AnI across the three EXIST subtasks.

Table 1
Runs and Methods Across Tasks. AnI: Annotator Information. RAG: Retrieval-Augmented-Generation

	Task 1	Task 2	Task 3
Run 1	BERT	Fewshot+RAG	Fewshot+RAG
Run 2	Fewshot+AnI	Fewshot+AnI	Fewshot+AnI
Run 3	Fewshot+RAG+AnI	Fewshot+RAG+AnI	Fewshot+RAG+AnI

2. Related Work

The EXIST challenge has driven significant advances in automated sexism detection since its inception [6]. Notable approaches from recent editions include multilingual and monolingual BERT [1] models with ensemble strategies, with winning systems typically employing combinations of transformer models such as mBERT, XLM-RoBERTa [7], and RoBERTa [8] variants [5, 9]. These approaches have consistently demonstrated that transformer-based models outperform traditional machine learning methods for sexism detection tasks.

Traditional annotation approaches favor majority opinion when multiple annotators disagree, potentially overlooking valuable insights that could enhance model effectiveness. The Learning with Disagreement (LeWiDi) framework [10] addresses this limitation by incorporating annotator perspectives directly into the learning process, moving beyond simple majority voting to leverage the full spectrum of annotator disagreement as a source of information rather than noise.

Despite large language models (LLMs) achieving state-of-the-art performance across numerous NLP tasks, shared tasks like EXIST continue to be dominated by BERT-based fine-tuning approaches. There has been limited exploration of prompt engineering techniques for sexism detection, with only one attempt at using prompt engineering on EXIST 2024 [11]. This gap between the broader NLP landscape and shared task methodologies leaves systematic prompt optimization and comprehensive comparisons with fine-tuning approaches underexplored. Our work addresses this gap by comparing BERT fine-tuning with DSPy-based automated prompt optimization while incorporating the Learning with Disagreement framework across multiple sexism detection subtasks.

3. Datasets

The EXIST 2025 Task 1 dataset contains 6,920 training tweets (3,660 Spanish, 3,260 English) with annotations from 6 demographically diverse annotators per instance. Each annotator is characterized by age, gender, ethnicity, education level, and country, enabling perspective-aware modeling. The development and test sets have 1,038 and 2,076 instances, respectively. The instances provided include the language of the tweet (`lang`), the content (`text`), and annotator demographics (gender, age, ethnicity,

study level, country), for the 6 annotators involved in each example. In terms of age and gender, the dataset is completely balanced, and for the other annotator details, there is no apparent bias.

3.1. Preprocessing

For both training and inference, we preprocessed tweets by removing URLs and user mentions, converting emojis to their textual descriptions, and retaining all hashtags. Following the LeWiDi framework, we leverage annotator disagreement as signal rather than noise. Each original instance was expanded into 6 annotator-specific examples (see Section 4.2).

4. Methodology

Our approach leverages two distinct methodologies to tackle the three subtasks of sexism detection. For Task 1.1 (binary sexism identification), we employ traditional BERT fine-tuning with annotator-aware prompts to establish a strong baseline classification. For Tasks 1.2 and 1.3 (multiclass and multilabel classification), we use DSPy's prompt optimization framework, building upon the binary predictions from Task 1.1. This hybrid approach allows us to compare the effectiveness of fine-tuned models versus prompt-engineered large language models while systematically evaluating the impact of annotator information and retrieval-augmented generation across all tasks.

All experiments were conducted on a single RTX 4090 GPU with 24GB VRAM.

4.1. DSPy and MIPROV2

DSPy is a Python framework [4] that aims to improve prompt quality. Instead of dealing with hard-coded prompts, it focuses on developing a systematic parameterized approach to optimize each component using actual code. The parameters for each module in the pipeline include the LLM, the input and output fields, and the few-shot examples.

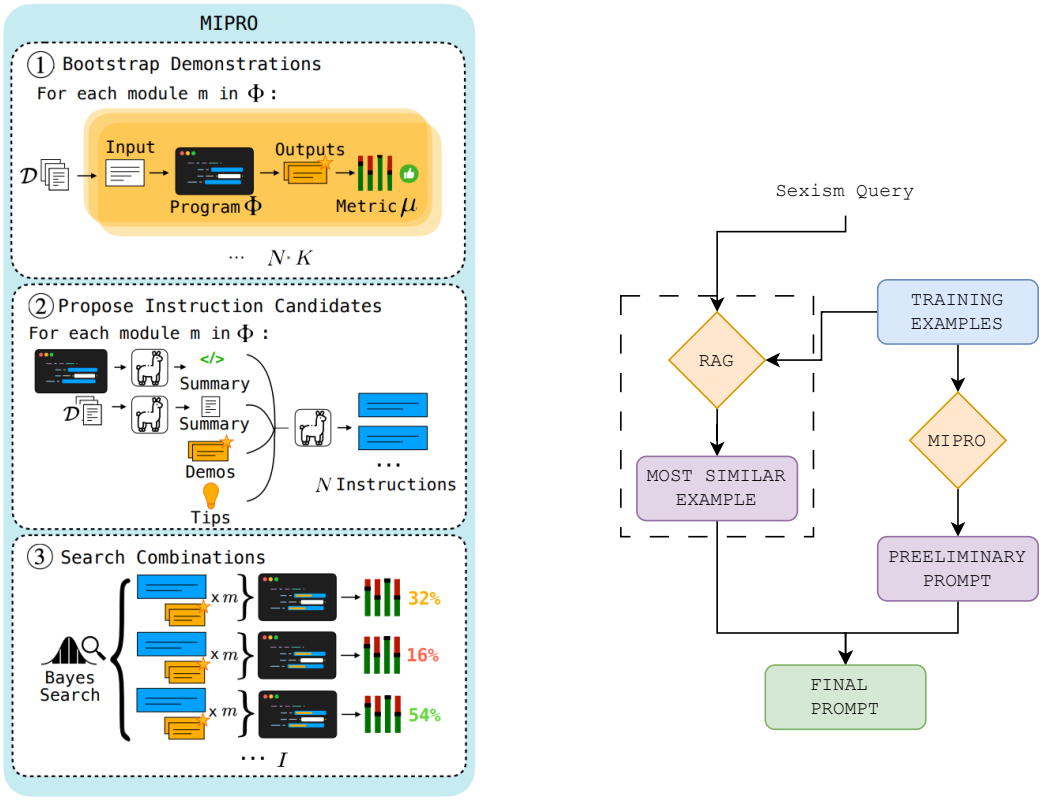
We were motivated to pursue prompt optimization over weight optimization by the strong results in the Better Together paper [12]. Their core finding is that jointly optimizing prompts and weights improves performance more than either alone. However, they also show that prompt optimization alone often outperforms weight optimization across three models and three tasks, and in some cases, it even rivals the combined approach.

As an optimizer, we selected MIPROv2, the faster and more accurate version of MIPRO [13], according to DSPy's benchmarks. At its core, it uses an iterative loop where it generates some prompt instructions as well as a set of few-shot examples, tests this prompt on a batch of training data, and evaluates the performance using a provided metric.

To generate satisfactory instructions (see Figure 1a), MIPROv2 may use another LLM called "proposer", the same LLM in our case, that leverages the available context and information for the task. This includes summaries of the data properties, input/output descriptors, a description of the pipeline of prediction, and some successful task executions. It also receives a history of previously tested prompts along with their performance. To obtain demonstrations, the optimizer performs bootstrapping on the available training data to get candidates and then generates sets of them via random sampling. Finally, it uses Bayesian Optimization to search among the net of possibilities, assigning performance scores to prompt components.

The implementation of MIPROv2 present in DSPy allows for flexible configuration options based on the task and available data. The `max_labeled_demos` parameter represents the maximum number of few-shot examples taken from the training set. Furthermore, `max_bootstrapped_demos` controls how many of them can be generated via bootstrapping (augmented). Equally important, MIPRO has three levels of exploration: light, medium, and heavy.

DSPy also offers the possibility of using predefined modules to produce outputs. In our case, we used `ChainOfThought`, which forces the model to output a reasoning field before making a prediction, increasing explainability and taking advantage of more test-time computation.



(a) (b)

Figure 1: (a) MIPRO algorithm steps, from [13]. (b) Program flow for Task 1.1. The RAG block with dashed lines is optional.

To perform optimization on the prompts and inference over the tasks, we used the open-source model Qwen2.5-7B-Instruct [14].

4.2. Task 1.1: Sexism Identification in Tweets

Task 1.1 was a binary classification problem, where each tweet must be classified as either sexist or non-sexist.

4.2.1. BERT models

We fine-tuned ModernBERT-large [15] with the English tweets and RoBERTa-large-BNE [16] with the Spanish ones. We decided to add the given annotator information for context, as providing context to BERT models may improve the results [17], as well as to take into account the possible biases each annotator may have. Thus, we did both training and predictions using each annotator information. We cleaned the annotator information to construct a prompt that was fed to the BERT models (technically we modified the text, as BERT is not an instructed model and does not take prompts as inputs). For English, the prompt had the structure shown in Listing 1, which (for text id 600,253) leads to Example (a). For Spanish, we translated the annotator information and used a Spanish prompt. This way, we obtained 6 predictions for each text, that we can compare with the 6 human annotations.

Listing 1: Prompt generation function for English text

```
english_prompt = ("Given the following text: \n{ " + row.text + " }\n"
    f"A {row.age} year-old {row.ethnicities} "
    f"{row.gender} from {row.countries} {row.study_levels} "
    "perceives it as sexist?")
```

(*a*) Given the following text:
{ Its nice that young women have a rapist to look up to! She really is an icon of empowerment. Women aren't guilty of rape if they identify as innocent. }
A 46+ year-old White or Caucasian woman from Spain with a Bachelor's degree perceives it as sexist?

Furthermore, the models were fine-tuned for a regression task using soft labels. The global soft label for each text was computed as the average of the 6 annotators (see Equation 1), and the soft label for each annotator was set to the average of the global soft label and the vote of the specific annotator (the hard label, which can only be 0 or 1), as shown in Equation 2.

$$\text{SoftLabel}^j = \frac{1}{6} \sum_{i \in Annotators} \text{HardLabel}_i^j, \tag{1}$$

$$\text{SoftLabel}_i^j = \frac{\text{SoftLabel}^j + \text{HardLabel}_i^j}{2}, \tag{2}$$

where j refers to the text index.
Both models were trained using a context length of 256 tokens for a maximum of 5 epochs, with a batch size of 32. We validated every 100 steps and kept the best model. The learning rate for RoBERTa-large-BNE was set to 5×10^{-6} and for ModernBERT-large to 1×10^{-5}.

4.2.2. Using RAG and Annotator Information

In this particular run, to optimize the initial prompt, we used MIPROv2 with the `heavy` configuration, accuracy as the training metric, `max_bootstrapped_demos` $= 4$ and `max_labeled_demos` $= 6$. We also differentiated between languages, creating two separate prompts.
For each inference example, a Retrieval-Augmented Generation (RAG) step was applied to the initial prompt. This process, illustrated in Figure 1b, involved retrieving the most similar example from the training set. The retrieved example, along with its soft labels (representing the combined predictions of the 6 annotators), was then added to the prompt. This provided the model with insight into how similar queries were handled during training. Tweet text similarity was calculated using the 'all-MiniLM-L6-v2' model (the specific model used is a fine-tuned version of [18] created by SBERT).
Then, with the specific prompt for each test example, we predict whether the text is sexist or not for each of the 6 annotators. To obtain the soft and hard labels using the predictions of each annotator (transformed into a binary representation $0 - 1$) we used the intuitive approach:

$$\text{SoftLabelPred}^j = \frac{1}{6} \sum_{i \in \text{Annotators}} \text{Prediction}_i^j, \tag{3}$$

$$\text{HardLabelPred}^j = \begin{cases} 0 & \text{if SoftLabelPred}^j \leq 0.5 \\ 1 & \text{if SoftLabelPred}^j > 0.5. \end{cases} \tag{4}$$

4.2.3. Other Considerations

Besides the usage of plain classes as output, we also considered other structures. This includes: forcing the model to output a confidence value for its prediction (in $[0, 1]$), using integers to display the level of sexism (in $\{0, 1, ..., 10\}$) instead of binary classification, similarly using floats, and explicitly asking for a reasoning field to justify the prediction.
The usage of confidence and reasoning was kept at the inference level, as it forced the model to reason further, and as it also increases explainability. On the other hand, we discarded the usage of integers and floats as we perceived a certain bias towards values like 0.5, 7 or 10. These tendencies are probably due to mode collapse or training biases, as it can be seen in [19].

4.3. Task 1.2: Source Intention in Tweets

Task 1.2 corresponds to a multiclass problem where each sexist tweet must be classified as either *judgmental*, *direct*, or *reported* sexism. As a starting point for this task, we used the binary classification from Task 1.1 that used a BERT fine-tuning, as we had already yielded good results with such techniques in the past.

To propagate the results, we considered two scenarios. If the soft label from Task 1.1 does not surpass the 0.5 threshold, this means it would have been classified as a non-sexist tweet in Task 1.2 as well (see Equation 5). Therefore, we did not try to predict its class. If the value was over the threshold, we predicted the class that suited the criteria the best, normalized accordingly, and assigned the same value to the non-sexist class from Task 1.2 (Equation 6).

$$\text{Pred}_{1.2}[\text{No Class}] = \text{Pred}_{1.1}[\text{Not Sexist}] \tag{5}$$

$$\text{Pred}_{1.2}[\text{Class}] \leftarrow \text{Pred}_{1.2}[\text{Class}] \times \text{Pred}_{1.1}[\text{Sexist}] \tag{6}$$

The prompt optimization process for the English and Spanish versions, was performed using MIPROv2 with the medium configuration, accuracy as the training metric, `max_bootstrapped_demos` $= 1$ and `max_labeled_demos` $= 6$.

To be able to analyze the impact of each of the elements present in the few-shot prompt construction (RAG and Annotator Specific Prediction), we performed the following runs: only RAG, only annotator information disclosed on the prompt, and both RAG and annotator information. This approach follows the same scheme as Figure 1b, changing the output field to be one of the sexist classes instead of just binary classification.

To generate the soft labels for each class we used the same concept as in Equations 3 and 4. However, for each class and annotator the associated prediction would be 1 if the class was the chosen one and 0 otherwise. Intuitively, the hard label was selected to be the class with the highest soft label.

4.4. Task 1.3: Sexism Categorization in Tweets

Task 1.3 corresponds with a multilabel problem where each sexist tweet can be marked with multiple labels representing different sexist behavior, those being: *objectification*, *ideological inequality*, *stereotyping dominance*, *sexual violence* and *misogyny non-sexual violence*. Again, for this task, we used the predictions from Task 1.1 obtained via the BERT models' fine-tuning.

To obtain both the English and Spanish prompts, we followed the same technique as in the previous tasks, with the configuration being: medium configuration, `max_labeled_demos` $= 6$ and `max_bootstrapped_demos` $= 1$. The main difference compared to the other tasks lays in how we scored the predictions for the training metric. Correctly guessing whether a label was present added 1 to the score, which was then normalized to the [0,1] range.

These prompts were optimized with modified output fields, as we configured 5 optional Pydantic outputs, one for each possible label. We used the same approach as in Task 1.2 to propagate the results, meaning that the model would not process a tweet predicted as non-sexist in Task 1.3 and that the final predictions were updated as it is shown in Equations 6 and 5.

To generate the labels from the predictions outputted by the model, followed the approaches presented in Sections 4.2 and 4.3, with the difference that each label has its associated hard and soft label.

5. Results

Tables 2, 3, and 4 show the results of each run for each of the target metrics in both the validation and test set. We use the official metric for EXIST 2025, the ICM metric [20], as well as its soft extension, ICM Soft.

Table 2 reveals a significant performance gap favoring BERT fine-tuning over prompt engineering for binary sexism detection. Run 1 (BERT) achieves ICM=0.506 vs. ICM=0.324/0.315 for DSPy approaches.

This performance gap contradicts findings from [12], suggesting that sexism detection may require domain-specific knowledge better captured through fine-tuning than prompting. The subjective nature of sexism judgment may necessitate parameter updates rather than instruction optimization.

Table 2
Task 1.1 results.

Run	Test				Development			
	ICM	ICM Norm	ICM Soft	ICM Soft Norm	ICM	ICM Norm	ICM Soft	ICM Soft Norm
1	0.506	0.754	0.739	0.618	0.584	0.792	0.766	0.624
2	0.324	0.663	-0.085	0.486	0.315	0.658	-0.033	0.495
3	0.315	0.658	-0.204	0.467	0.305	0.652	-0.031	0.495

The inclusion of Annotator Information (AnI) consistently improves DSPy performance, as evidenced by run 1 in both Tables 3 and 4, which performs worse than runs 2 and 3, demonstrating that perspective-aware modeling benefits prompt-based approaches. Finally, it is inconclusive whether the use of Retrieval-Augmented Generation (RAG) leads to performance gains, since runs 2 and 3 have comparable results across tasks, with no consistent advantage.
Note that the bad performance of run 1 in Task 1.2 and Task 1.3 with soft labels is due to the LLM generating the predictions without annotator information, meaning that the soft labels are truly hard labels. These runs were delivered for the soft label category for completeness.

Table 3
Task 1.2 results.

Run	Test				Development			
	ICM	ICM Norm	ICM Soft	ICM Soft Norm	ICM	ICM Norm	ICM Soft	ICM Soft Norm
1	-0.097	0.469	-8.518	0.000	0.029	0.509	-7.575	0.000
2	0.056	0.518	-3.501	0.218	0.141	0.544	-3.023	0.257
3	0.058	0.519	-4.006	0.174	0.134	0.542	-3.676	0.205

Table 4
Task 1.3 results.

Run	Test				Development			
	ICM	ICM Norm	ICM Soft	ICM Soft Norm	ICM	ICM Norm	ICM Soft	ICM Soft Norm
1	-0.157	0.464	-9.689	0.000	-0.239	0.447	-8.947	0.026
2	-0.182	0.458	-5.515	0.209	-0.139	0.469	-6.749	0.142
3	-0.036	0.492	-5.645	0.202	-0.004	0.499	-7.796	0.087

Table 5 shows the ranking we obtained for all tasks. Our best-performing approach (Run 1, Task 1.1 soft labels) achieved 9th place out of 67 submissions, demonstrating competitive performance for BERT-based methods. However, prompt engineering approaches ranked significantly lower (43rd-45th), highlighting the current limitations of pure prompting for sexism detection. Notably, incorporating annotator information improved DSPy rankings across Tasks 1.2 and 1.3, suggesting this strategy's potential for future prompt-based approaches.

6. Conclusion

In this work, we present CLiC's participation in the EXIST 2025 shared task for Tasks 1.1, 1.2, and 1.3. The main objective was to evaluate the viability of prompt engineering on its own. Our findings show that prompt-based methods alone fail to match the performance of standard techniques such as BERT

Table 5
Ranking across tasks and label types. Format: Rank/Number of Participants.

Task	Label	Run 1	Run 2	Run 3
Task 1.1	Soft	**9/67**	43/67	45/67
	Hard	50/160	120/160	121/160
Task 1.2	Soft	48/56	21/56	27/56
	Hard	82/140	68/140	66/140
Task 1.3	Soft	28/53	19/53	21/53
	Hard	49/132	52/132	42/132

fine-tuning for binary sexism text classification, Also, the performance on multilabel and multiclass tasks is not near the top of the rankings. We also observed that incorporating annotator information into prompt optimization leads to improved results. However, the effect of Retrieval-Augmented Generation (RAG) on performance remains inconclusive. Future work could explore the combined impact of model fine-tuning and prompt optimization for similar tasks, given that we were unable to pursue this due to resource constraints, as well as the application of these techniques to larger, more powerful LLMs.

Acknowledgments

This work has been possible as part of the FairTransNLP-Language project (PID2021-124361OB-C33), funded by MICIU/AEI/10.13039/501100011033/FEDER, UE. It has also been funded by the Generalitat de Catalunya (2024 PROD 00016 and 2021 SGR 00313 grants).

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers), 2019, pp. 4171–4186.

[2] L. Plaza, J. C. de Albornoz, I. Arcos, P. Rosso, D. Spina, E. Amigó, J. Gonzalo, R. Morante, Overview of exist 2025: Learning with disagreement for sexism identification and characterization in tweets, memes, and tiktok videos, in: Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Sixteenth International Conference of the CLEF Association (CLEF 2025). Jorge Carrillo-de-Albornoz, Julio Gonzalo, Laura Plaza, Alba García Seco de Herrera, Josiane Mothe, Florina Piroi, Paolo Rosso, Damiano Spina, Guglielmo Faggioli, Nicola Ferro (Eds.), 2025.

[3] L. Plaza, J. C. de Albornoz, I. Arcos, P. Rosso, D. Spina, E. Amigó, J. Gonzalo, R. Morante, Overview of exist 2025: Learning with disagreement for sexism identification and characterization in tweets, memes, and tiktok videos (extended overview), in: CLEF 2025 Working Notes. Guglielmo Faggioli, Nicola Ferro, Paolo Rosso, Damiano Spina (Eds.), 2025.

[4] O. Khattab, A. Singhvi, P. Maheshwari, Z. Zhang, K. Santhanam, S. Vardhamanan, S. Haq, A. Sharma, T. T. Joshi, H. Moazam, H. Miller, M. Zaharia, C. Potts, Dspy: Compiling declarative language model calls into self-improving pipelines, 2024.

[5] T.-M. Lin, Z.-Y. Xu, J.-Y. Zhou, L.-H. Lee, NYCU-NLP at EXALT 2024: Assembling large language models for cross-lingual emotion and trigger detection, in: Proceedings of the 14th Workshop on Computational Approaches to Subjectivity, Sentiment, & Social Media Analysis, Association for

Computational Linguistics, Bangkok, Thailand, 2024, pp. 505–510. URL: https://aclanthology.org/2024.wassa-1.50/. doi:10.18653/v1/2024.wassa-1.50.
[6] F. Rodríguez-Sánchez, J. Carrillo-de Albornoz, L. Plaza, J. Gonzalo, P. Rosso, M. Comet, T. Donoso, Overview of exist 2021: sexism identification in social networks, Procesamiento del Lenguaje Natural 67 (2021) 195–207.
[7] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, V. Stoyanov, Unsupervised cross-lingual representation learning at scale, arXiv preprint arXiv:1911.02116 (2019).
[8] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized bert pretraining approach, arXiv preprint arXiv:1907.11692 (2019).
[9] L. Plaza, J. Carrillo-de Albornoz, V. Ruiz, A. Maeso, B. Chulvi, P. Rosso, E. Amigó, J. Gonzalo, R. Morante, D. Spina, Overview of exist 2024—learning with disagreement for sexism identification and characterization in tweets and memes, in: International Conference of the Cross-Language Evaluation Forum for European Languages, Springer, 2024, pp. 93–117.
[10] E. Leonardelli, A. Uma, G. Abercrombie, D. Almanea, V. Basile, T. Fornaciari, B. Plank, V. Rieser, M. Poesio, Semeval-2023 task 11: Learning with disagreements (lewidi), arXiv preprint arXiv:2304.14803 (2023).
[11] M. Siino, I. Tinnirello, Prompt engineering for identifying sexism using gpt mistral 7b, Working Notes of CLEF (2024).
[12] D. Soylu, C. Potts, O. Khattab, Fine-tuning and prompt optimization: Two great steps that work better together, arXiv preprint arXiv:2407.10930 (2024).
[13] K. Opsahl-Ong, M. J. Ryan, J. Purtell, D. Broman, C. Potts, M. Zaharia, O. Khattab, Optimizing instructions and demonstrations for multi-stage language model programs, 2024. URL: https://arxiv.org/abs/2406.11695. arXiv:2406.11695.
[14] Q. Team, Qwen2.5: A party of foundation models, 2024. URL: https://qwenlm.github.io/blog/qwen2.5/.
[15] B. Warner, A. Chaffin, B. Clavié, O. Weller, O. Hallström, S. Taghadouini, A. Gallagher, R. Biswas, F. Ladhak, T. Aarsen, et al., Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference, arXiv preprint arXiv:2412.13663 (2024).
[16] A. G. Fandiño, J. A. Estapé, M. Pàmies, J. L. Palao, J. S. Ocampo, C. P. Carrino, C. A. Oller, C. R. Penagos, A. G. Agirre, M. Villegas, Maria: Spanish language models, Procesamiento del Lenguaje Natural 68 (2022). URL: https://upcommons.upc.edu/handle/2117/367156#.YyMTB4X9A-0.mendeley. doi:10.26342/2022-68-3.
[17] P. Pastells, W. S. Schmeisser-Nieto, S. Frenda, M. Taulé, Context-aware stereotype detection: Conversational thread analysis on bert-based models, in: SEPLN Posters, 2024.
[18] W. Wang, F. Wei, L. Dong, H. Bao, N. Yang, M. Zhou, Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers, 2020. URL: https://arxiv.org/abs/2002.10957. arXiv:2002.10957.
[19] Janus, Mysteries of mode collapse, 2022. URL: https://www.alignmentforum.org/posts/t9svvNPNmFf5Qa3TA/mysteries-of-mode-collapse, accessed: 2025-06-10.
[20] E. Amigo, A. Delgado, Evaluating extreme hierarchical multi-label classification, in: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2022, pp. 5809–5819.

A. Generated prompts

As an example, we present the optimized prompt for Task 1.2 in Spanish Listing 2. The rest of the prompt are available together with our code at https://github.com/clic-ub/EXIST_2025. We can see that it includes a reasoning field to generate a more complex thought process, and even though it is not the case, it could also include augmented examples. Given the great amount of training data we decided to

avoid this scenario. This prompt would get modified depending on the run; if no annotator information is needed, the associated fields would be removed, and if RAG is used, extra examples would be added to the *demos* section for each query. The optimized instructions, as well as the fields and examples, would be different for each specific task and language. Some fields that appear in the examples are not sent to the LLM, such as *hard_label* or *soft_label*. The fields that are stored along the prompt are simply determined by how the training set is formed. It is also possible that the examples for the few-shot are incorporated into the instructions. This was the case for the generated prompt for Task 1.3.

Listing 2: DSPy Spanish prompt for Task 1.2


```
{
  "predict": {
    "demos": [
      {
        "text": "No es que Awada sea una estúpida descerebrada (o sí). Pero le inventan
            notas donde la describen como tal porque es el modelo de mujer dócil y
            sumisa que los machirulos de derecha esperan para el resto.",
        "language": "Spanish",
        "category": "sexist",
        "labels_task1_2": "DIRECT",
        "hard_label": 1,
        "soft_label": 1.0,
        "annotator_gender": "female",
        "annotator_age": "18-22",
        "annotator_ethnicity": "Hispano or Latino",
        "annotator_studies": "Bachelor's degree",
        "annotator_country": "Chile"
      },
      {
        "text": "Tu mujer rebelde y locata contigo puede que se vaya pero te avisa y no
            te traiciona ",
        "language": "Spanish",
        "category": "sexist",
        "labels_task1_2": "DIRECT",
        "hard_label": 0,
        "soft_label": 0.33333333330000003,
        "annotator_gender": "female",
        "annotator_age": "46+",
        "annotator_ethnicity": "Hispano or Latino",
        "annotator_studies": "Bachelor's degree",
        "annotator_country": "Mexico"
      },
      {
        "text": "Alejandro Saavedra, violador parte de #laManada en SCZ finalmente
            sentenciado a 20 años #SeVaACaer",
        "language": "Spanish",
        "category": "sexist",
        "labels_task1_2": "REPORTED",
        "hard_label": 1,
        "soft_label": 0.6666666667000001,
        "annotator_gender": "female",
        "annotator_age": "46+",
        "annotator_ethnicity": "Hispano or Latino",
        "annotator_studies": "Master's degree",
        "annotator_country": "Venezuela"
      },
      {
        "text": " ANDA EN PRIMERA PUES COMO SABEMOS MUCHOS HOMBRES LAS MUJERES NO SABEN
```

```
            MANEJAR MUY BIEN LES CUESTA METER LA SEGUNDA Y POCO A POCO HAY QUE METER LA
            TERCERA Y DESDE AHÍ AGARRA INBION",
          "language": "Spanish",
          "category": "sexist",
          "labels_task1_2": "DIRECT",
          "hard_label": 1,
          "soft_label": 1.0,
          "annotator_gender": "female",
          "annotator_age": "46+",
          "annotator_ethnicity": "Hispano or Latino",
          "annotator_studies": "Bachelor's degree",
          "annotator_country": "Mexico"
        },
        {
          "text": " Pinche DT y administrador! Seguro ke no juegan ni canicas. Si no saben
              tratar bien a las mujeres NO DEBEN ASUMIR UNA DIRIGENCIA.",
          "language": "Spanish",
          "category": "sexist",
          "labels_task1_2": "REPORTED",
          "hard_label": 1,
          "soft_label": 0.8333333333,
          "annotator_gender": "female",
          "annotator_age": "18-22",
          "annotator_ethnicity": "White or Caucasian",
          "annotator_studies": "Bachelor's degree",
          "annotator_country": "Portugal"
        }
      ],
      "signature": {
        "instructions": "Dado el texto en español, proporciona una categoría que indique
            el tipo de sexismo presente (DIRECT, REPORTED, JUDGEMENTAL), una explicación
            detallada de por qué se clasifica así y un nivel de confianza en tu
            clasificación. Considera el contexto del texto y cualquier información demográ
            fica relevante proporcionada por el anotador, como su género, edad, etnia,
            estudios y país.",
        "fields": [
          {
            "prefix": "Text:",
            "description": "${text}"
          },
          {
            "prefix": "Language:",
            "description": "${language}"
          },
          {
            "prefix": "Annotator Gender:",
            "description": "${annotator_gender}"
          },
          {
            "prefix": "Annotator Age:",
            "description": "${annotator_age}"
          },
          {
            "prefix": "Annotator Ethnicity:",
            "description": "${annotator_ethnicity}"
          },
          {
            "prefix": "Annotator Studies:",
```

```
            "description": "${annotator_studies}"
          },
          {
            "prefix": "Annotator Country:",
            "description": "${annotator_country}"
          },
          {
            "prefix": "Reasoning: Let's think step by step in order to",
            "description": "${reasoning}"
          },
          {
            "prefix": "Category:",
            "description": "clase de sexismo"
          },
          {
            "prefix": "Confidence:",
            "description": "${confidence}"
          }
        ]
      },
      "lm": null
    }
}
```