Sexism Identification in Social Networks using LLMs

Notebook for the sEXism Identification in Social neTworks (EXIST) Lab at CLEF 2025

Leire Dominguez-Sol, Isabel Segura Bedmar

Human Language and Accessibility Technologies Group (HULAT), Computer Science and Engineering Department, Universidad Carlos III de Madrid, Leganés, 28911, Madrid, Spain

Abstract

This paper describes our participation in the EXIST 2025 shared task on sexism detection in social media. We developed a variety of systems for both Task 1.1 (binary classification of sexism) and Task 1.2 (fine-grained categorization), combining traditional machine learning models, Transformer-based architectures, ensemble methods, and hybrid CNN-BERT approaches. Our approach incorporates data augmentation, and multilingual modeling strategies to address challenges such as label disagreement and language variation. Results indicate that ensembles of fine-tuned models and hybrid architectures are especially effective in handling noisy annotations and capturing nuanced sexist content. This work highlights the importance of combining architectural diversity with robust preprocessing and evaluation strategies in sensitive NLP tasks.

Keywords

Sexism detection, NLP, Transformer models, Ensemble, Multilingual classification, EXIST 2025

1. Introduction

Sexism generally refers to prejudice, stereotyping, or discrimination, generally targeting women, based on sex or gender.. According to the European Institute for Gender Equality, it includes "any act, gesture, visual representation, spoken or written words, practice or behavior, based upon the idea that a person or a group is inferior because of their sex" [1]. It is based in beliefs about the fundamental nature and roles of men and women, usually ending in the view that one sex is superior or more valuable than the other. It can appear at multiple levels of society, from laws and institutions, to personal interactions, as well as internalized attitudes. It can also be expressed in many ways and forms, including speech, writing, and images, among others.

The consequences of sexism are deep and significant. It often leads to discrimination, marginalization, and unfair treatment towards women. The roots of sexism can be traced back to ancient civilizations, where social structures were built on rigid gender roles. Traditionally, men have been seen as strong leaders who participate in public life, while women have often been expected to focus on home and family, occupying more subordinate roles. Throughout the centuries, these divisions have become deeply embedded in cultural norms, legal systems, and even everyday language [2]. In addition to its direct consequences, sexism reinforces existing power hierarchies and cultural narratives that normalize inequality. Such dynamics produce cycles of disadvantage that are challenging to disrupt without deliberate and collective societal change. Addressing sexism, therefore, is not only a matter of individual behavior but a broader project of transforming institutions, cultural values, and social expectations.

While sexism has been transmitted through traditional media, education, and institutions over the years, the rise of digital platforms has given it new dimensions. In recent years, social media has become a powerful space where sexist attitudes can be amplified, normalized and even monetized. This digital environment facilitates the fast spread of sexist content, including stereotypes, hate speech, and harassment which often targets women and girls. Studies show that a significant proportion of young women (72,2%) are exposed to sexist remarks, body shaming, and unrealistic beauty standards on social platforms, leading to negative emotions such as frustration, anxiety, and social isolation [3].

Research also demonstrates that social media algorithms can increase the visibility of harmful content by promoting posts that generates strong emotional reactions [4]. This digital sexism not only mirrors

CLEF 2025 Working Notes, 9 – 12 September 2025, Madrid, Spain

existing inequalities but also creates new forms of psychological, social, and political harm.

Moreover, the persistence of sexist discourse in online spaces can have broader societal impacts. It affects the younger generations views on gender roles and relationships, shapes public discourse, and influences political decision-making. In this way, preventing sexism on social media is crucial for both individual protection and the creation of a more inclusive and equitable online community.

One of the primary issues in combating sexism on social media resides in its scale and quick expansion. Unlike conventional forms of discrimination, digital sexism can be pervasive, instantaneous, and impossible to monitor. It is often embedded in memes, comments and videos, which, along with the fact that content is often user-generated and rapidly shared, makes it harder to detect using conventional systems and even attempts to intervene [5].

In response to this growing concern, multiple initiatives have emerged from both the public or private sectors with the goal of both raising awareness and actively counteract gender bias. Among the most innovative ones are coding competitions and hackathons, which tackle the creativity and technical skills of diverse participants to propose tangible solutions for gender equality.

One notable example is the **womENcourage Hackathon** [6], which challenges participants to design and prototype technological solutions addressing issues like bias in language technology. The 2024 edition, themed "AI Fair Play Hackathon: Ensuring Equity in Language Technology," specifically targets the detection and mitigation of sexist stereotypes and unfair treatment in digital communication [6]. Another important competition is **EXIST (sEXism Identification in Social neTworks)** [7], organized within the IberLEF (Iberian Languages Evaluation Forum), whose challenge is to bring together experts in natural language processing (NLP), machine learning, and data science to develop automated systems capable of identifying sexist content in text from social platforms such as Twitter [8].

Beyond the technical aspect, these competitions also increase awareness of the relevance of ethical and social dimensions in artificial intelligence. They serve as platforms for advocacy, education, and empowerment. By engaging a broad audience, which include students, teachers, and technologists among others, they help build a shared understanding of the challenges posed by online sexism and the collective responsibility to develop fair, inclusive, and socially aware technological solutions.

In recent years, the EXIST shared task has established itself as a key benchmark for the automatic detection of sexism in social media texts. Since its first edition in 2021, the top-performing systems have predominantly relied on Transformer-based models often combined through ensemble strategies or enhanced with data augmentation techniques. In the 2024 edition, the best-performing system for the text-based task integrated multilingual Transformers with weighted ensembles, achieving state-of-the-art results through the fusion of different architectural variants and voting mechanisms [8].

Building upon these findings, the present work contributes to the EXIST 2025 [7] challenge by developing a comprehensive pipeline that combines fine-tuned Transformer models with linguistic and architectural enhancements. In particular, the project explores monolingual and multilingual models, ensemble techniques and LLM prompting. It also incorporates variants such as CNN-BERT hybrids and Retrieval-Augmented Classification (RAC), along with a detailed evaluation based on both overall and per-class performance.

The main contributions of this work include:

1. The implementation and evaluation of multiple Transformer-based architectures under the constraints of the EXIST 2025 setup
2. The integration of ensemble models and data augmentation techniques with the goal of improving robustness in ambiguous cases.

This paper also aims to provide a critical perspective on the social implications of automated sexism detection, especially in multilingual and cross-cultural contexts.

2. Methodology

This section presents the methodological framework designed to address the two classification tasks defined in the EXIST 2025 challenge. As outlined in previous sections, the detection and interpretation of sexism in tweets present a number of inherent challenges for both tasks: the annotations are provided by multiple annotators with potential disagreements; the linguistic expression of sexism can be subtle, sarcastic, or culturally specific; and the dataset includes tweets in two languages, which requires models capable of understanding both English and Spanish.

To address these challenges and maximize model performance, a wide range of modeling approaches, from traditional machine learning baselines to state-of-the-art transformers architectures was explored. First, standard classifiers trained on bag-of-words and TF-IDF representations to establish a classical baseline, were implemented. Focus was then shifted to fine-tuning a series of Transformer-based models, including both multilingual and monolingual models trained specifically for English or Spanish. These models form the core of the system. All models model were trained on both classification tasks of the EXIST 2025 challenge. For Task 1.1, each model was trained to predict whether a tweet is sexist. For Task 1.2, each model was trained to classify the intention behind the sexist content into three categories: **DIRECT**, **REPORTED**, or **JUDGEMENTAL**

In parallel, prompt-based learning with large language models (LLMs) was explored as well, testing zero-shot and few-shot strategies [9]. This allowed to assess the performance of foundation models without fine-tuning, and to compare them against supervised approaches under the same task definitions. Beyond purely textual approaches, it was also experimented with hybrid models, integrating BERT-style encoders with convolutional neural network (CNN) [10] components to assess whether injecting additional learned features could improve classification.

To further enhance performance and robustness, back translation [11] was applied as a data augmentation strategy, introducing lexical and syntactic variation through intermediate translation. This technique was integrated into the training pipeline of Transformer-based models to improve generalization across both languages.

Finally, ensemble strategies were explored, combining predictions from multiple Transformer models via majority voting and F1-weighted averaging. In parallel, we implemented a Retrieval-Augmented Classification (RAC) framework [12], where a Sentence-BERT model was used to retrieve the top-3 most similar training tweets, which were concatenated as contextual input to a base Transformer. This setup aimed to enhance the model's ability to understand nuanced or ambiguous tweets by providing relevant examples. RAC experiments were conducted on both monolingual and multilingual variants of the underlying LLMs.

The two classification tasks addressed in this work required slightly different modeling strategies. For Task 1.1, the objective was to predict whether a tweet contains or refers to sexist content. This was framed as a standard binary classification problem (YES / NO), and each multilingual model was fine-tuned accordingly. The label aggregation strategy used to define the ground truth 2.1 was kept consistent within each experiment and was applied to both the training and test data. All multilingual models were evaluated on both labeling schemes (majority vote or female-vote) to assess the impact of the annotation strategy on classification performance.

In contrast, Task 1.2 required a more sophisticated design. Two modeling approaches were explored:

1. **Two-step pipeline**: In this setup, a model was first trained on the binary labels of Task 1.1 to identify sexist tweets. Then, a separate model was trained to classify the intention (DIRECT, REPORTED, or JUDGEMENTAL) using only the subset of tweets labeled as YES. At inference time, Task 1.1 was applied first, and the stance classification model was applied only to those tweets predicted as sexist. This approach mirrors the task's conceptual structure but could compound errors from the first stage into the second.
2. **Unified multiclass classification**: Alternatively, a single model was trained to predict among four classes: the three stance categories plus a fourth NO class representing non-sexist content. This allowed the model to learn both the binary and fine-grained labels simultaneously, eliminating the need for a separate model or an explicit dependency on the output of Task 1.1. Label construction in this approach was adapted accordingly, applying the same aggregation strategies but assigning the NO label to tweets originally labeled as non-sexist in Task 1.1.

Both strategies were implemented for each model and for both label aggregation methods, enabling a comprehensive comparison across architectures and training paradigms.

The remainder of this section details each one of these approaches, explaining the motivations behind their selection, their implementation details, and how they fit into the overall system design, as well as two different approaches for obtaining the target label from the dataset.

2.1. Label Construction

One of the key challenges of this task lies in the definition of the target label, as we mentioned. To obtain the labels required for Task 1.1 and Task 1.2, we had to construct the label column ourselves. This introduces a critical design choice that directly affects the dataset composition and, consequently, model performance. We explored and compared two strategies for Task 1.1, and applied analogous logic to Task 1.2.

1. Strict Majority Voting: A tweet is labeled as sexist (YES) if it has at least four or more of the six annotators' labels set to YES. Otherwise, it is labeled as non-sexist (NO). In cases where there is no majority (i.e., a tie of 3 YES and 3 NO), the tweet is discarded and excluded from the dataset. This aggregation strategy reflects the approach officially used by the organizers for generating the hard ground truth labels in the test set, ensuring consistency between training and evaluation.
2. Filtered Aggregation: In this alternative approach, we implemented a more conservative aggregation strategy that leverages annotator metadata. Specifically, instead of discarding tweets with tied or ambiguous votes, we resolved these cases by assigning the label that was most frequently selected by the female annotators. This decision was motivated by the fact that the dataset includes the gender of each annotator, and the annotation setup always includes three women and three men per tweet. As a result, we could break ties systematically by prioritizing the majority vote among female annotators. As a result, this version retains all tweets, including those that would be discarded under strict majority voting, and therefore yields a larger training set.

For Task 1.2, which focuses on identifying the author's intention behind a sexist tweet, we applied a labeling strategy that mirrors the logic used for Task 1.1, while accounting for the specific structure of this subtask. Since Task 1.2 is only defined for tweets labeled as sexist (i.e., with label YES in Task 1.1), we first filtered the dataset accordingly.

To assign a unique intention label (DIRECT, REPORTED or JUDGEMENTAL), it was considered only the annotations provided by those annotators who had also labeled the tweet as sexist in Task 1.1. If a clear majority emerged among these annotators (i.e., one label appeared more frequently than any other), that label was assigned as the final Task 1.2 label. In cases where there was no majority, meaning a tie among the most frequent labels, the intention labels provided by the three female annotators were examined, restricted again to those who had voted YES in Task 1.1. If this subset exhibited a majority for one label, that label was assigned.

When neither of the previous steps yielded a conclusive result, a final fallback mechanism was applied: one of the most frequent valid labels among the Task 1.2, was randomly selected provided the annotators had voted YES in Task 1.1. This random choice was constrained to valid classes (excluding - and UNKNOWN). If no valid label was available at all, an edge case occurring when all labels were either missing or invalid, we assigned the label UNKNOWN as a last resort, which was later filtered out of our training set.

This multi-step resolution strategy ensures that we maximize the number of usable annotations for Task 1.2 without introducing arbitrary biases, while also making full use of the gender metadata provided by the dataset. It also ensures that annotation disagreements are resolved using a reproducible and interpretable process that leverages annotator metadata when necessary.

Both approaches were used during experimentation to understand their effect on model performance, dataset balance, and generalization. The choice of label construction has a direct impact on the class distribution and the number of training examples, which must be considered carefully when designing classifiers and evaluation pipelines.

Finally, it is worth noting that the dataset comprises tweets collected from Twitter, a platform characterized by informal language, abbreviations, emojis, and a strong reliance on cultural and social context. This makes the detection of sexism particularly challenging, as sexist content is often expressed in implicit or ironic ways, rather than through overtly offensive language. This motivates the use of context-aware models such as Transformers and Large Language Models (LLMs), which are capable of capturing nuanced patterns in language, even with limited surface-level cues.

2.2. Traditional Machine Learning Approaches

This section presents the traditional machine learning approach used as a baseline for the binary classification task. The method relies on transforming raw textual data into structured feature representations, followed by training well-established classifiers. The goal is to assess how these classical pipelines perform when applied to social media texts, before comparing them with modern deep learning methods.

To transform raw tweets into numerical feature representations, TF-IDF vectorization was applied. This model is an extension of the Bag of Words (BoW) model, where BoW represents each document as a vector in which each feature corresponds to a word from a vocabulary, and its associated value is the frequency of that word in the document. The vocabulary is built from a corpus of documents, where each unique word across the entire corpus defines a feature in the bag-of-words representation. TF-IDF scales term frequencies inversely by their document frequency, down-weighting frequent terms that are less informative and emphasizing rare but potentially discriminative ones. A detailed description of these text representation models can be found in Salton and Buckley (1988) [13]. Bigrams were included in addition to unigrams (individual words), allowing the models to capture short contextual expressions that are often relevant in social media texts. Three different text normalization strategies were compared in order to assess their impact on classification performance:

1. The basic variant involved lowercasing, accent normalization (e.g., á → a), stopword removal, using NLTK stopword lists for English and Spanish [14], and tokenization. User mentions and non-alphabetic characters were removed via regular expressions. No morphological reduction was applied in this version.
2. The lemmatized variant followed a similar cleaning procedure but applied lemmatization using spaCy tool to reduce each token to its base form. This strategy was intended to group inflected forms of the same word (e.g., "drives", "driving" → "drive"), potentially improving generalization.
3. The stemmed variant replaced lemmatization with stemming, using the Snowball stemmer implemented in the NLTK library for each language [14]. Stemming reduces words to their root forms using rule-based truncation (e.g., “driving”, “driven” → “driv”), which is computationally cheaper but less linguistically accurate [14].

As a baseline for the binary classification task, traditional machine learning algorithms were implemented using the previously described text representation pipelines. Two classifiers were selected for this purpose: Multinomial Naive Bayes (NB) [15] and Support Vector Machine (SVM) [16] with a linear kernel. Both models have been widely used in text classification tasks due to their efficiency and competitive performance, especially when combined with appropriate preprocessing and feature extraction methods [15, 16].

Naive Bayes classifiers rely on probabilistic reasoning under the assumption of feature independence. In the case of the Multinomial Naive Bayes model, word frequencies are treated as features drawn from a multinomial distribution, which is particularly suited for bag-of-words representations [17]. On the other hand, the Support Vector Machine classifier constructs a hyperplane in a high-dimensional space to separate classes, and has been shown to perform well in sparse feature spaces, such as those generated by text data [18].

Each preprocessing method was applied independently to the training and test sets, resulting in three parallel datasets. These were then paired with each classifier (NB or SVM), producing a total of six distinct classification pipelines.

The final architecture for each pipeline can be summarized as shown in Figure 1.

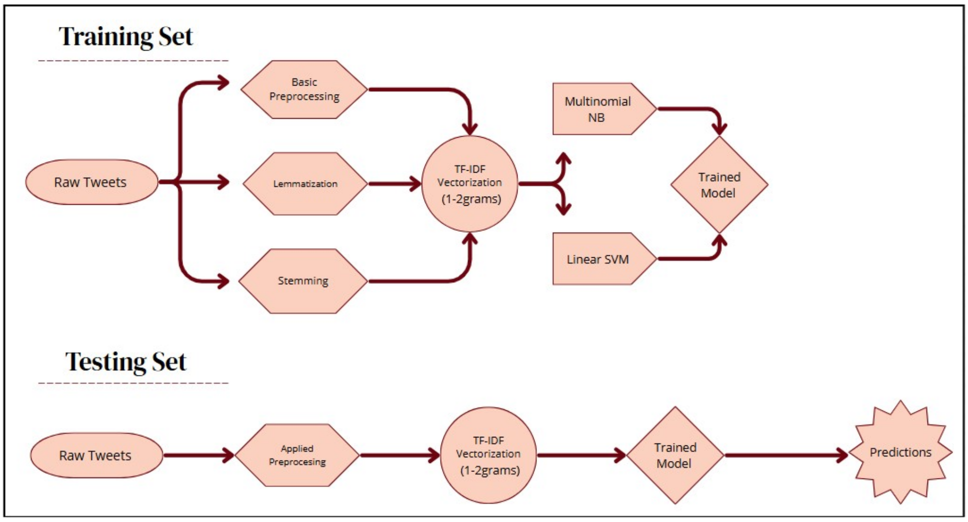


Figure 1: Overview of traditional machine learning pipelines evaluated in this work.

This baseline setup was designed not only to provide initial reference scores, but also to evaluate how different levels of linguistic normalization affect model performance in a controlled environment. These results later serve as a benchmark for more advanced architectures explored in subsequent sections.

2.3. Transformer-Based Models for Monolingual Datasets

Transformer-based architectures have become the foundation of modern NLP due to their ability to model contextual relationships between words using self-attention mechanisms [19]. Unlike recurrent or convolutional neural networks, Transformers allow for direct access to all positions in the input sequence simultaneously through self-attention mechanisms. This architecture enables efficient parallel training and has led to significant improvements in a wide range of natural language understanding tasks [19]. Although a more detailed overview of their theoretical foundations is provided in the State of the Art section, a brief introduction is included here to contextualize the models used in this work.

The original Transformer architecture, introduced by Vaswani et al. (2017) [19], consisted of both an encoder and a decoder, and was originally designed for machine translation tasks. Subsequent models have adapted this architecture for other NLP applications by using only one of the two modules. Encoder-only architectures, such as BERT [20], are commonly used for classification and understanding tasks, while decoder-only architectures, like GPT [21], are typically employed for text generation.

In the standard Transformer encoder, each token in the input sequence is represented by a contextualized embedding computed via multi-head self-attention and feed-forward layers. These representations are then used for downstream tasks through additional components, such as classification heads. In the case of text classification, a special token (typically [CLS]) is added to the input and its final representation is used as the aggregate embedding for classification.

Two monolingual Transformer models were selected for fine-tuning: BERT and RoBERTa. These models were pretrained exclusively on English or Spanish corpora and subsequently fine-tuned on task-specific data depending on the language of the tweet.

BERT (Bidirectional Encoder Representations from Transformers) was introduced by Devlin et al. in 2019 as one of the first language models to leverage deep bidirectional representations by jointly conditioning on both left and right context in all layers [20]. The model is built upon the encoder architecture of the Transformer and is pretrained using two self-supervised objectives:

- Masked Language Modeling (MLM): A subset of tokens (typically 15%) in the input is replaced with a special [MASK] token, and the model is trained to predict the original tokens based on the surrounding context [20].

- Next Sentence Prediction (NSP): Given a pair of sentences, the model is trained to predict whether the second sentence logically follows the first. This objective helps the model capture inter-sentence relationships [20].

In this work, the base version of BERT ***(BERT-Base)*** was employed, which consists of 12 Transformer encoder layers, 12 self-attention heads per layer, and a hidden size of 768, amounting to approximately 110 million parameters. The English model used was *bert-base-uncased* [22], pretrained on BookCorpus and English Wikipedia, comprising over 3.3 billion words.

RoBERTa (Robustly Optimized BERT Approach) is a variant of BERT introduced by Facebook AI in 2019 with the aim of improving the pretraining process of Transformer-based language models [23]. While maintaining the original BERT architecture, RoBERTa introduces a number of key modifications that enhance model performance across a wide range of NLP tasks.

One of the main differences lies in the removal of the Next Sentence Prediction (NSP) objective. In BERT, NSP was used during pretraining to help the model learn sentence-level coherence, but later studies showed that removing it can lead to improved performance in downstream tasks. RoBERTa discards NSP and focuses entirely on Masked Language Modeling (MLM), which enables the model to better capture token-level dependencies.

Another important modification is the use of dynamic masking. While BERT applies masking once during data preprocessing, RoBERTa regenerates mask patterns at each epoch. This introduces more variability and forces the model to learn contextual representations more robustly. In addition, RoBERTa was trained on significantly more data (over 160 GB of text) from multiple sources such as BookCorpus, English Wikipedia, CC-News, OpenWebText, and Stories [23]. This extended corpus, combined with longer training schedules, larger batch sizes, and higher learning rates, results in a more powerful and generalizable language model. In this work, the pretrained ***roberta-base-bne*** [24] was used for Spanish tweets, which comprises around 125 million parameters.

Both models were retrieved from the Hugging Face Model Hub and fine-tuned independently on their respective language subsets and later merged to create the final output for evaluation.

In both cases, predictions were generated independently for English and Spanish, and then merged during inference to construct unified outputs for evaluation. This strategy enabled the models to specialize in the linguistic nuances of each language while still contributing to the overall multilingual classification pipeline.

This monolingual modeling approach allowed the classifiers to benefit from language-specific pretraining, which can be especially advantageous for handling subtle or idiomatic expressions of sexism that vary across linguistic and cultural contexts.

2.4. Transformer-Based Models for Multilingual Datasets

Multilingual versions of transformer models are pretrained on corpora covering many languages, making them capable of handling cross-lingual input without requiring explicit translation or separate models. This is particularly advantageous in the context of EXIST 2025, where the dataset consists of tweets in both English and Spanish.

For this reason, several multilingual Transformer models were fine-tuned to address both subtasks. All models were fine-tuned on the whole dataset (containing tweets from both languages) and trained using a consistent setup, including identical training procedures and evaluation metrics. Tokenization was performed using the model-specific tokenizer, and classification was implemented via a linear layer applied to the [CLS] representation, followed by a task-appropriate activation function.

The multilingual models evaluated in this work include bert-base-multilingual-cased (Multilingual BERT) [25], mdeberta-v3-base (Multilingual DeBERTa) [26], distilbert-base-multilingual-cased (Multilingual DistilBERT) [27], and xlm-roberta-base (XLM-RoBERTa) [28]. In addition, the monolingual English model roberta-base [29] was included in the experiments to assess its performance on multilingual data.

Each of these models will be described, focusing on their specific architecture, pretraining objectives, and adaptation to the EXIST 2025 tasks.

The general architecture and training objectives of BERT were already introduced in the monolingual Section 2.3 for English tweets. In this section, the focus shifts to its multilingual counterpart, **bert-base-multilingual-cased**, which was employed to jointly model both English and Spanish texts.

This model, available on the Hugging Face Model Hub [25], was pretrained on the concatenation of Wikipedia dumps from the 104 largest languages using a cased WordPiece vocabulary. Its multilingual pretraining makes it capable of handling inputs in both English and Spanish without the need for language-specific customization. Unicode support and case preservation further enhance its applicability to social media content, where capitalized words and named entities often carry important semantic information.

The architectural details and pretraining objectives of RoBERTa were already introduced in the monolingual section 2.3, where a Spanish-adapted variant, **roberta-base-bne**, was fine-tuned on Spanish tweets. In addition to this monolingual model, the general-purpose English model **roberta-base** from Facebook AI [29] was also evaluated in the multilingual track to assess its robustness when fine-tuned on bilingual data. This model shares the same architecture as the Spanish variant, but was pretrained on English-only corpora.

The motivation behind this choice was to investigate whether a high-capacity model trained exclusively on English data could generalize effectively when exposed to a mixed-language dataset without any additional cross-lingual adaptation. While this model lacks native multilingual support, it was included to explore its ability to handle bilingual social media content implicitly through fine-tuning on the combined English–Spanish dataset.

mDeBERTa (Multilingual Decoding-enhanced BERT with disentangled attention) is a multilingual extension of DeBERTa, a Transformer-based model architecture introduced by Microsoft with the goal of improving contextual representation by modifying both attention mechanisms and positional encoding strategies. The original DeBERTa architecture separates the representation of content and positional information in the attention layers and includes an enhanced mask decoder, resulting in better performance on several benchmarks compared to BERT and RoBERTa [30].

Two main innovations characterize the DeBERTa architecture. First, the disentangled attention mechanism separates the representation of token content and position across different vector subspaces. Unlike traditional Transformer models, where positional and semantic information are combined within the same vector, DeBERTa encodes them independently, allowing the model to better capture structural and contextual relationships. Second, an Enhanced Mask Decoder (EMD) is used during pretraining, which improves the model's ability to recover masked tokens, resulting in more informative internal representations.

The multilingual version, mdeberta-v3-base, builds upon the improvements of DeBERTa v3 and extends them to a cross-lingual setting. Unlike previous multilingual models that rely on shared vocabularies and token-level transfer, mDeBERTa leverages a disentangled attention mechanism, which allows it to represent word meaning and position more flexibly, leading to improved performance across a range of languages.

In this project, the version **mdeberta-v3-base** from Microsoft was used, retrieved via Hugging Face [26]. It consists of 12 layers, 768 hidden dimensions, and 12 attention heads, and was fine-tuned on the combined English–Spanish dataset provided by the EXIST 2025 competition. Tokenization was performed using the default tokenizer for mDeBERTa, which uses SentencePiece with a vocabulary of 250,000 subwords.

Due to its architectural innovations and multilingual capabilities, mDeBERTa serves as a strong candidate for addressing complex tasks such as sexist language detection in heterogeneous social media content.

DistilBERT is a lightweight and faster version of the original BERT model, developed through a process known as knowledge distillation. It was introduced by Sanh et al. in 2019 with the goal of reducing the size and computational cost of BERT, while maintaining most of its performance on standard NLP benchmarks [31]. The model is designed to serve as a compact student network that learns to approximate the output distributions of a larger, pretrained teacher model, in this case, BERT.

Architecturally, DistilBERT follows the same encoder-based Transformer design as BERT, but with

Table 1
Summary of Transformer-based models evaluated in the multilingual setting

Model Name	Language	Type
bert-base-multilingual-cased	Multilingual	BERT
mdeberta-v3-base	Multilingual	DeBERTa
distilbert-base-multilingual-cased	Multilingual	DistilBERT
xlm-roberta-base	Multilingual	RoBERTa
roberta-base	English	RoBERTa

a reduced number of layers and simplified training objectives. Specifically, it reduces the number of Transformer layers from 12 to 6, resulting in a model that is roughly 40% smaller and 60% faster in inference time, with approximately 66 million parameters. Despite this reduction, it preserves the same embedding dimension (768) and the same number of attention heads (12) as BERT-base. In addition, regularization techniques such as dropout and input noise were incorporated during training to improve generalization and stability.

The model was pretrained using a triple loss function combining distillation loss, masked language modeling (MLM) loss, and cosine embedding loss, which together enable the student model to mimic the hidden states and output predictions of its teacher [31].

In this work, the **distilbert-base-multilingual-cased** model from Hugging Face [27] was used. Unlike the original DistilBERT, which was pretrained on English-only corpora, this multilingual version was trained on data from 104 languages and tokenization was carried out using the default WordPiece tokenizer associated with the model.

XLM-RoBERTa is a multilingual extension of the RoBERTa architecture, introduced by Facebook AI in 2020 as part of their efforts to improve cross-lingual language modeling [32]. It inherits the same Transformer-based encoder design and masked language modeling (MLM) objective from RoBERTa, but is pretrained on a significantly broader linguistic corpus. While RoBERTa was trained exclusively on English text, XLM-RoBERTa was trained on 100 languages using data from CommonCrawl, totaling 2.5 terabytes of filtered text.

Its tokenizer is based on SentencePiece, which allows the model to operate on byte-level subwords, making it suitable for multilingual processing without requiring language-specific preprocessing or vocabularies. This enables robust cross-lingual transfer and zero-shot learning.

The variant used in this work is **xlm-roberta-base**, which consists of 12 layers, 768 hidden dimensions, and 12 attention heads, summing up to approximately 270 million parameters. The model was obtained from Hugging Face [28] and fine-tuned on the EXIST 2025 dataset.

XLM-RoBERTa serves as one of the most powerful pre-trained multilingual models available, demonstrating competitive performance in a variety of multilingual benchmarks such as XNLI and MLQA. In this context, it was included to assess whether a model explicitly trained on multilingual corpora outperforms other alternatives in detecting sexist content across both Spanish and English tweets.

In sum, we use the following transformers to address the multilingual setting of both tasks.

All models were trained and adapted to each task by simply adjusting the number of output classes. The training pipeline used identical hyperparameters, including a linear learning rate scheduler, early stopping, and evaluation based on validation loss. Table 2 summarizes the key training configurations used in this work.

2.5. Hybrid CNN-BERT Architectures

To investigate whether convolutional representations can complement transformer-based embeddings in the task of sexism detection, a hybrid architecture combining CNNs with pre-trained BERT-style tokenizers was implemented. This approach seeks to leverage the local pattern extraction capability of CNNs alongside the rich contextual embeddings provided by Transformer tokenizers.

Table 2
Training hyperparameters used for multilingual Transformer models.

Hyperparameter	Value
Number of epochs	5
Batch size (train / eval)	8 / 8
Learning rate	2e-5
Scheduler	Linear
Warmup ratio	0.1
Evaluation steps	100
Save strategy	Every 100 steps
Metric for best model	Validation loss
Early stopping	Yes (patience = 3)
Seed	400

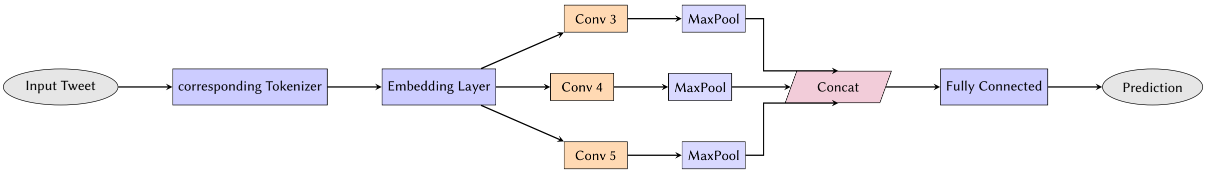


Figure 2: Architecture of the hybrid model: Transformer tokenizer + TextCNN.

Rather than using the contextual embeddings generated by the full transformer encoder, the proposed architecture integrates only the corresponding tokenization and vocabulary structure of the pre-trained models. The tokens were first converted into integer input_ids using pre-trained tokenizers such as ***bert-base-uncased***, ***bert-base-multilingual-cased***,***roberta-base***, and ***distilbert-base-uncased***. These token IDs were then passed to a CNN model, without using the transformer encoders themselves, enabling a lighter-weight architecture while still preserving the semantic structure captured by the tokenizer.

The CNN model adopted in this work is based on the classical The CNN model adopted in this work is based on the classical TextCNN architecture introduced by Kim (2014) [10]. It includes an embedding layer followed by parallel convolutional layers with kernel sizes of 3, 4, and 5, each capturing n-gram features of different lengths. These feature maps are max-pooled and concatenated before being passed through a dropout layer and a fully connected classification head. This design allows the model to learn multiple local patterns that are potentially relevant to the task, such as discriminatory phrases or idiomatic expressions.

This hybrid CNN-BERT setup offers a lightweight yet competitive alternative to full fine-tuning of large transformer models, enabling faster training while still benefiting from pre-trained linguistic knowledge embedded in the tokenizer vocabularies. The final architecture implemented in this hybrid setup is summarized in Figure 2, where the input tweets are first tokenized using pre-trained Transformer tokenizers and subsequently processed by a convolutional neural network with multiple kernel sizes and max-pooling layers.

2.6. LLM Prompting

Recent advances in large-scale pretraining have enabled the development of LLMs, which are transformer-based architectures trained on massive text corpora to learn general-purpose language representations [9]. Unlike traditional models that require task-specific fine-tuning, LLMs such as GPT-3 [9], GPT-4 [33], LLaMA [34], and PaLM [35] are capable of performing a wide range of downstream tasks, such as classification, question answering, summarization, or sentiment analysis, by conditioning on natural language instructions alone, a paradigm known as prompt-based learning or in-context learning.

Table 3
Example of zero-shot prompt in Task 1.1

Prompt:
Classify the following tweet as sexist (YES) or not sexist (NO):
"I guess she got the job because she’s cute."

Table 4
Example of few-shot prompt in Task 1.1

Input: Given a tweet, classify it as either YES (sexist) or NO (non-sexist).
Example 1: "She got the job just because she’s a woman." → YES
Example 2: "Everyone deserves equal rights regardless of gender." → NO
Test Tweet: "Maybe she’d understand it if she stopped being so emotional." →

These LLMs are pretrained using the causal language modeling objective, which predicts the next token given a preceding context. This simple objective, when scaled up to billions of parameters and trained on hundreds of billions of tokens, results in emergent capabilities such as instruction-following, multilinguality, reasoning, and even zero-shot or few-shot generalization. Importantly, these capabilities can be obtained without updating the model weights, simply by providing a well-crafted input prompt.

In this setting, the model is given a task description and, optionally, a few examples in natural language, and it generates an output that satisfies the task requirement (see example in Table 3 and Table 4). This framework is highly appealing in practical scenarios because it allows for task adaptation without additional fine-tuning, making it especially suitable when labeled data is scarce, or when inference is needed across multiple tasks or domains [9]. In this work, the following prompting strategies have been explored:

1. Zero-shot prompting, where the model receives only an instruction or a question describing the task (see Table 3). In this setup, the model is asked to classify a tweet as sexist (YES) or not sexist (NO) using only a natural language prompt, without access to labeled data or examples. This evaluates the model’s ability to generalize from its pretrained knowledge. Outputs were post-processed to normalize variations like `yes`, `Sí`, or NOPE into canonical labels.
2. Few-shot prompting, where the model is provided with a natural language instruction along with a small set of manually curated tweet–label pairs to illustrate the task, typically 3–5 examples (see table 4). This strategy, popularized by GPT-3 [9], enables the model to infer the desired output format by analogy. The examples were selected to reflect diverse linguistic patterns and label contexts. The same decoding setup as in the zero-shot configuration was used, and outputs were post-processed to standardize the label format.

The goal of evaluating these methods in the EXIST 2025 challenge was to understand whether LLMs can classify sexist content in tweets without task-specific fine-tuning, and how their performance compares to supervised models trained with domain-specific labels.

To carry out the experiments in all prompting strategies, we relied on several publicly available large language models from the Hugging Face Model Hub. These include encoder-only models fine-tuned for Natural Language Inference (NLI), such as vicgalle/xlm-roberta-large-xnli-anli [36], joeddav/xlm-roberta-large-xnli [37], MoritzLaurer/mDeBERTa-v3-base-mnli-xnli [38], and cross-encoder/nli-deberta-v3-large [39], as well as the generative model google/flan-t5-large [40]. These models were selected to cover both classification-oriented and generative prompting paradigms.

The first group of models is based on pretrained transformers (such as RoBERTa, XLM-RoBERTa, DeBERTa, and mDeBERTa) that were fine-tuned for NLI tasks like MNLI or XNLI. In zero-shot settings, these models can be applied to new classification tasks by reframing the input as an NLI hypothesis-premise pair and selecting the most probable label. Since their architectures are encoder-only and operate on fixed input-output label sets, they were not compatible with few-shot prompting.

In contrast, flan-t5-large is a 770M-parameter encoder–decoder model developed by Google as part of the FLAN project [41]. It was instruction-tuned on a large collection of diverse NLP tasks using textual prompts and natural language feedback. Unlike traditional fine-tuning that adapts models to a specific dataset, instruction tuning guides the model to follow task instructions expressed in natural language. This strategy has shown to significantly enhance the zero-shot and few-shot generalization abilities of LLMs.

In summary, prompting offers a flexible and lightweight approach to adapting LLMs to classification tasks and provides a valuable alternative when labeled data is scarce or quick prototyping is required.

2.7. Transformer Ensembles

Transformer ensemble models are a well-established technique in machine learning to improve the robustness, stability, and predictive accuracy of a system by combining the outputs of multiple individual models. Rather than relying on a single model, ensembles aggregate predictions from a diverse set of architectures or training runs, mitigating individual biases and taking advantage of complementary strengths [42].

In this project, ensemble methods were applied to the multilingual Transformer models described previously (see section 2.4. The goal was to leverage the different generalization patterns and inductive biases of architectures such as BERT, RoBERTa, mDeBERTa, DistilBERT, and XLM-RoBERTa to create a unified output that is more reliable than any of the individual models alone.

Each of the five multilingual models was fine-tuned independently on the combined English–Spanish dataset for both tasks. During inference, the predictions produced by each model were collected and combined using two ensemble strategies:

1. **Majority Voting:** For each tweet, the predicted labels from all five multilingual models were collected and aggregated. In Task 1.1, the number of models is odd, and only two possible labels ("YES" or "NO") exist. As a result, a unique majority always emerges, making tie-breaking unnecessary.
   However, in Task 1.2, the set of possible labels includes "DIRECT", "REPORTED", "JUDGEMENTAL", and "NO", which makes ties more likely. In these cases, the following rule was adopted:
   - If a single label receives more votes than any other, it is selected as the final prediction.
   - In case of a tie between multiple labels, the final label is randomly chosen among the tied candidates. However, if one of the tied labels is NO and at least one of the others is not, NO is excluded from the random selection to favor the detection of sexist content.

   This heuristic prioritizes detecting sexist content over misclassifying it as non-sexist, aligning with the task's sensitivity to false negatives in real-world applications.
2. **Weighted ICM Voting:** In this strategy, each model's prediction was weighted according to its validation ICM-score. The final prediction for each tweet was the class with the highest total weighted score across all model outputs. This method gives more influence to models that have demonstrated stronger performance during training, particularly useful when certain models are more reliable under specific linguistic patterns or data conditions.

The ensemble architecture consists of two main stages. First, each input tweet is passed independently through all five multilingual models. Each model returns a label prediction, which is then aggregated by a decision module. Depending on the selected strategy, this module either computes the most frequently predicted label (majority vote) or calculates a weighted score for each label using the ICM-score weights (weighted voting). The final decision is the label with the highest aggregated support. Figure 3 illustrates the architecture followed by an ensemble with majority vote.

By combining multiple models in this way, the ensemble approach aims to reduce variance and capture diverse perspectives embedded in different pretraining corpora and architectures. This is particularly beneficial in challenging classification tasks such as sexism detection in social media, where subtle linguistic cues and cross-lingual variability require nuanced interpretation.

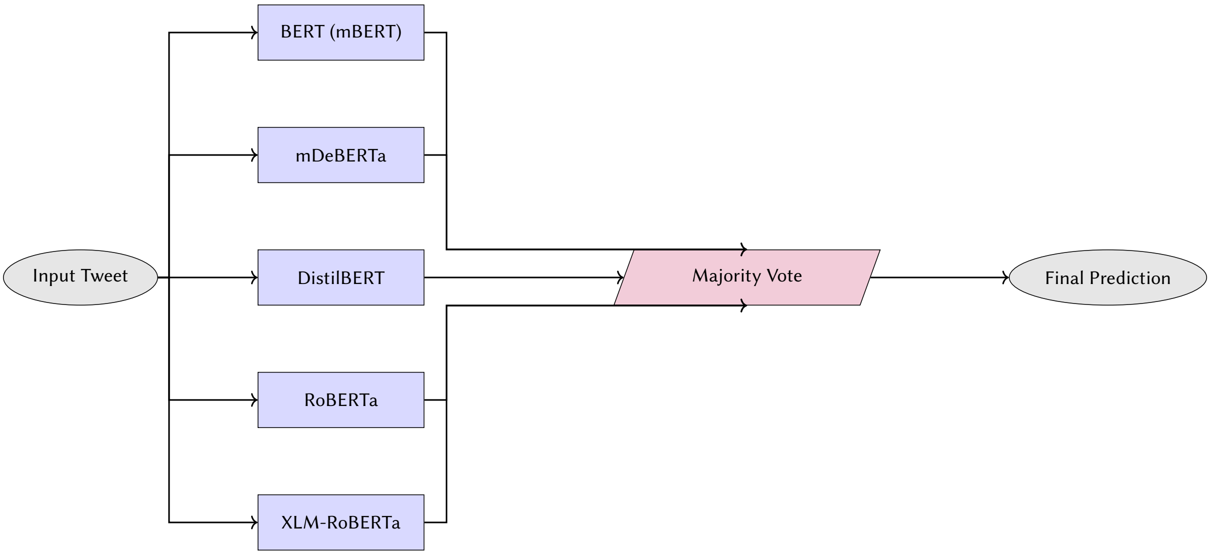


Figure 3: Ensemble of Transformer models using majority voting.

2.8. Retrieval-Augmented Classification (RAC) with Fine-Tuning

In addition to traditional fine-tuning and prompting strategies, this work explores a Retrieval-Augmented Classification (RAC) approach, inspired by the Retrieval-Augmented Generation (RAG) framework introduced by Lewis et al. (2020) in the context of open-domain question answering and knowledge-intensive tasks [43]. While RAG was originally designed for generation, its core idea, which is injecting relevant retrieved documents into the model's input, has inspired adaptations to classification tasks as well [44].

RAC is a hybrid modeling paradigm that combines neural language models with a non-parametric retrieval mechanism to improve performance on downstream tasks, particularly in low-data or linguistically complex settings. Unlike standard prompting methods that rely on a fixed template or a small set of manually selected examples, RAC dynamically retrieves semantically similar examples from a **support set**, that is, a predefined corpus consisting of labeled training examples. Each instance in this corpus is composed of a tweet and its corresponding ground-truth label.

These retrieved examples are incorporated into the input both during training and inference, providing additional context that is specific to each input tweet. This allows the model to condition its prediction on relevant past examples, effectively adapting the decision boundary in a local and context-aware manner. In this work, the RAC framework was applied both during training and inference. The models were fine-tuned on inputs enriched with retrieved examples, allowing them to learn from semantically similar instances throughout the training process. This approach enables the model to incorporate retrieved context into its reasoning not only at inference time, but also during the optimization of model parameters. This setup enables the model to leverage the diversity of the training corpus by fine-tuning on inputs enriched with semantically similar examples, allowing the model to adapt its internal parameters based on the retrieved context.

The main motivation for using RAC lies in its ability to enhance classification performance in low-resource or ambiguous settings by injecting external examples that mirror the semantics of the input. Prior work has shown that this strategy can improve generalization, particularly when the test data contains nuanced patterns that benefit from analogical reasoning or contextual reinforcement [45].

To implement the RAC framework, a two-stage pipeline was implemented. The first stage involves retrieving contextually relevant examples for each tweet, and the second stage augments the tweet with this retrieved information before feeding it to the classifier.

For the retrieval step, a Sentence-BERT model, ***paraphrase-multilingual-MiniLM-L12-v2*** [46], was used to encode the training tweets into dense vector representations. This model supports over 50 languages and is based on a distilled version of the Transformer architecture, optimized for cross-lingual sentence similarity tasks. These embeddings were normalized and indexed using FAISS with inner product similarity, equivalent to cosine similarity on normalized vectors. For each tweet in the training and test sets, the top-k most similar examples were retrieved from the training corpus. In order to prevent unintended information leakage during training, self-retrieval was prevented by excluding the query tweet from its own results.

This structure ensures that the classifier receives both the original tweet and relevant auxiliary content that may guide its decision. The complete pipeline was implemented using PyTorch and Hugging Face Transformers, allowing seamless integration with existing training and evaluation code.

By choice, the top three retrieved results (k =3) were used as contextual information. However, the retrieval setup is flexible and can be easily extended to incorporate a larger number of neighbors if needed. The same retrieval and context injection procedure was applied symmetrically to both training and test sets to ensure consistency and avoid domain shift between training and inference.

This setup allows the downstream classifier to make predictions not only based on the original tweet, but also informed by semantically related examples from the training distribution. Unlike earlier stages of this project, where static, hand-crafted contexts were used, this approach leverages real semantic similarity computed from large-scale data, aligning more closely with the principles of retrieval-augmented learning and enabling more informed and context-aware predictions.

Once the retrieval step had enriched each input tweet with semantically related content, the resulting inputs were integrated directly into a Transformer-based sequence classification model, such as XLM-RoBERTa, which was then fine-tuned on these enriched inputs. Specifically, each training and test instance consisted of the original tweet followed by a context string containing the top-k retrieved neighbors, structured as:

```
[TWEET]: {original tweet} [CONTEXT]: {retrieved tweet 1}. {retrieved tweet
2}. {retrieved tweet 3}
```

This format was treated as a single input sequence and tokenized using the pretrained tokenizer corresponding to the base model, in the context of this project, the five multilingual models 2.4. The tokenized inputs were then fed into a Transformer encoder with a classification head, following the same fine-tuning procedure described in earlier sections.

However, no architectural changes were made to the base classifier. The only modification was the inclusion of retrieved context within the input string. This decision was based on the design philosophy of retrieval-augmented classification (RAC): enhancing model reasoning through external information without modifying its internal structure [47, 12]. This seamless integration into the pipeline provided two main benefits:

1. It enabled the model to directly leverage semantically similar examples from the training tweets, both during training and inference, without requiring external knowledge bases. The retrieval step was always performed using the available training data.
2. It was fully compatible with the existing fine-tuning framework in Hugging Face Transformers, as the retrieval-augmented inputs could be processed and tokenized in the same way as standard text classification inputs, without introducing architectural changes to the model.

2.9. Data Augmentation

Data augmentation is a well-established technique in Natural Language Processing (NLP) aimed at improving model generalization and robustness, especially in scenarios with limited annotated data or class imbalance. The goal is to artificially expand the training dataset with additional examples that preserve the underlying semantics but vary in form, thus reducing overfitting and enhancing performance on unseen inputs [48].

Among the various augmentation strategies, back translation has proven to be particularly effective for text classification tasks [49], as it introduces lexical and syntactic diversity while preserving the original semantic intent of the input text. It involves translating a text into an intermediate language and then translating it back to the original language. This process generates paraphrased versions of the input while maintaining its core meaning. In this work, back translation was applied bidirectionally for both English and Spanish tweets. The intermediate language selected was French, based on its wide support in machine translation models and its linguistic distance from both source languages.

To implement this, a two-stage translation pipeline was built using pretrained ***MarianMT*** models from the Hugging Face Model Hub [50]. For English tweets, the pipeline involved translation from English to French (en → fr) followed by French to English (fr → en). For Spanish tweets, a similar pipeline was constructed using es → fr followed by fr → es. The process was batch-optimized using PyTorch and executed on GPU for efficiency.

Language detection was performed using the *langdetect* library to route each tweet to the appropriate translation path. The architecture ensured that only tweets in English or Spanish were processed, and unsupported languages or malformed entries were skipped or defaulted to their original version.

To augment the training data, the entire training set was passed through this pipeline, generating one augmented instance for each original example. The resulting paraphrased tweets were then merged with the original training set, effectively doubling its size. This process aimed to introduce lexical and syntactic variation, improving the model's ability to generalize across diverse formulations of sexist or non-sexist content. It is important to establish that only the training set was augmented to prevent data leakage or evaluation bias.

This strategy aimed to increase linguistic diversity in the training data, particularly in expressions of sexism that may vary subtly across phrasing. By exposing the model to semantically equivalent yet lexically distinct examples, the goal was to improve robustness and reduce overfitting to specific lexical patterns.

3. Experiments and Results

This section presents the evaluation framework and results obtained for the models developed in this work, following the official guidelines of the EXIST 2025 lab. Our analysis focuses on the subtasks addressed in this study: Task 1.1 (binary classification) and Task 1.2 (multi-class classification), both applied to tweets. These tasks aim to detect sexist content and identify the author's intention, respectively.

To ensure the consistency and fairness of the evaluation process, all models were assessed using the official metric defined by the organizers of EXIST 2025: the Information Contrast Measure (ICM). This measure, introduced by Amigó and Delgado (2022) [51], was specifically designed to address some of the limitations of classical evaluation metrics in complex classification scenarios such as those involving semantic hierarchies, label imbalance, or subjective categories, all of which are inherent to the problem of sexism detection.

ICM is grounded in information theory and builds upon the notion of Pointwise Mutual Information (PMI), which quantifies the association between two discrete variables by measuring the divergence of their co-occurrence from statistical independence. While PMI has traditionally been used in unsupervised tasks, ICM adapts this concept to the supervised classification setting, where the goal is to compare predicted labels with gold standard annotations.

More formally, ICM quantifies the informativeness of a system's predictions by contrasting the information content conveyed by the predicted labels against a baseline of random predictions. In doing so, it not only captures whether the prediction is correct, but also how informative and meaningful the prediction is with respect to the underlying class distribution. This makes it particularly appropriate for tasks where misclassifications may differ in severity depending on the semantic distance between the predicted and true labels.

Furthermore, the normalized version of ICM (ICM-Norm) scales the resulting value between 0 and 1, enhancing its interpretability and allowing fair comparisons across different systems. A score of 1 indicates optimal informativeness, whereas a score close to 0 reflects low informativeness, comparable to a random or uninformative classifier.

Given these properties, ICM offers two key advantages over standard metrics such as accuracy or F1-score:

1. It accounts for the semantic structure of the label space, penalizing severe misclassifications more than minor ones.
2. It provides a quantitative estimate of informational value, rather than treating all correct predictions equally.

The evaluation of submitted runs supports two formats:

- Hard-hard evaluation: both system outputs and gold annotations are converted into hard labels. This means that only one definitive label is assigned per instance. For example, in Task 1.1, a tweet is either "YES" or "NO". The hard label is derived from majority vote among annotators.
- Soft-soft evaluation: both the model predictions and the gold annotations are expressed as probability distributions. This is especially relevant in Learning with Disagreement (LeWiDi) scenarios, where annotators may disagree, and the ground truth is represented as a distribution over labels.

These metrics evaluate the similarity between the predicted and true label sets, accounting for hierarchy, probability mass, and class severity. In addition, traditional metrics such as the F1-score are reported for interpretability, although they are not optimal for the hierarchical structure of the task.

Task 1.2 poses an additional challenge due to its hierarchical class structure. Predictions must distinguish not only between "NO" and "YES", but also among the three sexist intentions: **DIRECT**, **REPORTED**, and **JUDGEMENTAL**.

ICM naturally accommodates this hierarchy by penalizing misclassifications between YES and NO more strongly than between, for instance, REPORTED and DIRECT.

In this project, both ICM and its normalized variant (ICM-Norm) were computed using the official evaluation library PyEvALL, as recommended by the competition guidelines. These metrics serve as the primary basis for comparing model performance across all experimental conditions [7].

The gold labels were constructed using majority voting among annotators, and only tweets with a clear majority were retained for training and evaluation. Tweets with ambiguous or tied votes were excluded to ensure robust training signals.

3.1. Results for Task 1.1 - Binary Classification

All models in this section were trained and evaluated using the hard-hard evaluation mode, with ICM as the official performance metric. The decision to use the hard-hard evaluation mode, rather than soft-soft, was motivated by two key considerations. First, the hard-hard setting aligns more closely with standard classification protocols, where the model is required to make a single, definitive prediction for each instance, facilitating comparability with classical baselines. Second, adopting the soft-soft mode would have required calibrating probabilistic thresholds across multiple classes and levels, introducing additional complexity that was beyond the scope of this work. Future work may explore soft evaluation modes to further investigate the impact of uncertainty in model predictions. Additionally, accuracy and F1-score were computed to facilitate comparison with standard classification baselines.

Below, we report the results for each family of models. It is also important to clarify that, although the dataset contains tweets in both Spanish and English, this work addressed the task as a single multilingual classification problem. The evaluation was conducted jointly across both languages, without reporting results separately for each language and conducted on the official validation set of EXIST 2025, using the provided gold labels and evaluation scripts. It is important to note that the final test set of the challenge was not publicly available at the time of writing which explains the use of the validation set.

Table 5
Performance of traditional models in Task 1.1 (binary classification)

Configuration	ICM	ICM-Norm	F1
Basic - Naive Bayes	0.2553	0.6277	0.7471
Basic - SVM	0.3586	0.6782	0.7853
Lemma - Naive Bayes	0.2104	0.6853	0.7318
Lemma - SVM	0.2134	0.6092	0.7739
Stem - Naive Bayes	0.2257	0.6143	0.7345
Stem - SVM	**0.3686**	**0.6845**	**0.7986**

Table 6
Class-wise F1-scores for traditional models (Task 1.1)

Configuration	F1_YES	F1_NO
Basic - Naive Bayes	0.7858	0.7883
Basic - SVM	0.7743	0.7963
Lemma - Naive Bayes	0.6878	0.7749
Lemma - SVM	0.7679	0.7979
Stem - Naive Bayes	0.6875	0.7833
Stem - SVM	**0.7798**	**0.7996**

3.1.1. Traditional Machine Learning Approaches

As an initial benchmark for the binary classification task, we implemented six pipelines combining traditional text classification algorithms with varying levels of text normalization. The models evaluated were Multinomial Naive Bayes (NB) and Support Vector Machine (SVM), each trained with three preprocessing variants: Basic, Lemmatized, and Stemmed (see Section 2.2). This setup allowed us to explore the impact of linguistic normalization on model performance under controlled conditions.

Table 5 summarizes the overall performance of each model using the official evaluation metrics for the EXIST 2025 challenge over the validation set. The best performance was achieved by the SVM model using stemmed input text, obtaining an ICM score of 0.3686 and an F1-score of 0.7986. This configuration also produced the highest ICM-Norm value of 0.6845, indicating strong relevance in the predictions relative to a random baseline.

In general, SVM-based models outperformed Naive Bayes across all preprocessing strategies, particularly in terms of F1-score. The stemmed and lemmatized versions showed consistent gains over the basic variant, suggesting that reducing morphological variation improves model generalization. Interestingly, lemmatization and stemming produced similar trends in ICM, although stemming slightly outperformed lemmatization in all configurations.

To better understand the model's behavior across both classes ("YES" and "NO"), we also computed the per-class F1-scores. Table 6 shows the breakdown of F1 performance for each class. While most configurations achieved relatively balanced performance, the SVM-stemmed model again stood out with F1 scores of 0.7798 for the sexist class and 0.7996 for the non-sexist class, highlighting its robustness across both categories.

Overall, these results confirm that traditional learning pipelines can achieve strong baseline performance on the binary sexism detection task. Although more advanced transformer-based models will probably outperform them in absolute terms, these models remain competitive in terms of efficiency, interpretability, and implementation cost.

3.1.2. Monolingual Transformer Models

In this analysis, the monolingual setup was constructed by combining the predictions from two independent models: one trained exclusively on Spanish tweets using the roberta-base-bne model, and another

Table 7
Overall performance of the monolingual approach under different label aggregation strategies (Task 1.1)

Label Strategy	ICM	ICM-Norm	F1
Majority Label ($\geq$ 4 votes)	**0.5414**	**0.7708**	**0.8474**
Female Majority Vote	0.4820	0.7411	0.7841

Table 8
Class-wise F1-scores for monolingual models under different label strategies (Task 1.1)

Label Strategy	F1_YES	F1_NO
Majority Label ($\geq$ 4 votes)	**0.8423**	**0.8524**
Female Majority Vote	0.7730	0.7953

trained on English tweets using the bert-base-uncased model. These models, previously described in Section 2.3, were trained monolingually, meaning that each model was both trained and evaluated solely on data in its corresponding language. This unified evaluation allows assessing global performance while maintaining language-specific training conditions.

To evaluate the impact of different label construction strategies, two versions of this setup were compared using the same classification pipeline:

1. The first version used majority voting among the six annotators to define the final binary label. Tweets with a tie (3 YES, 3 NO) were excluded.
2. The second version applied the female majority resolution, where tied votes were resolved based on the majority opinion of the three female annotators.

Table 7 presents the evaluation results for the majority voting version. The model achieved an ICM score of 0.541, an ICM-Norm of 0.770, and an overall F1-score of 0.847. These values indicate strong alignment with the hierarchical structure of the labels and high binary classification performance.

In contrast, the table shows for the results obtained using the female-majority strategy, that the ICM decreased to 0.482, the ICM-Norm to 0.741, and the overall F1-score dropped to 0.784. These results suggest that, while the female-resolved strategy enables the inclusion of more data points by avoiding tweet exclusion, it introduces more label noise or inconsistency with the gold standard used in evaluation.

When analyzing class-wise performance in Table 8, the superiority of the majority label strategy remains evident. Both the "YES" and "NO" classes achieve higher F1-scores under this setting. It is worth noting that in both versions, separate F1-scores were also computed for the YES and NO classes. The results show a consistent trend: although both models perform well in identifying non-sexist content (NO), their ability to detect sexist tweets (YES) is more sensitive to the label construction strategy.

Overall, the monolingual setup using the majority-vote labels achieved the best balance between structural consistency (ICM), normalized agreement (ICM-Norm), and binary performance (F1).

3.1.3. Multilingual Transformer Models

This section presents the results obtained by multilingual Transformer models in the binary classification task. The models evaluated in this section correspond to the multilingual Transformer architectures previously introduced in Section 2.4, where their theoretical foundations and implementation details were described. All models were evaluated under both label aggregation strategies, as consistently applied throughout the study.

As observed in Table 9, all models show slightly better performance under the majority label setting compared to the female-only voting strategy. The model *mdeberta-v3-base* stands out across both configurations, achieving the highest ICM (0.515), ICM-Norm (0.757), and F1 (0.839) when using the

Table 9
Performance of multilingual models using different label strategies - Task 1.1

Model	ICM	ICM-Norm	F1
Majority Label			
bert-base-multilingual-cased	0.447724	0.723960	0.816075
distilbert-base-multilingual-cased	0.374388	0.687276	0.791643
mdeberta-v3-base	**0.515474**	**0.757850**	**0.838725**
roberta-base	0.419252	0.709717	0.806611
xlm-roberta-base	0.471750	0.735978	0.823969
Female Majority Vote			
bert-base-multilingual-cased	0.466331	0.733267	0.779267
distilbert-base-multilingual-cased	0.385892	0.693043	0.753727
mdeberta-v3-base	0.501906	0.751063	0.790458
roberta-base	0.444355	0.722247	0.772299
xlm-roberta-base	0.491175	0.745695	0.786895

Table 10
Per-class F1 scores for multilingual models - Task 1.1

Model	F1_YES	F1_NO
Majority Label		
bert-base-multilingual-cased	0.807175	0.824974
distilbert-base-multilingual-cased	0.784530	0.798755
mdeberta-v3-base	**0.832408**	**0.845041**
roberta-base	0.799557	0.813665
xlm-roberta-base	0.823151	0.824786
Female Majority Vote		
bert-base-multilingual-cased	0.771488	0.788746
distilbert-base-multilingual-cased	0.739267	0.768173
mdeberta-v3-base	0.778926	0.801990
roberta-base	0.760248	0.784353
xlm-roberta-base	0.773019	0.800770

majority label. This suggests its robustness across linguistic variations and annotation schemes. *Xlm-roberta-base* also performs consistently well compared to the other models.

Table 10 provides a more detailed view of the classification behavior for each class. The scores for F1_NO tend to be slightly higher, indicating that detecting non-sexist content is somewhat easier across all models. Still, models like *mdeberta-v3-base* and *xlm-roberta-base* achieve strong performance in both classes, maintaining F1_YES scores above 0.83 under the majority label setup.

In contrast, when trained with female-only voting labels, all models experienced a slight drop in both F1_YES and F1_NO. This confirms the trend already observed in monolingual models, where the female-only labeling introduces greater variability or subjectivity, potentially affecting model generalization.

Overall, multilingual models proved highly competitive, particularly *mdeberta-v3-base*, which outperformed others across all metrics. These findings validate the usefulness of multilingual pretraining and highlight the effectiveness of cross-lingual encoders in capturing complex societal constructs such as sexism.

Compared to the monolingual Transformer models, the best multilingual models achieved slightly lower ICM scores but remained close in overall F1 performance, especially under the majority label setting. In contrast, all multilingual Transformers clearly outperformed the traditional machine learning

Table 11
Comparison of best-performing models by approach – Task 1.1

Approach	Model	ICM	F1
Traditional ML	Stem - SVM	0.3686	0.7986
Monolingual Transformers (Majority Label strategy)	roberta-base-bne + bert-base-uncased	0.5414	0.8474
Multilingual Transformers	mdeberta-v3-base	0.5155	0.8387

Table 12
Performance of ensemble methods using different label aggregation strategies (Task 1.1)

Method	ICM	ICM-Norm	F1
Majority Label			
Majority Voting	0.525370	0.762800	0.842019
Weighted ICM	0.523000	0.761000	0.841800
Female Majority Vote			
Majority Voting	0.530454	0.765343	0.799440
Weighted ICM	0.529500	0.764200	0.799000

Table 13
Per-class F1 scores of ensemble methods under different label aggregation strategies (Task 1.1)

Method	F1_YES	F1_NO
Majority Label		
Majority Voting	0.837004	0.847034
Weighted ICM	0.836700	0.846800
Female Majority Vote		
Majority Voting	0.787056	0.811823
Weighted ICM	0.786200	0.811300

models (SVM and Naive Bayes), both in terms of structural alignment and class-wise F1-scores. This reinforces the advantage of deep multilingual architectures in complex classification tasks like sexism detection.

3.1.4. Ensemble Methods

To further improve classification performance, ensemble techniques were explored by combining the outputs of the five multilingual Transformer models introduced in section 2.4. This ensemble of Transformer-based models was designed to leverage the diversity in architectures and training dynamics to enhance robustness and generalization. Two strategies were implemented: majority voting and weighted voting based on ICM.

The following tables show the results of the ensemble under both label aggregation settings. Notably, the majority label configuration yielded slightly better scores overall. Among the two ensemble strategies, the majority voting performed best, with marginal differences compared to the other,

As shown in Table 12, all ensemble strategies yielded highly similar results under both label settings, indicating robustness across aggregation methods. Under the majority label, the best performance was achieved using majority voting, with an ICM of 0.525 and F1 of 0.842, marginally outperforming the weighted alternative. Similarly, for the female-vote strategy, majority voting again achieved the highest F1 score (0.799), though the differences across methods were minimal.

In Table 13, we observe that ensemble methods maintain a strong balance between classes, with

Table 14
Performance of CNN-BERT hybrid models on Task 1.1

Model	ICM	ICM-Norm	F1
CNN + bert-base-uncased	0.282004	0.641063	0.759337
CNN + bert-base-multilingual-cased	**0.29114**	**0.645634**	**0.761842**
CNN + roberta-base	0.261377	0.630746	0.750468
CNN + distilbert-base-uncased	0.2783	0.639211	0.757504

Table 15
F1 scores for CNN-BERT hybrid models - Task 1.1

Model	F1_YES	F1_NO
CNN + bert-base-uncased	0.734287	0.784466
CNN + bert-base-multilingual-cased	0.733656	0.790290
CNN + roberta-base	0.714819	0.786116
CNN + distilbert-base-uncased	0.728814	0.786194

slightly higher scores for the "NO" class across all configurations. This trend holds for both label aggregation strategies, with F1_NO exceeding F1_YES by approximately 1%–2% in most cases.

When using the majority label strategy, ensemble models achieved F1_YES scores above 0.836 and F1_NO values around 0.846, indicating excellent sensitivity to both classes. The female-vote variant showed a small drop in performance, with F1_YES ranging from 0.785 to 0.787 and F1_NO remaining above 0.811. These results are still strong, but the wider gap suggests that detecting sexist tweets becomes slightly more challenging under the female-only label aggregation.

Overall, ensemble transformers not only improved macro performance (as seen in the previous section), but also ensured stable and high-quality predictions across both classes. This consistency reinforces their reliability, especially in applications where balanced classification between positive and negative cases is essential.

3.1.5. Hybrid CNN-BERT Architectures

The hybrid CNN-BERT architectures evaluated in this section aim to combine the contextual encoding capacity of pretrained Transformer models with the local feature extraction ability of Convolutional Neural Networks (CNNs). The goal was to assess whether adding a CNN classification head on top of static token embeddings would improve the model's performance.

All models in this section were trained using the majority label strategy, which had shown consistently better results in previous experiments. The architecture used was based on extracting token embeddings from the pretrained BERT-based encoder (without fine-tuning) and feeding them into a CNN classifier trained on top. The convolutional block included filters of multiple sizes to capture different n-gram patterns, followed by max pooling, dropout, and a fully connected layer for classification. The tokenizers corresponding to each encoder were used during preprocessing.

The following table summarizes the results obtained using this hybrid architecture:

Among the four models tested, the best performing configuration was CNN + *bert-base-multilingual-cased*, achieving the highest ICM (0.291), ICM-Norm (0.646), and F1 (0.762). This suggests that multilingual pretraining, even without encoder fine-tuning, provides stronger token-level representations when paired with CNN classifiers.

To better understand class-level behavior, Table 15 reports the F1 scores for each class (YES and NO).

In terms of class-specific performance, all models exhibit slightly higher F1 scores for the non-sexist class. This is consistent with prior experiments and reflects the tendency of classifiers to favor dominant or less ambiguous categories. The gap between F1_YES and F1_NO is most pronounced in the *RoBERTa* configuration, potentially indicating limitations in capturing the subtle linguistic indicators of sexist

Table 16
Performance of zero-shot LLMs on sample of Task 1.1

Model	ICM	ICM-Norm	F1	Accuracy
vicgalle/xlm-roberta-large-xnli-anli	**-0.1289**	**0.4395**	**0.5783**	**0.6351**
MoritzLaurer/mDeBERTa-v3-base-mnli-xnli	-0.2693	0.3642	0.5225	0.5694
joeddav/xlm-roberta-large-xnli	-0.2600	0.3688	0.5439	0.5848
cross-encoder/nli-deberta-v3-large	-0.1698	0.4144	0.5358	0.5975
google/flan-t5-large (zero-shot)	-0.9844	0.0080	0.0173	0.0086

intent when representations are not fine-tuned.

Overall, while hybrid CNN-BERT models are outperformed by fully fine-tuned Transformer models, their performance is competitive and demonstrates that learned static embeddings can be successfully reused with simpler architectures when training efficiency or architectural modularity is a priority.

3.1.6. LLM Prompting

This section evaluates the performance of large language models (LLMs) using prompt-based learning strategies, specifically zero-shot and few-shot prompting. Due to the computational cost associated with running inference over the entire dataset, the experiments in this section were conducted on a representative sample of the data. In the event that one of the LLM approaches had proven superior, a full-scale evaluation would have followed. However, as discussed below, this was not the case.

In the zero-shot setting, the model is given a task description, such as “Is this tweet sexist?” and candidate labels (“YES” / “NO”) without having seen any task-specific examples. The evaluation includes five different models:

1. vicgalle/xlm-roberta-large-xnli-anli
2. MoritzLaurer/mDeBERTa-v3-base-mnli-xnli
3. joeddav/xlm-roberta-large-xnli
4. cross-encoder/nli-deberta-v3-large
5. google/flan-t5-large

All models except the last one were evaluated using the Hugging Face zero-shot-classification pipeline. These models are optimized for natural language inference (NLI) and treat classification as a premise-hypothesis matching task. In contrast, *flan-t5-large* is a generative model that processes prompts as free-form text, which also enables few-shot learning.

The results for zero-shot prompting are summarized in Table 16.

As shown, none of the evaluated zero-shot models outperform traditional fine-tuned classifiers. Among them, *vicgalle/xlm-roberta-large-xnli-anli* yields the best results with an F1-score of 0.578 and an ICM score of -0.128, although still significantly lower than supervised approaches. Surprisingly, *flan-t5-large* performs poorly in the zero-shot setting, possibly due to its generative nature and sensitivity to prompt formulation.

Although all models were tested in a zero-shot setting, few-shot prompting was only applicable to *google/flan-t5-large*, which supports textual prompts with in-context examples. Other models, designed for structured classification via NLI pipelines, do not support this mode of inference.

In this experiment, we designed a multilingual instructional prompt including five curated examples, three for the "YES" class and two for the "NO" class. The examples were chosen to represent various expressions of sexism and non-sexism in both English and Spanish.

Despite the richness and balanced nature of the prompt, performance did not improve over the zero-shot variant. As shown in Table 17 all key metrics were extremely low.

Per-class F1 scores, presented in Table 18, further confirm that the model failed to distinguish between the two classes, barely exceeding random guessing.

Table 17
Performance of few-shot prompting

Model	ICM	ICM-Norm	F1	Accuracy
flan-t5-large (few-shot)	-0.9847	0.0075	0.0168	0.0086

Table 18
Per-class F1 scores for few-shot prompting

Model	F1_YES	F1_NO
flan-t5-large (few-shot)	0.0131	0.0206

Table 19
Performance of RAC-enhanced models on Task 1.1 (binary classification)

Model	ICM	ICM-Norm	F1
XLM-RoBERTa (RAC)	0.406943	0.703560	0.802489
mDeBERTa-v3 (RAC)	**0.454886**	**0.727542**	**0.817209**
Monolingual Combination (RAC)	0.122839	0.561446	0.691822

Table 20
Per-class F1 scores of RAC-enhanced models - Task 1.1

Model	F1_YES	F1_NO
XLM-RoBERTa (RAC)	0.798255	0.806723
mDeBERTa-v3 (RAC)	0.795647	0.838772
Monolingual Ensemble (RAC)	0.615603	0.768041

These results suggest that, at least under this setup, the model was unable to effectively leverage in-context examples. This may be due to the limited size of the prompt, differences in domain between the examples and the test data, or intrinsic limitations in adapting generative LLMs to binary classification without additional fine-tuning or calibration strategies.

3.1.7. Retrieval-Augmented Classification (RAC)

This section presents the performance of the RAC-enhanced Transformer models described in Section 2.8 . Each model was trained using the majority label aggregation strategy and evaluated using the ICM metric suite and per-class F1 scores. Both multilingual and monolingual variants were tested, and the retrieval mechanism was implemented using a Sentence-BERT encoder with top-3 neighbor retrieval from the training corpus.

As shown in Table 19, integrating retrieved context via RAC did not yield consistent improvements over the baseline models. Both *XLM-RoBERTa* and *mDeBERTa-v3* show reduced ICM and ICM-Norm scores compared to their vanilla versions, although their overall F1-scores remain relatively strong. The monolingual ensemble, however, experienced a considerable drop in performance across all metrics, suggesting that the injected context may have introduced noise rather than helpful information.

In Table 20, we observe that *mDeBERTa-v3* achieves the highest F1_NO score (0.838772), while *XLM-RoBERTa* shows more balanced performance across both classes. On the other hand, the monolingual RAC model drastically underperforms in F1_YES, indicating a reduced ability to identify sexist tweets once the context is injected. This highlights that multilingual models handle the additional contextual input more effectively than monolingual architectures.

When comparing multilingual models, `mdeberta-v3-base` shows the strongest performance under the RAC setup, consistent with its competitive results in the standard fine-tuning setting. However,

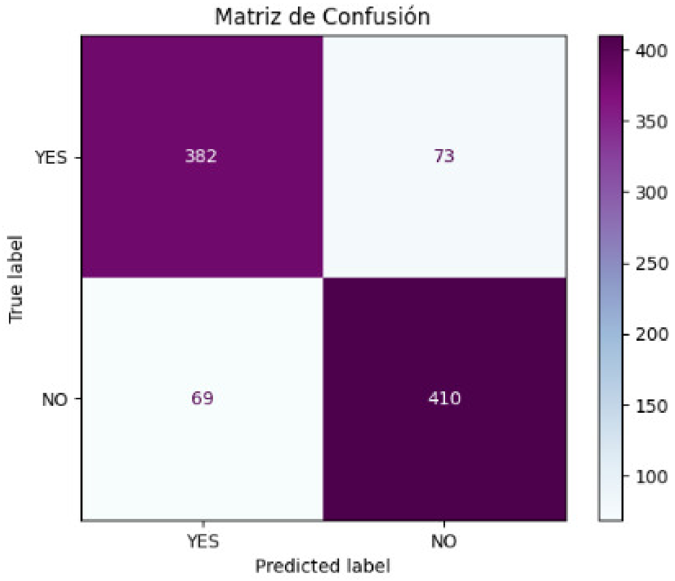


Figure 4: Confusion matrix – Task 1.1.

`xlm-roberta-base` struggles to benefit from context injection, indicating potential limitations in how it integrates cross-sentence information.

To conclude, the Retrieval-Augmented Classification setup does not consistently outperform the base fine-tuned models. These findings suggest that, while promising in theory, RAC requires more careful retrieval and filtering mechanisms to deliver measurable gains in this task.

3.1.8. Error Analysis

Table 21 presents a comparative summary of all models evaluated on Task 1.1 using ICM. Among all individual models, the best overall performance was achieved by the monolingual model trained with majority label aggregation, reaching an ICM of 0.5414. In the multilingual category, ensemble methods also improved performance, with the best ensemble (majority voting with majority labels) achieving an ICM of 0.5305.

Before examining the specific types of errors made by individual models, it is essential to explore how and why even high-performing systems can fail in nuanced classification scenarios. Error analysis provides qualitative insights that complement aggregate metrics like ICM or F1-score. By identifying systematic misclassifications, especially between semantically close classes or in the presence of linguistic ambiguity, we can better understand model behavior and uncover areas where further improvements or adjustments may be required.

To obtain a richer understanding of the model's limitations and decision patterns, we perform an error analysis focused on the final predictions. This analysis is conducted exclusively on the best-performing model for Task 1.1, which is the monolingual approach using majority-based label construction.

We begin by analyzing the overall distribution of prediction outcomes. A confusion matrix reveals that the model maintains a relatively balanced precision and recall across the "YES" and "NO" classes, though it makes a comparable number of false positives and false negatives (Figure 4). To better understand the nature of these errors, we label each prediction as either correct, a false positive, or a false negative, which serves as the foundation for the following analyses.

Table 22 shows the classification performance of the model evaluated. The results indicate that the model achieves balanced precision and recall across both classes, with a slight advantage in the detection of the "NO" class. The overall F1-scores are comparable, and the global accuracy reaches 84.8%, confirming the robustness of the system across categories

An initial qualitative analysis is provided using word clouds generated from the tweets that were

Table 21
Summary of ICM performance across all model families on Task 1.1.

Model Type	Model	ICM
Traditional ML	Basic - Naive Bayes	0.2553
	Basic - SVM	0.3586
	Lemma - Naive Bayes	0.2104
	Lemma - SVM	0.2134
	Stem - Naive Bayes	0.2257
	Stem - SVM	**0.3686**
Monolingual	**Majority Label**	**0.5414**
	Female Majority Vote	0.4820
Multilingual	bert-base-multilingual-cased (Majority)	0.4477
	distilbert-base-multilingual-cased (Majority)	0.3744
	mdeberta-v3-base (Majority)	**0.5155**
	roberta-base (Majority)	0.4193
	xlm-roberta-base (Majority)	0.4718
	bert-base-multilingual-cased (Female)	0.4663
	distilbert-base-multilingual-cased (Female)	0.3859
	mdeberta-v3-base (Female)	0.5019
	roberta-base (Female)	0.4444
	xlm-roberta-base (Female)	0.4912
CNN-BERT Hybrid	CNN + bert-base-uncased	0.2820
	CNN + bert-base-multilingual-cased	**0.2911**
	CNN + roberta-base	0.2614
	CNN + distilbert-base-uncased	0.2783
Zero-shot LLMs	**vicgalle/xlm-roberta-large-xnli-anli**	**-0.1289**
	MoritzLaurer/mDeBERTa-v3-base-mnli-xnli	-0.2693
	joeddav/xlm-roberta-large-xnli	-0.2600
	cross-encoder/nli-deberta-v3-large	-0.1698
	google/flan-t5-large (zero-shot)	-0.9844
Few-shot LLM	flan-t5-large (few-shot)	-0.9847
Ensemble	Majority Voting (Majority Label)	0.5254
	Weighted ICM (Majority Label)	0.5230
	Majority Voting (Female Vote)	**0.5305**
	Weighted ICM (Female Vote)	0.5295
RAC (Retrieval-Augmented)	XLM-RoBERTa (RAC)	0.4069
	mDeBERTa-v3 (RAC)	**0.4549**
	Monolingual Ensemble (RAC)	0.1228

misclassified. Separate visualizations were created for false positives and false negatives to highlight recurring terms associated with each error type (Figures 5a and 5b).

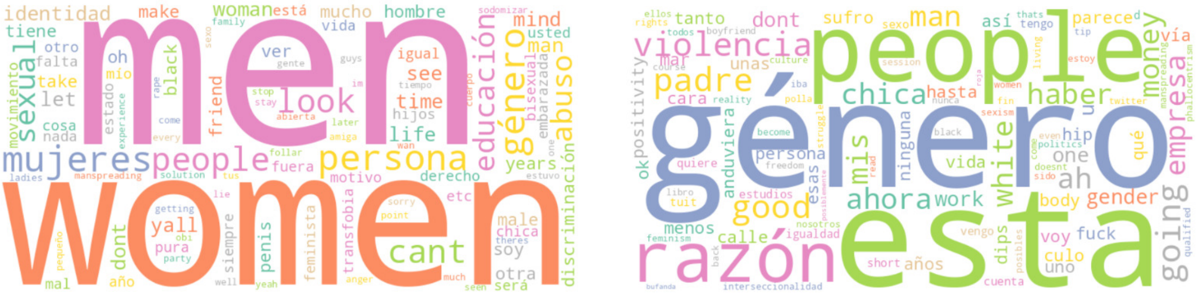
In both error categories, terms related to gender and identity, such as *women, men, género,* and *persona*, appear prominently, indicating that the model often struggles with correctly interpreting tweets that revolve around gender issues. This could be attributed to semantic overlap between neutral statements and those expressing subtle forms of sexism, making the classification task inherently challenging.

In the case of false positives, the word cloud (Figure 5a) shows a notable presence of words like *women, men, abuso, género, and educación*, which often occur in educational or awareness-raising contexts. This suggests that the model may be overly sensitive to these terms, flagging non-sexist content as sexist due to keyword presence without fully grasping the intent or tone.

Conversely, the false negatives word cloud (Figure 5b) includes emotionally charged or socially critical terms such as *violencia, padre, culpa*, and *money*. This implies that in some cases, tweets containing

Table 22
Classification report for the best monolingual model - Task 1.1

Label	Precision	Recall	F1-score	Support
NO	0.849	0.856	0.852	479
YES	0.847	0.840	0.843	455
Accuracy		0.848		934



(a) False Positives

(b) False Negatives

Figure 5: Most frequent terms found in tweets misclassified as false positives and false negatives.

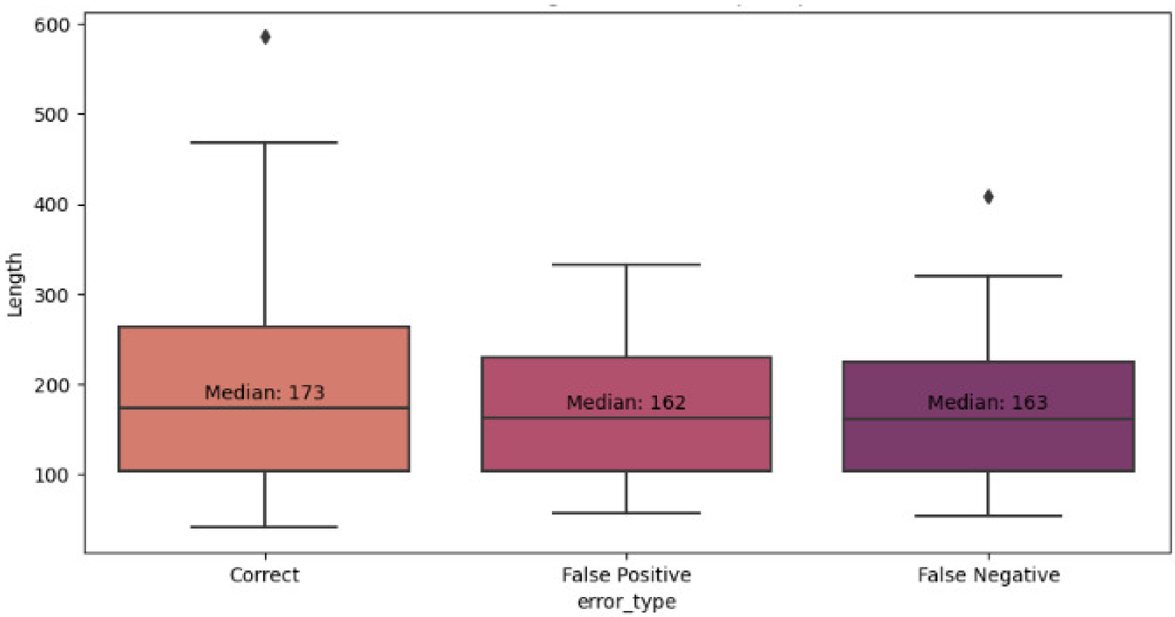


Figure 6: Distribution of tweet length by error type.

actual sexist content may go undetected if the language is less explicit or uses indirect expressions.

Overall, this visualization highlights the limitations of surface-level keyword-based associations and underlines the need for better contextual reasoning in detecting subtle or implicit forms of sexism.

We also examine whether tweet length is associated with classification performance. Figure 6 shows a boxplot comparing tweet length (in number of characters) across error types. While the overall distributions are similar, correct predictions tend to have slightly longer tweets on average (median: 173), compared to false positives (median: 162) and false negatives (median: 163). This suggests that longer tweets may provide more contextual clues that help the model make correct decisions.

To explore whether sentiment influences the model's behavior, we compute the sentiment polarity of each tweet using the TextBlob library.

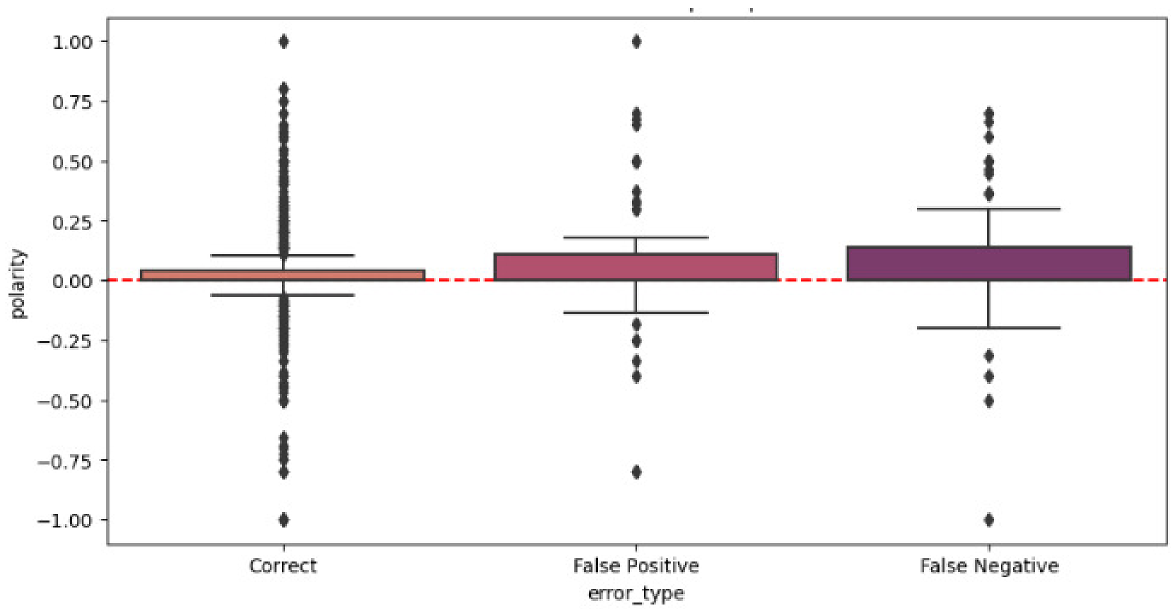


Figure 7: Distribution of tweet's polarity by error type.

Table 23
Examples of false positive classifications

Tweet	True	Pred
—Insistir en que los genitales determinan el género. Puede ser microagresión por falta de educación, puede ser transfobia pura y dura. —Relacionar determinados diagnósticos con el colectivo LGTBIQA+. Me ahorro el comentario.	NO	YES
@Dialogo_es @FuerzasMilCol si asesinar mujeres embarazadas, bombardear niños y sacar los ojos a estudiantes, pero con los actores armados si son cagado no a otro perro con ese hueso	NO	YES
Calling A Man Bald Is Sexual Harassment" https://t.co/MiSGkB89bs via @YouTube	NO	YES
the question is do i wear the very short skirt that literally shows my ass with kneehighs or opt out for a short	NO	YES

As shown in Figure 7, tweets misclassified by the model tend to display a slightly higher variance in sentiment polarity compared to correctly classified instances. This suggests that emotionally charged tweets may pose additional challenges for the classifier, possibly due to the presence of sarcasm, indirect language, or nuanced judgment.

We also examine the distribution of errors across languages (Figure 8). Since the monolingual strategy involved using a dedicated Spanish model for tweets in Spanish and a separate English model for tweets in English, this visualization helps compare their behavior. The number of correct predictions is slightly higher in Spanish, but both languages show a similar and balanced distribution of false positives and false negatives. This indicates that both monolingual models performed comparably and that no major disparity in misclassification patterns emerged between languages.

Lastly, to complement the quantitative findings, we conduct a qualitative inspection of misclassified examples, with special attention to challenging cases. We reviewed a selection of tweets from both false positives and false negatives, focusing on instances that involve irony, implicit bias, or subtle linguistic cues. These examples help illustrate the types of content that are particularly difficult for the model to classify correctly and provide insight into the nuanced nature of sexist discourse on social media.

Together, these findings provide valuable insight into how the model succeeds and where it fails and highlights the importance of complementing quantitative evaluation with detailed qualitative analysis

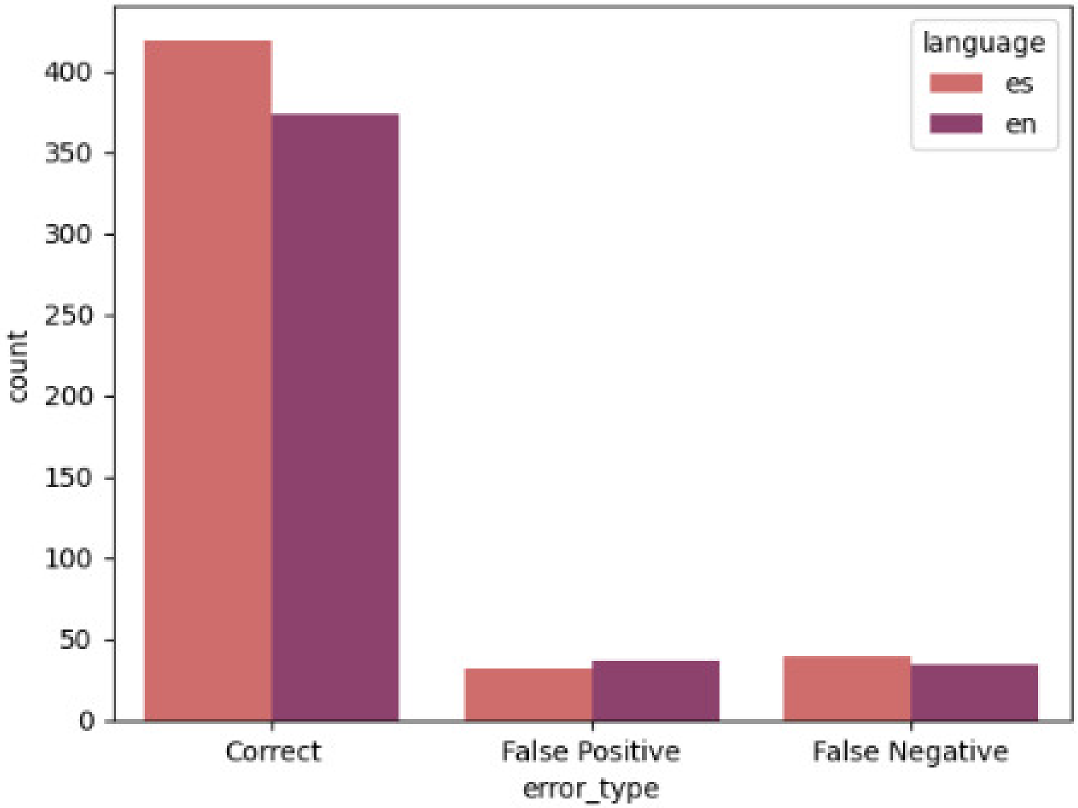


Figure 8: Distribution of errors across languages.

Table 24
Examples of false negative classifications.

Tweet	**True**	**Pred**
con q motivación voy a entrenar ahora si ya no veo a ninguna de las chicas	YES	NO
@Mistywoman1 Have I missed something, or has phallocentrism become the latest cult?	YES	NO
#Women should focus on money freedom, and not retirement! Today's women may still think about #retirement as "not having to work."	YES	NO

in sensitive classification tasks.

3.2. Results for Task 1.2 - Multi-class

This section presents the evaluation results for Task 1.2 of the EXIST 2025 challenge, which focuses on identifying the type of sexist content expressed in tweets. Unlike Task 1.1, which involved binary classification, this task adopts a more granular and hierarchical structure, requiring models to distinguish between four possible labels: *NO*, *DIRECT*, *REPORTED*, and *JUDGEMENTAL*.

As in the binary setting, all models were trained and evaluated using the official validation set provided by the organizers, following the hard-hard evaluation protocol. The label space exhibits a hierarchical structure, with *DIRECT*, *REPORTED*, and *JUDGEMENTAL* being subclasses of the broader category *YES*, while *NO* denotes the absence of sexist intent. This hierarchy introduces additional complexity, as misclassifications across the YES–NO boundary are considered more severe than misclassifications among YES subtypes.

All experiments in this section were assessed using the Information Contrast Measure (ICM) and its normalized version (ICM-Norm), which remain the official metrics of the competition. These were

Table 25
Performance of monolingual models in Task 1.2

Configuration	ICM	ICM-Norm	F1
Monolingual - (binary+multi, Female Voting)	**0.112682**	**0.535237**	**0.494928**
Monolingual - (binary+multi, Majority Voting)	-0.746904	0.266436	0.368215
Monolingual - (4-label, Majority Voting)	0.053612	0.518250	0.488920

complemented by standard classification metrics such as macro-averaged F1-score and per-class F1 scores to support interpretability and comparative analysis. The labels for Task 1.2 were constructed using the same procedure as in Task 1.1

Moreover, two training strategies were adopted to handle the hierarchical nature of Task 1.2:

1. In the first strategy, **Binary + Multi-Class Pipeline**, the classification process was split into two stages: a binary classifier was first applied to determine whether a tweet was sexist (*YES*) or not (*NO*), based on models previously described for Task 1.1. Only tweets predicted as sexist were then passed through a second classifier trained exclusively to distinguish among the three sexism subcategories: *DIRECT*, *REPORTED*, and *JUDGEMENTAL*. The training data for this second model was filtered accordingly to include only tweets labeled with these three classes.
2. The second strategy, **Single Multi-Class Model (4 Labels)**, followed a unified approach, where a single model was trained to classify among all four possible classes simultaneously: *NO*, *DIRECT*, *REPORTED*, and *JUDGEMENTAL*. This setup allowed for direct inference in a single step, without the need for a preceding binary decision. Both strategies were evaluated using the same official metrics and validation set to enable fair comparison.

The remainder of this section is organized by model family, beginning with monolingual transformer models and progressing toward multilingual, hybrid, LLM prompting and ensemble-based systems. Each subsection reports quantitative results and qualitative observations based on the model's behavior and its ability to discriminate among nuanced categories of sexist expression. Retrieval-augmented classification (RAC) approaches were not explored in Task 1.2, as their performance in Task 1.1 did not yield notable improvements compared to standard transformer-based models.

3.2.1. Monolingual Transformer Models

This section presents the results obtained by monolingual Transformer-based models in Task 1.2. We evaluated two pretrained models: bert-base-uncased, a general-purpose English BERT model, and roberta-base-bne, a RoBERTa variant trained on a large Spanish corpus. Both were fine-tuned for the multiclass classification task using the different strategies described earlier.

The following table reports the global metrics for each configuration:

Among the three configurations, the best results were achieved by the first configuration model trained using the binary-to-multiclass pipeline with labels derived from female annotators. It reached the highest ICM (0.113) and macro F1 (0.495), suggesting that this configuration captures sexist nuances more effectively than those based on majority voting.

The unified 4-label model trained with bert-base-uncased performed slightly better than the second configuration but still lagged behind the binary-to-multiclass pipeline using female-vote labels. Although this single-step approach simplifies inference by avoiding the need for a cascade, its performance remained modest, especially in capturing nuanced sexist categories.

In contrast, the model trained on majority vote labels via the binary+multi pipeline underperformed across all metrics, indicating that this configuration may introduce label noise or be less robust to the subtleties of sexism classification.

The next table shows the per-class F1 scores to analyze where the differences in performance arise:

Table 26
Per-class F1 scores for monolingual models in Task 1.2

Configuration	F1_NO	F1_DIRECT	F1_REPORTED	F1_JUDGEMENTAL
bert-base-uncased (binary+multi, Female Voting)	0.795522	0.567850	0.358209	0.258333
roberta-base-bne (binary+multi, Majority Voting)	0.673333	0.419068	0.146789	0.233129
bert-base-uncased (4-label, Majority Voting)	0.869935	0.639269	0.289474	0.160000

From the per-class analysis, we observe that all models are most accurate when predicting the NO class. The unified 4-label model obtains the highest F1 for NO (0.87), reflecting a bias toward the majority class. However, its performance on minority categories like JUDGEMENTAL remains weak (0.16).

On the other hand, the model trained with female-only labels (binary+multi) achieves more balanced results across all classes, especially for REPORTED (0.36) and JUDGEMENTAL (0.26), suggesting better sensitivity to minority and potentially ambiguous instances.

3.2.2. Multilingual Transformer Models

To evaluate the multilingual capabilities of transformer-based systems on Task 1.2, we explored a diverse set of 25 model combinations, covering various architectures (e.g., `bert-base-multilingual-cased`, `xlm-roberta-base`, `mdeberta-v3-base`) and training configurations. These include the binary+multi-class pipeline, under the female-majority and majority-vote label strategies, and unified 4-label classification.

Given the large number of experiments (55), Table 27 highlights the top five multilingual configurations based on ICM for each configuration, which reflects the system's ability to preserve label hierarchy and semantic alignment.

Among the top-performing setups, the binary-to-multiclass pipeline using majority-vote labels achieved the highest overall ICM, with the best configuration combining `mdeberta-v3-base` for binary classification and `bert-base-multilingual-cased` for multiclassification (ICM = 0.1982). Models trained using female-vote labels demonstrated more consistent and stable ICM values across the top-5, albeit slightly lower in absolute terms. This may point to a trade-off between label specificity and sample diversity.

Lastly, unified 4-label configurations delivered moderate results, highlighting the challenges of capturing class granularity without hierarchical decomposition. Nonetheless, all evaluated configurations outperform the baseline systems provided by the organizers, demonstrating the effectiveness of both the training strategies and the selected Transformer architectures.

3.2.3. Hybrid CNN-BERT Architectures

This subsection presents the results obtained with hybrid models that combine convolutional neural networks (CNNs) with BERT-based embeddings for the classification task. These models use pre-trained BERT variants to encode the input tweets, followed by a CNN architecture designed to capture local n-gram patterns from the contextualized token embeddings.

Table 28 shows the performance metrics for these models. Overall, the results indicate that CNN-BERT hybrids underperform in comparison to full transformer-based architectures. The best performing setup, CNN + `distilbert-base-uncased`, reached an ICM of -0.222653 and macro-F1 of 0.286267. While this configuration shows some ability to differentiate among classes, its limited capacity to capture deeper contextual relationships likely hinders its performance.

Interestingly, all hybrid models still outperform the official majority and minority baselines provided by the competition organizers, demonstrating that even simplified architectures can retain value when leveraging pre-trained embeddings.

Table 27
Top 5 multilingual configurations for Task 1.2.

Configuration	ICM	ICM-Norm	F1
Binary + Multiclass (Female Voting)			
mDeBERTa-v3 + BERT-multilingual	0.117429	0.536721	0.477857
mDeBERTa-v3 + DistilBERT-multilingual	0.101535	0.531751	0.462145
XLM-RoBERTa + BERT-multilingual	0.077933	0.524370	0.464292
XLM-RoBERTa + DistilBERT-multilingual	0.074999	0.523453	0.453387
mDeBERTa-v3 + XLM-RoBERTa	0.075769	0.523694	0.461574
Binary + Multiclass (Majority Voting)			
mDeBERTa-v3 + BERT-multilingual	**0.198160**	0.561966	0.564848
mDeBERTa-v3 + DistilBERT-multilingual	0.115280	0.536049	0.512752
mDeBERTa-v3 + mDeBERTa-v3	0.096999	0.530333	0.467753
BERT-multilingual + BERT-multilingual	0.100916	0.531557	0.552607
RoBERTa-base + BERT-multilingual	0.084005	0.526269	0.550814
Single Multi-Class Model (4 Labels)			
mDeBERTa-v3	0.178670	0.555872	0.557172
XLM-RoBERTa	0.004536	0.501419	0.503301
RoBERTa-base	-0.051916	0.483765	0.510500
BERT-multilingual	-0.067171	0.478995	0.489443
DistilBERT-multilingual	-0.240749	0.424715	0.417933
Baselines			
Majority baseline	-1.023420	0.179966	0.157877
Minority baseline	-2.908090	0.000000	0.033693

Table 28
Performance of CNN-BERT hybrid models on Task 1.2.

Model	ICM	ICM-Norm	F1
CNN + distilbert-base-uncased	**-0.222653**	0.439374	0.286267
CNN + roberta-base	-0.256391	0.430184	0.270992
CNN + bert-base-uncased	-0.246319	0.422974	0.261366
CNN + bert-base-multilingual-cased	-0.283758	0.411266	0.207143
Majority Baseline	-1.023420	0.179966	0.157877
Minority Baseline	-2.908090	0.000000	0.033693

3.2.4. LLM Prompting

This section presents the results obtained by LLMs under the zero-shot and few-shot prompting paradigms. As discussed in the methodology, the models were queried using task-specific instructions without any fine-tuning on the EXIST 2025 data. The goal was to evaluate their ability to handle the multi-class classification task of sexist intent detection, where predictions must distinguish between four categories: *NO, DIRECT, REPORTED, and JUDGEMENTAL.*

For the zero-shot experiments, five models were evaluated:

1. vicgalle/xlm-roberta-large-xnli-anli
2. MoritzLaurer/mDeBERTa-v3-base-mnli-xnli
3. joeddav/xlm-roberta-large-xnli

Table 29
Overall performance of LLMs on Task 1.2.

Model	ICM	ICM-Norm	F1
vicgalle/xlm-roberta-large-xnli-anli	-2.2356	0.0000	0.1603
MoritzLaurer/mDeBERTa-v3-base-mnli-xnli	-2.6275	0.0000	0.1218
joeddav/xlm-roberta-large-xnli	-2.4741	0.0000	0.1593
cross-encoder/nli-deberta-v3-large	-2.6751	0.0000	0.1195
flan-t5-large (zero-shot)	-1.5941	0.0015	0.0051
flan-t5-large (few-shot)	-1.5708	0.0088	0.0123

Table 30
Performance of ensemble methods - Task 1.2

Method	ICM	ICM-Norm	F1
Majority Label			
Majority Voting	0.0948	0.5296	0.4416
Weighted ICM	0.0940	0.5280	0.4390
Female Majority Vote			
Majority Voting	0.1092	0.5342	0.4717
Weighted ICM	0.1087	0.5335	0.4698

4. cross-encoder/nli-deberta-v3-large
5. google/flan-t5-large

In addition, the flan-t5-large model was also used in a few-shot configuration, leveraging manually constructed prompts with representative examples of each class.

All evaluations were performed on a representative sample of the validation set, and the results are summarized in the table below.

The results reveal that none of the prompting-based LLMs reached competitive performance on Task 1.2. All zero-shot models obtained negative ICM scores, with only marginal F1 scores, especially in the more nuanced categories. While some models showed better performance on the NO class, they struggled to correctly identify the sexist subcategories such as REPORTED or JUDGEMENTAL.

The few-shot configuration with *flan-t5-large* also failed to deliver meaningful improvements, despite the inclusion of manually curated examples. This confirms that Task 1.2, due to its fine-grained semantic structure and hierarchical label space, poses a significant challenge for prompting-only methods.

These findings support the broader observation made in Task 1.1: instruction-following models are still limited in their ability to perform complex social classification tasks without in-domain training or adaptation. While prompting remains a flexible and cost-effective approach, it does not currently outperform supervised learning in nuanced classification problems like sexist intent detection.

3.2.5. Ensemble Methods

To assess whether combining predictions from multiple multilingual models can further improve performance, we evaluated ensemble methods based on majority voting. As in Task 1.1, we tested two configurations: (i) plain majority vote, and (ii) weighted vote based on ICM.

Although ensemble methods yielded a slight performance boost in Task 1.1, this trend did not hold in Task 1.2. Despite improving over the competition baselines, ensemble models underperformed compared to the best individual models. This indicates that, in the context of multiclass classification, aggregation through majority or weighted voting can dilute the strengths of specialized models, leading to lower ICM scores than those achieved by carefully tuned single-model configurations.

Table 31
Summary of ICM performance across all model families on Task 1.2.

Model Type	Model	ICM
Monolingual	Binary+Multi (Female Voting)	0.1127
	Binary+Multi (Majority Voting)	-0.7469
	4-label (Majority Voting)	0.0536
Multilingual	mDeBERTa-v3 + BERT (Female)	0.1174
	mDeBERTa-v3 + DistilBERT (Female)	0.1093
	XLM-RoBERTa + BERT (Female)	0.0779
	XLM-RoBERTa + DistilBERT (Female)	0.0593
	mDeBERTa-v3 + XLM-RoBERTa (Female)	0.0796
	mDeBERTa-v3 + BERT (Majority)	**0.1982**
	mDeBERTa-v3 + DistilBERT (Majority)	0.1583
	BERT + BERT (Majority)	0.1096
	RoBERTa-base + BERT (Majority)	0.0840
	mDeBERTa-v3 (4-label)	0.1787
CNN-BERT Hybrid	CNN + distilbert-base-uncased	-0.2227
	CNN + roberta-base	-0.2564
	CNN + bert-base-uncased	-0.2463
	CNN + bert-base-multilingual-cased	-0.2838
Zero-shot LLMs	vicgalle/xlm-roberta-large-xnli-anli	-2.2356
	MoritzLaurer/mDeBERTa-v3-base-mnli-xnli	-2.6675
	joeddav/xlm-roberta-large-xnli	-2.4741
	cross-encoder/nli-deberta-v3-large	-2.6751
	google/flan-t5-large (zero-shot)	-1.5941
Few-shot LLM	flan-t5-large (few-shot)	-1.5708
Ensemble	Majority Voting (Majority Label)	0.0948
	Weighted ICM (Majority Label)	0.0940
	Majority Voting (Female Vote)	0.1092
	Weighted ICM (Female Vote)	0.1087

3.2.6. Error Analysis

Table 31 summarizes the performance of all model families evaluated in Task 1.2 using the Information Contrast Measure, the official metric of the EXIST 2025 challenge. This task required multi-class classification of sexist intent, including fine-grained distinctions between types of sexist expression. The configurations encompass monolingual and multilingual transformers, hybrid CNN-BERT architectures, zero- and few-shot LLM prompting, as well as ensemble strategies. For multilingual setups, we evaluated both cascaded pipelines (binary followed by multi-class) and unified 4-label models.

To complement the quantitative evaluation presented above, we conducted a detailed error analysis focused on the best-performing model in Task 1.2. The objective of this analysis is to better understand the limitations and failure modes of the system, particularly in distinguishing between the nuanced classes of sexist content defined in the EXIST 2025 framework.

The selected model for this study is the multilingual pipeline composed of *mDeBERTa-v3* followed by *BERT-multilingual*, trained using majority-vote labels. This configuration achieved the highest ICM among all tested systems in Task 1.2 and serves as a strong candidate for qualitative inspection.

This analysis aims to uncover potential sources of bias or ambiguity that challenge the model's decision-making process, and to inform future improvements in handling complex, real-world instances of sexist expression.

Figure 9 shows the confusion matrix over the validation set. Most predictions for the NO class are accurate, with 409 correct predictions. However, the model frequently confuses NO with DIRECT (46 instances) and, to a lesser extent, with REPORTED and JUDGEMENTAL. Among the sexist categories, the

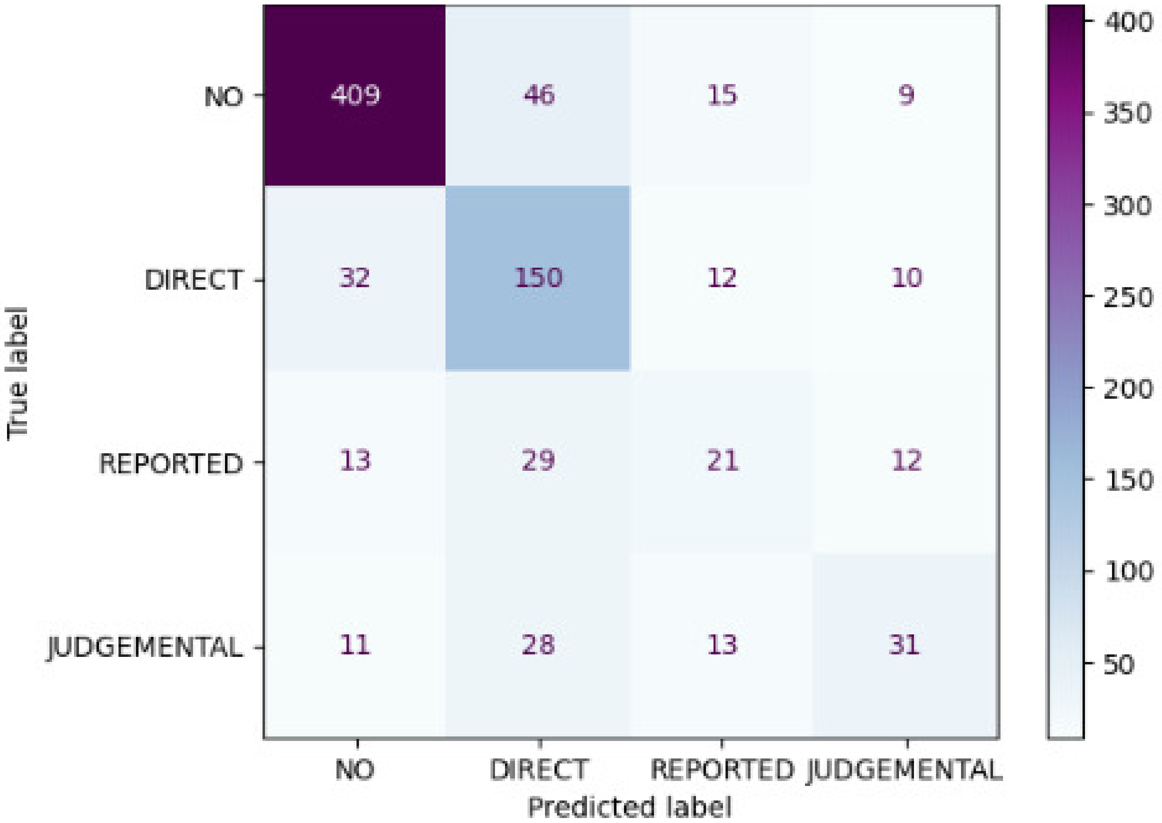


Figure 9: Confusion matrix for Task 1.2.

Table 32
Most frequent misclassification types in Task 1.2.

True Label	Predicted Label	Count
NO	DIRECT	46
DIRECT	NO	32
REPORTED	DIRECT	29
JUDGEMENTAL	DIRECT	28

most common misclassification is between REPORTED and DIRECT (29 cases), which are semantically closer and often share overlapping lexical patterns.

To further analyze these errors, Table 32 summarizes the most frequent misclassification pairs.

These patterns suggest that the model tends to favor more explicit categories like DIRECT over more contextual ones such as JUDGEMENTAL or REPORTED, especially when lacking overt sexist keywords.

To explore whether the model's errors correlate with the length of the input, we compared the character length of tweets across true labels (Figure 10). We observed that tweets labeled as REPORTED and JUDGEMENTAL tend to be longer on average than NO and DIRECT, possibly indicating that more complex or narrative-like expressions are harder to classify correctly.

To identify lexical patterns that may contribute to prediction biases, we generated word clouds for tweets predicted as each of the four classes, as shown in Figures 11a–11d.

These visualizations confirm the presence of overlapping terms across classes. For example, *"mujer"*, *"mujeres"*, *"men"*, *"women"*, and *"feminismo"* appear across all sexist categories, making fine-grained distinction particularly challenging.

We also examined the sentiment polarity of tweets predicted as each class using TextBlob. Figure 12 shows the distribution. Tweets predicted as NO tend to have slightly more positive sentiment, while other categories remain centered near neutrality. However, the presence of negative sentiment in

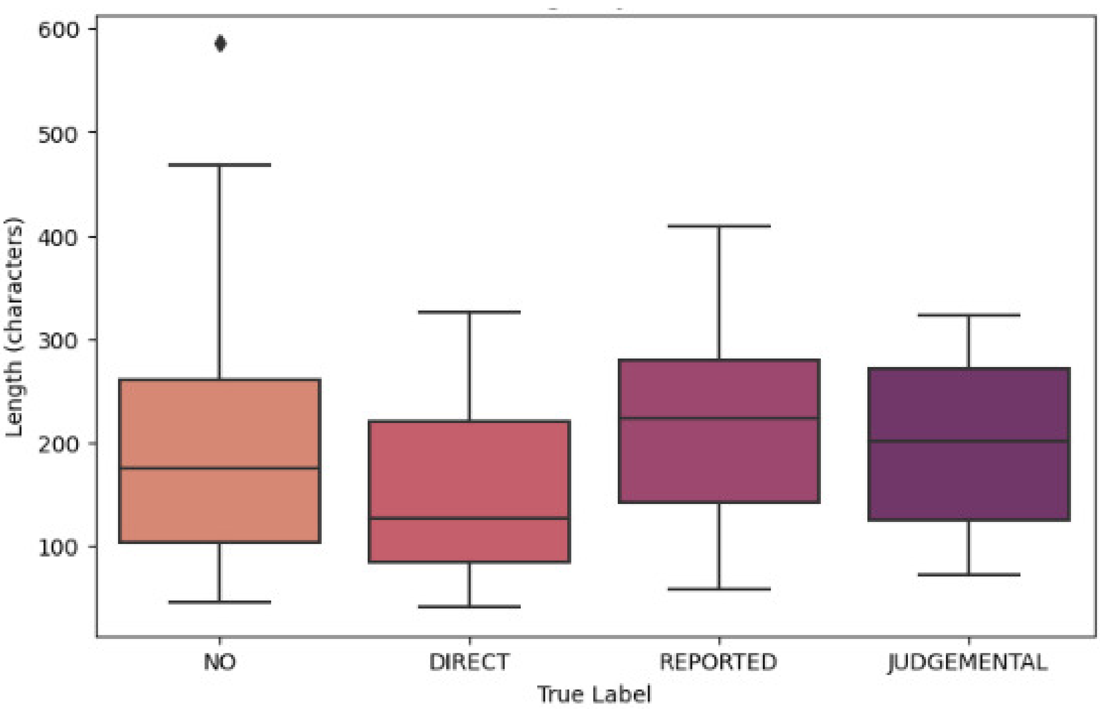
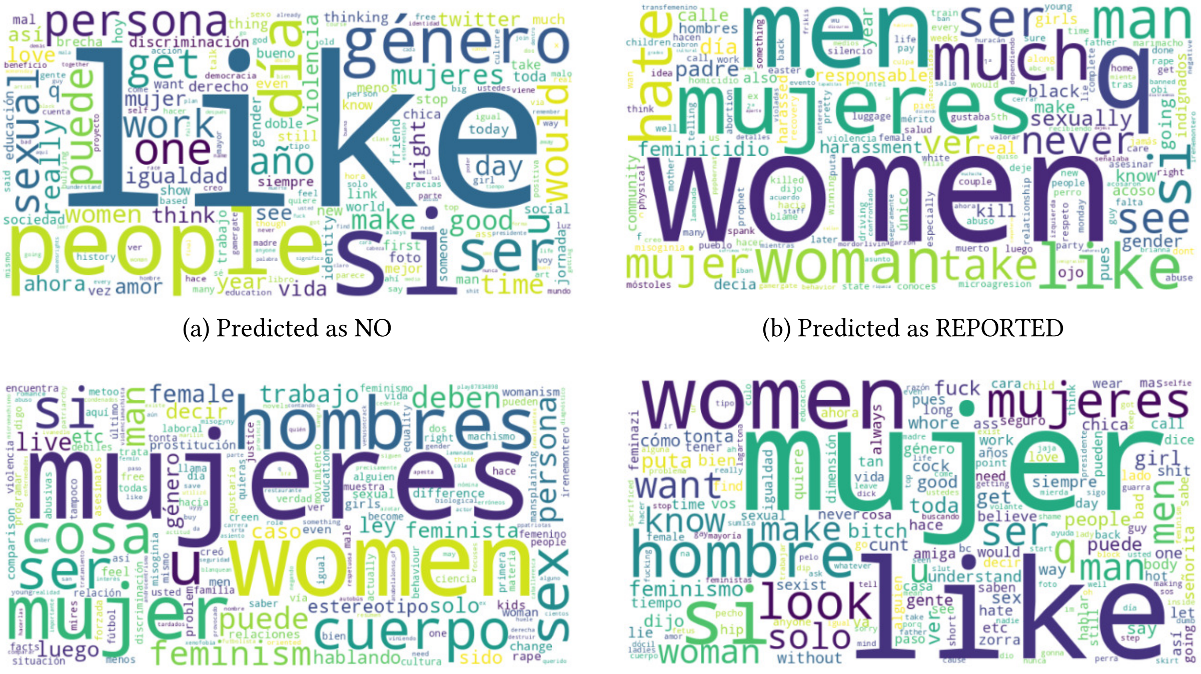


Figure 10: Tweet length distribution by true label - Task 1.2.



(a) Predicted as NO

(b) Predicted as REPORTED

(c) Predicted as JUDGEMENTAL

(d) Predicted as DIRECT

Figure 11: Word clouds for tweets predicted as each class (Task 1.2).

JUDGEMENTAL and REPORTED tweets may reflect the emotional tone of feminist critique, which the model may misinterpret.

To complement the quantitative results, Table 33 shows a selection of representative misclassified tweets. These examples illustrate typical edge cases, such as subtle irony, vague references to gender, or complex constructions that defy straightforward labeling.

Overall, this analysis reveals that most misclassifications stem from subtle semantic differences

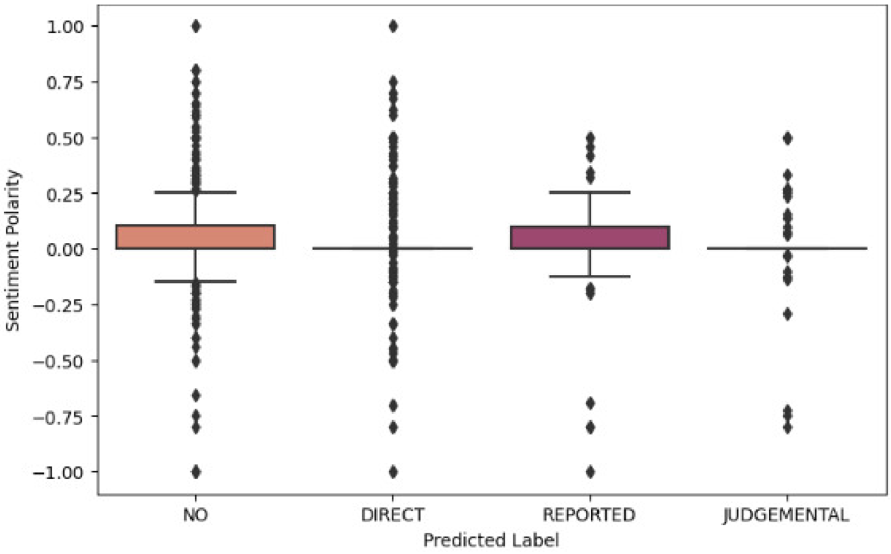


Figure 12: Sentiment polarity distribution by predicted label (Task 1.2).

Table 33
Example misclassifications for the top model in Task 1.2

Tweet	True Label	Predicted Label
@JetCelestial @polgara28951124 @CharlotteEmmaUK I don't know what you're on about? My point was that you shouldn't mock other people's looks, because it's very shallow and makes you look like a cunt. Thanks for your input though	NO	DIRECT
@joelaltonmoore @KMisGrand @TheMrBarramundi @ZipPulse A vasectomy is a simple surgery done in an office, hospital, or clinic. 24 hours later recovery is pretty much complete! Tubal ligation recovery takes 1-3 weeks, longer if it's done following a C-section or childbirth. Many women can't afford to take 1-3 weeks off work!	NO	REPORTED
@tuckednuts Notice something else tho…Almost every single one, is a man. That's such a fucking problem that this is how some men believe they should let out their angers and hatred. This society needs to do better, it does not even benefit the men. Patriarchy/misogyny fails everyone	NO	JUDGEMENTAL
@TheNon_Nun Never knew you all share tits, how did we stop being pals again yeah ?	DIRECT	NO
made this top and my mom gave me the "you look like a whore" expression anyways here's this shirt i braided myself https://t.co/0j2mMv47wl	REPORTED	DIRECT
That's why they always tell you..you will never understand women. https://t.co/Q6KaaNVYRL	JUDGEMENTAL	DIRECT

between classes, the use of implicit or ironic language, and overlapping vocabulary across categories. While the model performs well at detecting clear sexist content (DIRECT), it still struggles with nuanced discourse and context-dependent expressions.

4. Conclusions and Future Work

4.1. Key Findings

This paper addressed the problem of automatic sexism detection in social media through participation in the EXIST 2025 shared task. The work explored two complementary tasks: the binary classification of sexist content (Task 1.1) and the multi-class classification of the communicative intent behind sexist expressions (Task 1.2). A wide range of models were implemented and compared, from traditional machine learning pipelines to advanced Transformer-based architectures, hybrid CNN-BERT structures,

Table 34
Comparison of Task 1.1 Results (EXIST 2025 and EXIST 2024)

System	Edition	ICM-Hard Norm	F1-YES
Mario_1	2025	0.8405	0.8167
CIMAT-GTO_2	2025	0.8165	0.7996
CIMAT-GTO_3	2025	0.8144	0.7968
UC3M-LI	2025	0.7581	0.7613
NYCU-NLP_1	2024	0.8002	0.7944

Table 35
Comparison of Task 1.2 Results (EXIST 2025 and EXIST 2024)

System	Edition	ICM-Hard Norm	Macro F1
Mario_1	2025	0.6623	0.5692
CIMAT-GTO_3	2025	0.6521	0.5555
CIMAT-GTO_2	2025	0.6428	0.5582
UC3M-LI	2025	0.5175	0.4583
ABCD Team_1	2024	0.6320	0.5677

prompting with large language models, ensemble techniques, and retrieval-augmented classification.

Among the key findings:

- The best overall performance in Task 1.1 was achieved by a monolingual Transformer-based approach combining RoBERTa-base for Spanish and BERT-base for English, reaching an ICM of 0.541 and an F1-score of 0.847.
- In Task 1.2, the highest performance was obtained by a two-step multilingual pipeline combining mDeBERTa-v3 for binary classification and multilingual BERT for the stance classification, with an ICM of 0.198 and F1-score of 0.564.
- Ensemble methods provided moderate gains in Task 1.1 but proved less effective in Task 1.2, where the hierarchical label structure introduced noise into majority-vote aggregation.
- Prompting-based approaches, both zero-shot and few-shot, were consistently outperformed by fine-tuned supervised models, confirming that complex socio-linguistic tasks like sexism detection still benefit from task-specific adaptation.
- Label construction strategies had a significant impact on performance. In particular, resolving ties through majority voting among female annotators improved the quality of the training data, especially in Task 1.2.

Additional techniques such as data augmentation and the use of hybrid CNN-BERT models contributed to robustness but did not surpass the performance of fine-tuned Transformers.

In addition to the internal experimental results, the official results for the EXIST 2025 shared task have been published and are available on the competition's official website [7].

In Task 1.1, the top three performing systems were Mario_1 (ICM-Hard Norm: 0.8405, F1-YES: 0.8167), CIMAT-GTO_2 (ICM-Hard Norm: 0.8165, F1-YES: 0.7996), and CIMAT-GTO_3 (ICM-Hard Norm: 0.8144, F1-YES: 0.7968). Our team (UC3M-LI) achieved an ICM-Hard Norm of 0.7581 and an F1-YES of 0.7613, positioning it competitively but behind the leading teams. For reference, the best system in the 2024 edition was NYCU-NLP_1, which achieved an ICM-Hard Norm of 0.8002 and an F1-YES of 0.7944.

In Task 1.2, the top three systems in 2025 were Mario_1 (ICM-Hard Norm: 0.6623, Macro F1: 0.5692), CIMAT-GTO_3 (ICM-Hard Norm: 0.6521, Macro F1: 0.5555), and CIMAT-GTO_2 (ICM-Hard Norm: 0.6428, Macro F1: 0.5582). The UC3M-LI system obtained an ICM-Hard Norm of 0.5175 and a Macro F1 of 0.4583. For comparison, the best system in the 2024 edition for this task was ABCD Team_1, with an ICM-Hard Norm of 0.6320 and a Macro F1 of 0.5677.

These comparative results confirm that while our systems delivered robust performance in both tasks, a measurable performance gap remains when compared to the top-ranked teams. This gap highlights areas for future improvement, particularly in addressing the hierarchical complexity of Task 1.2 and further optimizing the handling of ambiguous and nuanced cases.

Overall, the project demonstrated that multilingual and monolingual Transformer models remain the most reliable and effective tools for sexism detection in textual data. Moreover, incorporating annotator metadata in the labeling process provided both methodological insight and a practical advantage in improving dataset quality.

4.2. Limitations

Despite the strong performance of several models, this work presents some limitations:

- The dataset used for training and evaluation, while large, presents a moderate class imbalance, particularly in Task 1.2, where the "DIRECT" category is notably more frequent than the others. Additionally, the subjective nature of the task may introduce annotation ambiguity, which could have constrained the performance ceiling.
- The reliance on hard label aggregation (majority vote or female vote) may oversimplify cases where annotators genuinely disagree, potentially discarding valuable uncertainty information.
- The exploration of prompting methods was limited to a small sample due to computational constraints, and no prompt tuning or calibration was applied.
- The use of retrieval-augmented classification (RAC) did not yield clear improvements, likely due to the simplicity of the retrieval strategy (top-k similarity), which may not have provided semantically helpful context.
- Finally, due to time and resource constraints, the generalization of models was not tested across other sexism-related datasets, which limits conclusions about domain transfer.

4.3. Future Directions

Building on the results and insights of this project, several directions can be considered for future research:

- Explore label modeling techniques that go beyond hard aggregation, such as soft labels or learning-with-disagreement (LwD), to better capture annotator uncertainty and subjectivity.
- Integrate demographic metadata (gender, age, education level) not only in labeling decisions but also as input features to train fairer and more interpretable models.
- Investigate more advanced prompting techniques, including chain-of-thought prompting, instruction tuning, or LLM calibration, and test models like GPT-4 or LLaMA 3 in full evaluations. In this work, a preliminary attempt was made using the instruction-tuned model ***google/gemma-2b-it*** [52]. However, the results were not satisfactory, and due to time limitations, no further tuning or prompt optimization was explored. As a consequence, this approach was excluded from the main methodology, although its initial evaluation is documented and may serve as a basis for future experiments.
- Refine the retrieval component in RAC models by incorporating semantic filtering, diversity-aware sampling, or using multi-hop context.
- Extend the framework to multimodal classification by including image-text memes and applying joint embedding models such as CLIP or BLIP.
- Finally, although monolingual models were trained separately for Spanish and English, no language-specific evaluation was reported. All systems, including multilingual ones, were assessed on the entire dataset without distinguishing performance per language. Future work could address this by analyzing potential differences in model behavior or bias across languages.

5. Acknowledgments

Grant PID2023-148577OB-C21 (Human-Centered AI: User-Driven Adapted Language Models- HUMAN_AI) by MICIU/AEI/ 10.13039/501100011033 and by FEDER/UE.

Declaration on Generative AI

During the preparation of this work, the author used Chat-GPT-4 in order to: Grammar and spelling check. After using these tool, the author reviewed and edited the content as needed and take full responsibility for the publication's content.

References

[1] European Institute for Gender Equality, Sexism – glossary & thesaurus, 2017. URL: https://eige.europa.eu/thesaurus/terms/1222.

[2] Encyclopedia Britannica Contributors, Sexism, 2024. URL: https://www.britannica.com/topic/sexism.

[3] Ministerio de Igualdad de España, Casi tres de cada cuatro mujeres jóvenes se han visto expuestas a comentarios sobre su físico en redes sociales, 2024. URL: https://www.igualdad.gob.es/comunicacion/notasprensa/casi-tres-de-cada-cuatro-mujeres-jovenes-se-han-visto-expuestas-a-comentarios-sobre-su-fisico-en-redes-sociale

[4] A. Amorim, T. Almeida, M. Silva, Measuring sexist content in social media: Evidence from machine learning analysis of tweets, Social Network Analysis and Mining (????). URL: https://ceur-ws.org/Vol-3740/paper-117.pdf.

[5] J. Drakett, B. Rickett, K. Day, K. Milnes, Old jokes, new media – online sexism and constructions of gender in internet memes, Feminism & Psychology 28 (2018) 89–109. URL: https://doi.org/10.1177/0959353517727560. doi:10.1177/0959353517727560.

[6] ACM womENcourage 2024, Hackathon: Hacking sexism online, 2024. URL: https://womencourage.acm.org/2024/hackathon/.

[7] EXIST 2025 Organizers, Exist 2025: sexism identification in social networks, 2024. URL: https://nlp.uned.es/exist2025/.

[8] L. Plaza, J. C. de Albornoz, V. Ruiz, A. Maeso, B. Chulvi, P. Rosso, D. Chulvi, D. Spina, Overview of exist 2024—learning with disagreement for sexism identification and characterization in tweets and memes, in: Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the CLEF 2024 Conference, Lecture Notes in Computer Science, Springer, 2024.

[9] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. 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 (2020).

[10] Y. Kim, Convolutional neural networks for sentence classification, Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP) (2014).

[11] S. Edunov, M. Ott, M. Auli, D. Grangier, Understanding back-translation at scale, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2018.

[12] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, P. Kuksa, A. Fan, M. Lewis, W.-t. Yih, T. Rocktäschel, S. Riedel, D. Kiela, Retrieval-augmented generation for knowledge-intensive nlp tasks, in: Advances in Neural Information Processing Systems, 2020. URL: https://proceedings.neurips.cc/paper_files/paper/2020/file/6b493230205f780e1bc26945df7481e5-Paper.pdf.

[13] G. Salton, C. Buckley, Term-weighting approaches in automatic text retrieval, Information Processing & Management (1988).

[14] S. Bird, E. Klein, E. Loper, Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit, O'Reilly Media Inc., 2009.

[15] C. D. Manning, P. Raghavan, H. Schütze, Introduction to Information Retrieval, Cambridge University Press, 2008.

[16] C. Cortes, V. Vapnik, Support-vector networks, Machine Learning (1995). doi:10.1007/BF00994018.

[17] A. McCallum, K. Nigam, A comparison of event models for naive bayes text classification, in: AAAI-98 workshop on learning for text categorization, 1998.

[18] T. Joachims, Text categorization with support vector machines: Learning with many relevant features, in: European conference on machine learning, Springer, 1998.

[19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in Neural Information Processing Systems (NeurIPS), 2017.

[20] 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), Association for Computational Linguistics, Minneapolis, Minnesota, 2019. URL: https://aclanthology.org/N19-1423.

[21] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, Language models are unsupervised multitask learners, OpenAI Blog (2019).

[22] G. A. L. Team, Bert base model (uncased), https://huggingface.co/bert-base-uncased, 2018.

[23] 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).

[24] B. S. Center, T. Group, Roberta base bne model for spanish, https://huggingface.co/BSC-TeMU/roberta-base-bne, 2021.

[25] Google AI Language, bert-base-multilingual-cased, https://huggingface.co/google-bert/bert-base-multilingual-cased, ????

[26] Microsoft, microsoft/mdeberta-v3-base, https://huggingface.co/microsoft/mdeberta-v3-base, 2021. Accessed: 2025-06-15.

[27] Hugging Face, distilbert-base-multilingual-cased, https://huggingface.co/distilbert/distilbert-base-multilingual-cased, 2019.

[28] Facebook AI, xlm-roberta-base, https://huggingface.co/facebook/xlm-roberta-base, 2020.

[29] Facebook AI, roberta-base, https://huggingface.co/facebook/roberta-base, 2019.

[30] P. He, X. Liu, J. Gao, W. Chen, Deberta: Decoding-enhanced bert with disentangled attention, arXiv preprint arXiv:2006.03654 (2020).

[31] V. Sanh, L. Debut, J. Chaumond, T. Wolf, Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter, arXiv preprint arXiv:1910.01108 (2019).

[32] 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 (2020).

[33] OpenAI, Gpt-4 technical report, https://openai.com/research/gpt-4, 2023.

[34] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, et al., Llama: Open and efficient foundation language models, arXiv preprint arXiv:2302.13971 (2023).

[35] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, et al., Palm: Scaling language modeling with pathways, arXiv preprint arXiv:2204.02311 (2022).

[36] V. Galle, vicgalle/xlm-roberta-large-xnli-anli, 2022. Accessed via Hugging Face at https://huggingface.co/vicgalle/xlm-roberta-large-xnli-anli.

[37] J. Davison, joeddav/xlm-roberta-large-xnli, 2020. Accessed via Hugging Face at https://huggingface.co/joeddav/xlm-roberta-large-xnli.

[38] M. Laurer, Moritzlaurer/mdeberta-v3-base-mnli-xnli, 2023. Accessed via Hugging Face at https://huggingface.co/MoritzLaurer/mDeBERTa-v3-base-mnli-xnli.

[39] H. Face, cross-encoder/nli-deberta-v3-large, 2021. Accessed via Hugging Face at https://huggingface.co/cross-encoder/nli-deberta-v3-large.

[40] Google, google/flan-t5-large, 2022. Accessed via Hugging Face at https://huggingface.co/google/flan-t5-large.

[41] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, X. He, S. N. Wang, X. Li, S. Narang, et al., Scaling instruction-finetuned language models, arXiv preprint arXiv:2210.11416 (2022).

[42] T. G. Dietterich, Ensemble methods in machine learning, International workshop on multiple classifier systems (2000) 1–15.

[43] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, D. Kulkarni, M. Lewis, W.-t. Yih, T. Rocktäschel, S. Riedel, V. Stoyanov, Retrieval-augmented generation for knowledge-intensive nlp tasks, arXiv preprint arXiv:2005.11401 (2020). URL: https://arxiv.org/pdf/2005.11401.

[44] G. Izacard, T. Hospedales, S. Riedel, D. Bouchacourt, Few-shot learning with retrieval-augmented language models, arXiv preprint arXiv:2208.03299 (2022).

[45] N. F. Liu, M. Gardner, S. Singh, M. Gardner, W. Merrill, Y. Belinkov, Lost in the middle: How language models use long contexts, arXiv preprint arXiv:2307.03172 (2023).

[46] N. Reimers, I. Gurevych, paraphrase-multilingual-minilm-l12-v2, https://huggingface.co/sentence-transformers/paraphrase-multilingual-MiniLM-L12-v2, 2020. Accessed: 2025-06-01.

[47] K. Guu, K. Lee, Z. Tung, P. Pasupat, M.-W. Chang, Retrieval augmented language model pre-training, in: Proceedings of the 37th International Conference on Machine Learning, PMLR, 2020. URL: http://proceedings.mlr.press/v119/guu20a.html.

[48] S. Feng, V. Gangal, J. Wei, S. Chandar, S. Vosoughi, D. R. Mitra, A survey of data augmentation approaches for nlp, arXiv preprint arXiv:2105.03075 (2021).

[49] R. Sennrich, B. Haddow, A. Birch, Improving neural machine translation models with monolingual data, in: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL), 2016.

[50] J. Tiedemann, S. Thottingal, Opus-mt — open translation models, https://huggingface.co/Helsinki-NLP, 2020.

[51] E. Amigó, H. Delgado, Information contrast model: Evaluating classifiers through information divergence from a random baseline, Information Processing & Management (2022). doi:10.1016/j.ipm.2022.103070.

[52] Google, google/gemma-2b-it, https://huggingface.co/google/gemma-2b-it, 2024.