DataSummit at MultiPRIDE: Context-Aware Multilingual Detection of Slur Reclamation in LGBTQ+ Contexts

Federico Dingeo, Marco Viviani*

Università degli Studi di Milano-Bicocca, Dipartimento di Informatica, Sistemistica e Comunicazione (DISCo) – Edificio U14 (ABACUS), Viale Sarca, 336 – 20126 Milan, Italy

Abstract

The detection of hate speech in social media is a core challenge for Natural Language Processing (NLP) systems, particularly when offensive terms are used in non-harmful ways through the phenomenon of semantic reclamation. Reclaimed slurs, often employed by targeted communities as markers of identity and empowerment, pose significant difficulties for automatic moderation systems, which risk misclassifying legitimate content as abusive. This issue is especially relevant in multilingual and identity-rich contexts, such as discourse related to the LGBTQ+ community. In this work, the task of reclaimed slur recognition is addressed within the MultiPRIDE challenge at EVALITA 2026, focusing on Task B (Contextual Content), which incorporates user-level contextual information. A stratified classification approach that combines contextual embeddings from state-of-the-art multilingual Transformer models with engineered features derived from sentiment analysis, emotion detection, and user metadata is proposed. A multilingual training strategy is adopted to enhance cross-lingual generalization. Promising results were achieved during the development phase, showing the benefits of combining linguistic and metadata features. Nevertheless, the evaluation on the official test set revealed limitations in robustness. This highlights the difficulty of modeling semantic reclamation and the need for further improvements to ensure consistent performance on new data.

Keywords

Reappropriative intent, Semantic reclamation, LGBTQ+, Natural Language Processing, Text Classification

1. Introduction

In recent years, social media have emerged as one of the primary channels for written communication, making the detection of unmoderated content a central concern for *Natural Language Processing* (NLP) systems [1]. Within this landscape, the identification of *hate speech* and *abusive language* represents a particularly challenging task, as it requires a careful balance between limiting harmful content and safeguarding freedom of expression. A crucial aspect in this domain is the phenomenon of *semantic reclamation*, defined as the reappropriation of terms historically used as insults or slurs by the communities that were originally targeted by them [2]. In such cases, the offensive connotation of a term may be attenuated or transformed, acquiring instead a function related to identity affirmation and collective pride. NLP systems that fail to account for this phenomenon risk producing systematic misclassifications, ultimately penalizing the very groups they are designed to protect. Despite the extensive body of literature on *hate speech detection*, only a limited number of recent studies have explicitly addressed the distinction between "offensive use" and "reclaimed use" of slurs, such as [3]. This limitation becomes particularly salient in contexts characterized by a high density of identity-related language and slang, such as discourse within the LGBTQ+ community [4].

This work is conducted in the context of the **MultiPRIDE** challenge [5] at EVALITA 2026 [6], which focuses on the automatic recognition of reclaimed slurs in texts primarily extracted from Twitter. The task covers three languages—Italian, Spanish, and English—and concentrates on content related to the LGBTQ+ community. In addition to textual data, the dataset provided by the organizers includes contextual user information, when available, enabling a more nuanced modeling of legitimacy in linguistic

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

*Corresponding author.

f.dingeo@campus.unimib.it (F. Dingeo); marco.viviani@unimib.it (M. Viviani)
https://ikr3.disco.unimib.it/people/marco-viviani/ (M. Viviani)
0009-0000-3203-0894 (F. Dingeo); 0000-0002-2274-9050 (M. Viviani)

usage. The objective of this study is twofold: to analyze the performance of different classification models on the reclaimed slur recognition task, and to contribute to a broader understanding of the challenges posed by semantic reclamation in multilingual NLP settings.

More specifically, our work addresses **Task B (Contextual Content)**, which leverages user biographical information to support classification decisions. To improve cross-lingual robustness, we adopted a multilingual training strategy aimed at enhancing model generalization. To tackle the task, we propose a stratified approach. As a *baseline*, we implement a Logistic Regression model based on TF–IDF representations. The core of the system, however, relies on contextual embeddings extracted from state-of-the-art multilingual Transformer models (*mDeBERTa V3*, *XLM-RoBERTa*, and *Multilingual MiniLM*), combined with linear classifiers. To better capture the semantic nuances associated with reclaimed language, we further enrich the vector representations through a *feature engineering* process that integrates signals derived from sentiment analysis, emotion detection, and metadata analysis (e.g., the use of emojis in user biographies). A subsequent feature selection phase is applied to reduce dimensionality and identify the most informative predictors. The robustness of the proposed system is assessed through a 10-fold cross-validation procedure on the training dataset.

The remainder of the paper is organized as follows: Section 2 introduces MultiPRIDE, describing its data and tasks; Section 3 presents the methodology adopted in this work, including exploratory data analysis, preprocessing pipeline, feature engineering strategies, and model architectures; Section 4 reports and discusses the experimental results; finally, Section 5 concludes the paper.

2. The MultiPRIDE Dataset and Tasks

This section presents the dataset released by MultiPRIDE and gives an overview of the proposed tasks, with a focus on the task considered in this work.

2.1. Description of the Dataset

The data collection provided for the MultiPRIDE task covers the period 2020–2022 and was constructed by integrating existing resources. The sources vary significantly depending on the language: while Italian and Spanish texts come exclusively from Twitter (sourced respectively from the TWITA collection and from specific LGBTQ+-themed datasets), the English subset was obtained by aggregating data from Twitter, Reddit, and TV series dialogues.

To build the final dataset, a common filtering process was applied across all languages. In the first phase, a keyword-based selection was performed using offensive terms (slurs) extracted from the HurtLex lexicon. Subsequently, the selection was refined to capture sentences with a high probability of reappropriative usage, filtering for positive terms expressing pride and community belonging (e.g., *pride*, *rainbow*).

The resulting dataset is multilingual and includes texts in Italian (1,086 samples), Spanish (876 samples), and English (1,026 samples). The three collections were merged into a single dataset containing 2,988 rows. Each instance in the dataset is associated with five variables:

- `id`: a unique identifier of the instance;
- `text`: the textual content;
- `bio`: the user biography, when available;
- `label`: the target label, a binary variable where 1 denotes the presence of words in reclaimed usage and 0 otherwise;
- `lang`: the language of the text.

2.2. Description of the Task(s)

The MultiPRIDE shared task comprises two binary classification tasks that aim to identify reappropriative intent in social media messages containing potentially derogatory terms related to the LGBTQ+ community.

- In **Task A**, systems are required to determine whether a message conveys reappropriative intent solely based on its textual content, i.e., considering only the message text;
- **Task B** extends this setting by permitting the use of contextual metadata associated with the author, specifically the biographical information provided in the *bio* field. This additional contextual information can assist in disambiguating between derogatory and reappropriative language use, enabling a more informed and accurate classification.

In this work, the focus is exclusively on **Task B**, as will be further clarified in the following Methodology section.

3. Methodology

This section describes the methodology adopted in this work. An exploratory analysis of the data is presented first, followed by the preprocessing steps applied to the dataset. Next, the feature engineering process and the model architectures used in the experiments are outlined.

3.1. Exploratory Data Analysis

Exploratory analysis proved essential to understanding the true nature of the data, identifying linguistic and contextual properties potentially relevant to the task, and to better prepare the next steps.

3.1.1. Assessing Data Quality and Class Distribution

A quick analysis of missing values revealed that the `bio` variable contains missing values for all English entries and for a portion of the Italian and Spanish entries (1,161 missing values in total). It is important to note that, although the task focused on exploiting biographical information, which is available only for Italian and Spanish, the English data were nevertheless included in the training phase. This strategy was followed with the aim of increasing the dataset size and helping the model generalize better by providing additional observations, even in the absence of biographical context.[1]

Next, we checked for duplicate observations and removed five entries where the entire row (except for the `id`) was identical.

Inspecting the target variable (`label`), we observed a clear prevalence of Class 0 (non-reclaimed/offensive) over Class 1 (reclaimed). This imbalance is consistent across the three languages in the dataset and reflects the relative rarity of reclaimed uses compared to non-reclaimed ones. Such a distribution poses additional challenges for automatic recognition of the minority class, making the task sensitive to class imbalance.

3.1.2. Analysis of Texts and Biographies

For the token analysis, the texts and biographies were tokenized using NLTK's `TweetTokenizer`,[2] which is well-suited for social media content where hashtags, emojis, and slang are common.

Token counts per text show a positive skew, with most observations between 20 and 50 tokens. Stratifying by class, Class 0 texts tend to be longer and more dispersed, with numerous outliers, whereas Class 1 texts are shorter and more compact. This suggests that text length could be an informative feature for classification.

Biography lengths are more homogeneous, ranging from a few tokens up to about 40, and no relevant differences are observed between classes, indicating a lesser role for this feature.

We also examined the presence of biographies in relation to the target variable, excluding English texts due to complete missingness. Although minor percentage differences exist, no clear patterns suggest this feature as discriminative.

[1] In light of the results actually obtained, as illustrated below, this solution may have proved suboptimal in retrospect due to potential meaning shifts or information loss, as also discussed in the limitations section.

[2] https://www.nltk.org/api/nltk.tokenize.casual.html

Additional stylistic features were considered. The proportion of fully uppercase words shows largely overlapping distributions between classes, with no significant differences in medians. Similarly, “aggressive punctuation”, i.e., defined as sequences of at least three consecutive ‘!’ or ‘?’, is rare and shows no meaningful class differences.

Overall, biography presence, uppercase usage, and aggressive punctuation do not provide sufficient evidence to warrant inclusion as primary features in the classification model.

3.1.3. Analysis of Hashtags and Emojis

Hashtags and emojis were analyzed using *Log-Odds* to identify discriminating terms for each class. Log-Odds measure how much a term is associated with one class versus the other by comparing its relative frequency in both groups. Positive values indicate a stronger connection to Class 1, while negative values indicate a stronger connection to Class 0.

It is important to note that the hashtag analysis was performed on the texts, while the emoji analysis was carried out on the biographies, with the goal of potentially identifying users belonging to the community more easily.

For Italian and English, no significant hashtags emerged. For Spanish, the term “orgullo” (pride) was strongly associated with Class 1. However, hashtags were **not** used as features in the final model, as the analysis showed high variability and inconsistent results: semantically similar hashtags sometimes had positive and sometimes negative Log-Odds values, even for those containing the term “orgullo”. This instability made the use of hashtags potentially unreliable and could have caused the model to overfit on specific keywords rather than learning the actual context of reclamation.

Regarding emojis in biographies, Class 1 users used emojis more frequently (~50%) compared to Class 0 users (~30%). In particular, three emojis showed a very strong association with the reclaimed class: the rainbow, the rainbow flag, and the transgender flag.

Table 1
Analysis of the top emojis in user biographies associated with the reclaimed class, based on Log-Odds.

Emoji	Prop_0	Prop_1	Log-Odds
🌈	0.020	0.055	1.010
🏳️‍🌈	0.088	0.187	0.759
🏳️‍⚧️	0.019	0.037	0.674

Note: Prop_0 and Prop_1 represent the relative frequency of the emoji within the respective class.

3.1.4. Analysis of Pronouns

Finally, the distribution of pronouns was analyzed using the `Stanza` library,[3] focusing on pronouns associated with the subject or the verb to avoid double-counting. For each text, the number of occurrences of first-person expressions (singular or plural) and of second- or third-person expressions (singular or plural) was calculated. Based on these counts, one of the following labels was assigned to each text:

- **First person**: if the count of first-person occurrences is greater than that of other persons;
- **Neutral**: if the counts are equal;
- **Other persons**: if the count of second- and third-person occurrences is greater than that of the first person.

The analysis showed that texts belonging to Class 1 have a significantly higher proportion of cases with a predominance of first-person pronouns compared to Class 0. This finding supports the linguistic hypothesis of *reclamation*: when community members reclaim a slur, they tend to speak in the first person to express identity and belonging. In contrast, offensive or hate speech is more frequently expressed in the second or third person, reflecting distance or attack.

[3] https://stanfordnlp.github.io/stanza/

3.2. Text Preprocessing and Representation

In this stage, three distinct data cleaning operations were performed at different points in the workflow, each targeting a specific purpose and corresponding to different stages of text representation and preprocessing.

1. The **first preprocessing phase** was designed to make the *Exploratory Data Analysis* (EDA) more effective. In this phase, a minimal cleaning was applied: multiple spaces were reduced to a single space, and emojis were converted into a textual representation (*demojization*). This step was necessary because, during the EDA, it was observed that after tokenizing texts and biographies, the `TweetTokenizer` tended to split some emojis into sequences of multiple symbols, making analyses of token counts and emojis less interpretable;
2. The **second cleaning phase** was applied immediately before using pre-trained models to extract additional features, such as sentiment and emotion associated with the text. In this phase, HTML entities, user mentions, and URLs were handled; multiple spaces were also reduced to one;
3. The **third and final preprocessing phase** was applied before creating the text representations used for model training. This phase was divided into two parts, reflecting the two types of text representations used:
   - (a) The first representation is based on **TF-IDF** (*Term Frequency Inverse Document Frequency*), which, together with a Logistic Regression classifier, served as a *baseline.* Here, preprocessing was more extensive and aimed at noise reduction: all characters were converted to lowercase, URLs and mentions were removed, while for hashtags only the '#' symbol was removed, preserving the alphabetic part. All non-alphabetic characters were removed, and multiple spaces were reduced. After tokenization, stopwords were removed using the `NLTK` library, and the remaining tokens were lemmatized using `SpaCy`.[4]
   - (b) The second representation relies on **contextual embeddings**, obtained via three different models (*XLM-RoBERTa*, *mDeBERTa*, and *MiniLM*) and subsequently used with various linear classifiers. Preprocessing in this case was kept minimal, as excessive cleaning could have led to a loss of context and useful information; only the reduction of multiple spaces was applied.

3.3. Feature Engineering

This section describes the additional features derived from the available textual data, aimed at enriching the representation of instances beyond the text content alone. The considered features fall into two main groups: (i) features extracted using external pre-trained models, such as sentiment and emotions associated with the text, and (ii) surface features derived directly from the data, capturing structural, stylistic, and grammatical aspects.

3.3.1. Sentiment and Emotion Features

To extract these specific auxiliary features, the Italian and Spanish texts were translated into English, as the models used for sentiment and emotion analysis were trained exclusively on English data.[5] In particular, the `GoogleTranslator` class from the `DeepTranslator` library was used for this step.[6] All models employed for feature extraction were retrieved from the *Hugging Face* platform.

The feature related to **sentiment** was extracted using the *Twitter-RoBERTa Sentiment* model [7], based on the *RoBERTa-base* architecture. The model outputs a probability distribution over three classes (*i.e., Negative, Neutral, Positive*) summing to 1. To ensure that only high-confidence predictions influenced the feature set, a thresholding logic was applied: if the confidence score of the predicted class was below

[4] https://spacy.io/

[5] It is important to note that this translation step was strictly limited to feature engineering; the core classification model relies on the original multilingual text processed via XLM-RoBERTa to preserve culture-specific nuances.

[6] https://pypi.org/project/deep-translator/#google-translate-1

0.5, the sentiment was remapped to *Neutral*, treating low-confidence predictions as non-informative. The results show a higher proportion of texts with positive or neutral sentiment in the minority class (Class 1), while Class 0 exhibits a clear prevalence of negative sentiment.

For **emotion** extraction, two complementary models were used. The first is the *DistilRoBERTa Emotion* model [8], which uses the *DistilRoBERTa* architecture. Trained on datasets including *TweetEval* and *Emotion*, it recognizes six basic emotions according to Ekman's theory (i.e., *Anger*, *Disgust*, *Fear*, *Joy*, *Sadness*, *Surprise*) plus a *Neutral* category. This model, working on a limited number of less ambiguous categories, is generally more stable and precise. It shows that *Anger* and *Disgust* are significantly more frequent in Class 0, while *Joy* is more frequent in Class 1, consistent with the nature of the two classes.

The second model, *RoBERTa-Base GoEmotions*, is based on RoBERTa [9] and fine-tuned on Google's *GoEmotions* dataset [10]. It handles colloquial language and direct opinions and classifies texts into 27 emotional labels, including *admiration*, *amusement*, *anger*, *annoyance*, *approval*, *caring*, *confusion*, *curiosity*, *desire*, *disappointment*, *disapproval*, *disgust*, *embarrassment*, *excitement*, *fear*, *gratitude*, *grief*, *joy*, *love*, *nervousness*, *optimism*, *pride*, *realization*, *relief*, *remorse*, *sadness*, and *surprise*, with a *neutral* category added. Results show higher proportions in Class 1 for *admiration*, *joy*, *love*, *pride*, and *sadness*, while *annoyance*, *anger*, *disapproval*, and *disgust* are more frequent in Class 0. This finer granularity captures detailed emotional aspects, showing that Class 1 is mainly associated with identity and belonging, whereas Class 0 contains more negative or judgment-related emotions.

3.3.2. Surface Features

Regarding the surface **additional features**, the *length of the text* and *lenght of the biography* were included, calculated as the number of tokens obtained via `TweetTokenizer`.[7] Additionally, three binary variables were considered, indicating the presence or absence in the user's biography of the following *emojis*: 🌈, 🏳️‍🌈, and 🏳️‍⚧️. Finally, information about the *dominant pronoun* within the text was also included.

Although these are simple variables, their inclusion aims to provide the model with additional information that may help distinguish more easily between observations belonging to the two classes.

Conversely, variables related to the *presence of hashtags* containing the word "orgullo" and the *frequency of aggressive punctuation* within the text were not included. In the first case, as previously discussed (Section 3.1.3), this information was excluded to avoid biasing the model towards decisions based on single keywords, which could prevent it from capturing the overall context and generalizing effectively. In the second case, aggressive punctuation was not considered because the analyses showed that it was not a discriminating variable: except for very few cases, the majority of texts did not exhibit this characteristic.

3.4. Classification Models

This subsection details the experimental framework, starting with the establishment of a baseline model based on statistical text representations. The analysis then progresses to the evaluation of Transformer-based contextual embeddings, followed by the specific procedures adopted for feature engineering, dimensionality reduction, and final hyperparameter tuning. Crucially, across all configurations, every model was trained with the parameter `class_weight='balanced'` to address the significant class imbalance present in the data.

3.4.1. Baseline Model

As a baseline, it was decided to train a Logistic Regression classifier [11] on a TF-IDF (Term Frequency–Inverse Document Frequency) representation [12] of the textual content. To incorporate contextual information as required for Task B, the user biography was concatenated to the text (separated by a space) to create a single input string for feature calculation.

[7] https://www.nltk.org/api/nltk.tokenize.casual.html

Since this step serves as a simple starting point, standard parameters were used: unigrams, removal of rare terms (present in fewer than 5 documents) and highly frequent terms (present in more than 90% of documents), and a maximum limit of 5,000 features. This ensures a balance between the information used and model simplicity.

As described in Section 3.2, a noise-reduction preprocessing phase was required to obtain this representation: converting text to lowercase, removing URLs and mentions, deleting the '#' symbol from hashtags, reducing multiple spaces, removing stopwords, and performing lemmatization.

As a final step, the TF-IDF vectors were used as features to train a Logistic Regression classifier, with *class_weight* setted to *balanced* in order to mitigate the class imbalance issue, which was then applied to classify each observation into one of the two classes. The classifier was evaluated using 10-fold cross-validation to better estimate its generalization performance.

This approach provides a simple, fast, and interpretable baseline, which is essential for a clear comparison with subsequent models based on contextual embeddings.

3.4.2. Contextual Embeddings

In this phase, different contextual embeddings, obtained via three pre-trained models, were tested on three linear classifiers: Logistic Regression, Linear SVM [13], and Ridge Classifier [14]. The models used to extract the embeddings were *mDeBERTa*, *XLM-RoBERTa*, and *MiniLM*, all belonging to the Transformer family.

The choice of these models was driven by the desire to compare solutions with different characteristics, all capable of producing contextual embeddings. *XLM-RoBERTa* and *mDeBERTa* are "strong" multilingual models, often used as benchmarks in multi-language scenarios, and are thus suitable for handling a dataset that includes Italian, Spanish, and English without having to train a separate model for each language. Conversely, *MiniLM* was selected as a more compact and faster alternative to evaluate whether a lighter model could still capture enough context to distinguish between reclaimed and non-reclaimed use of terms.

The goal of this phase was to identify which of the three textual representations performed best on the available data, in order to focus on a single one for subsequent improvement steps. For this reason, tests were conducted using exclusively embeddings extracted from the text (`text` column), excluding biography information to evaluate the potential of the three representations more clearly. Furthermore, the representations were obtained by applying *mean pooling* to the contextual token vectors; this choice allowed for a simple and stable single text representation without capturing overly specific features, making the comparison between models more direct.

In this comparison, the embedding extracted with *XLM-RoBERTa* achieved systematically better results than the other two, regardless of the linear classifier used, based on a 10-fold cross-validation evaluation.

3.4.3. XLM-RoBERTa Representation

After observing that the representation extracted via *XLM-RoBERTa* achieved better results with all tested classifiers, even when limited to text alone, the data representation was enriched. First, the embedding representation was modified by concatenating the *mean pooling* vector with the *max pooling* vector. The rationale is that these two methods capture different and partially complementary information: *mean pooling* summarizes the average text content and provides a more stable representation, while *max pooling* highlights dimensions where strong signals appear, often linked to particularly informative tokens.

Subsequently, the user biography representation was added, using the same procedure as for the text (*XLM-RoBERTa* plus concatenation of *mean pooling* and *max pooling*). This step is central to Task B , as it allows for the inclusion of the author's context, thereby integrating the information provided in the text more effectively.

Finally, to further increase the signals available to the classifier, features obtained from the feature engineering phase were concatenated. Features extracted through pre-trained models regarding sentiment (*Twitter-RoBERTa Sentiment*) and emotion (*DistilRoBERTa Emotion* and *RoBERTa-Base GoEmotions*) were introduced to add high-level information that the model can exploit directly. Manually constructed features included token counts for both text and biography, variables indicating the presence in the biography of emojis strongly associated with the reclaimed class (🌈, 🏳️‍🌈, and 🏳️‍⚧️), and information about the dominant pronoun in the text. These features were intended to add signals related to the structure, grammar, and style of the data.

A transformation was applied to all these variables to make them directly usable by the model. Specifically, continuous variables were normalized within the $[0, 1]$ interval, categorical variables were coded using *one-hot encoding*, and binary variables were left unchanged.

3.4.4. Feature Selection and Hyperparameter Tuning

After defining the representations for text, biography, and additional features, a *feature selection* phase was performed to identify the most informative variables and reduce dimensionality. This step was essential to decrease computational training costs and eliminate redundant or uninformative features, with the aim of improving the model's predictive performance for the binary classification task.

Specifically, the initial feature set was reduced from 3,115 to 800. The 3,115 starting features were composed as follows:

- 1,536 features related to the text and 1,536 related to the user biography. Following the requirements for Task B, which encourages leveraging contextual information, both representations consist of the concatenation of 768 dimensions extracted from *XLM-RoBERTa* via *mean pooling* and 768 via *max pooling*.
- 43 additional features obtained during the feature engineering phase.

The reduction was carried out using *Recursive Feature Elimination* (RFE) [15], utilizing the implementation available in the `scikit-learn` library.[8] This method iteratively removes the least relevant features based on the model's coefficients. As a result, the usefulness of the engineered features—such as emoji indicators—was implicitly evaluated, keeping only those that added significant value to the contextual embeddings.

Since the linear *Support Vector Machine* (SVM) was observed to be the best-performing classifier during this phase, the RFE parameters were also optimized—specifically `n_features_to_select`, representing the final number of features, and `step`, indicating the number of features to remove at each iteration.

Finally, a search for the best hyperparameters for the SVM classifier was conducted, considering both the *linear* and *Radial Basis Function* (RBF) kernels, along with the regularization parameter C. Specifically, values for C of 0.1, 1, 10, and 100 were tested. The experimental results indicated that the best-performing configuration coincided with the default settings: a *linear* kernel and $C = 1$.

3.4.5. Final Training on the Dataset

Up to this point, the various configurations of data representations and classifiers have been evaluated using 10-fold cross-validation on the provided training set. Since a labeled test set was not available for local evaluation during the development phase, as the official test set is released without labels for the challenge assessment, cross-validation was preferred over a standard training/test split. This approach provided a more stable performance estimate and reduced the risk of results being overly dependent on a single data partition.

Consequently, in this final phase, textual representations for all observations were constructed using *XLM-RoBERTa*, concatenating the vectors obtained via *mean pooling* and *max pooling*. The features identified during the feature engineering stage were added to these representations, specifically

[8] https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.RFE.html

incorporating contextual information like biographies as required for Task B. Subsequently, only the features identified during the previous *feature selection* phase were retained to maintain the optimized dimensionality of 800 features.

Finally, the linear SVM classifier was trained on all available observations in the dataset. This final model was used to generate the predictions submitted for the official evaluation of the **MultiPRIDE** challenge.

4. Results

The following section presents the experimental results obtained during the development and evaluation phases. The analysis is divided into two main parts. First, the performance on the labeled training dataset is analyzed to compare different model configurations and select the best approach. Second, the results obtained on the official blind test set are reported, evaluating the final model's effectiveness on unseen data.

4.1. Labeled Dataset

The experiments described in this section were conducted using the labeled training dataset provided by the MultiPRIDE challenge organizers. To rigorously assess model performance and generalization capability during the development phase, a 10-fold cross-validation strategy was applied across all reported configurations.

Table 2 compares the TF-IDF baseline combined with Logistic Regression against various setups based on contextual embeddings from pre-trained models, paired with different classifiers, considering exclusively the text. In this phase, designed to identify the most suitable model for the case study and lacking additional contextual information, the embedding-based solutions do not show an improvement over the baseline.

Table 2
10-fold Cross-validation results for different model configurations. Bold values indicate the best performance.

Model Configuration	F1-score Class 0	F1-score Class 1	Macro F1-score
Baseline: TF-IDF + Logistic Regression	**0.9123**	**0.5690**	**0.7406**
mDeBERTa + Logistic Regression	0.8872	0.5074	0.6973
MiniLM + Logistic Regression	0.8655	0.5186	0.6920
XLM-RoBERTa + Logistic Regression	0.8791	0.5186	0.6988
mDeBERTa + Linear SVM	0.8782	0.4670	0.6726
MiniLM + Linear SVM	0.8853	0.5235	0.7044
XLM-RoBERTa + Linear SVM	0.8551	0.4612	0.6581
mDeBERTa + Ridge Classifier	0.8832	0.4834	0.6833
MiniLM + Ridge Classifier	0.8627	0.4756	0.6691
XLM-RoBERTa + Ridge Classifier	0.8915	0.5371	0.7143

Note: The embeddings used are generated solely via mean pooling and extracted only from the text data.

Table 3 reports the results obtained using the representation based on XLM-RoBERTa (on text and biography), to which the engineered features were concatenated, in combination with various classifiers. Considering the modest performance observed in the previous phase, an improvement over the baseline was not expected at this stage: the overall representation comprises 3,115 features, a high number that can introduce significant noise and redundancy, increase training times, and penalize generalization capabilities, resulting in suboptimal performance.

Table 4 presents the results obtained by applying feature selection via RFE to the same representation described in Table 3. Even with RFE(300), an improvement over the baseline is observed, indicating that

Table 3
10-fold Cross-validation results for different model configurations. Bold values indicate the best performance.

Model Configuration	F1-score Class 0	F1-score Class 1	Macro F1-score
Baseline: TF-IDF + Logistic Regression	**0.9123**	**0.5690**	**0.7406**
XLM-RoBERTa + Logistic Regression	0.9067	0.5257	0.7162
XLM-RoBERTa + Linear SVM	0.8949	0.4236	0.6592
XLM-RoBERTa + Ridge Classifier	0.8902	0.4418	0.6660

Note: The representation is obtained by concatenating the mean- and max-pooled embeddings from the text and biography fields (processed identically), and then appending the engineered features extracted during the feature engineering step.

the previous representation was not uninformative but rather penalized by a high level of noise and redundant features. The three configurations using RFE(300) were used to identify the most suitable classifier for this representation, which proved to be the Linear SVM. Once the classifier was fixed, further experiments showed that the overall best configuration is obtained by selecting 800 features (RFE(800)).

Table 4
10-fold Cross-validation results for different model configurations. Bold values indicate the best performance.

Model Configuration	F1-score Class 0	F1-score Class 1	Macro F1-score
Baseline: TF-IDF + Logistic Regression	0.9123	0.5690	0.7406
XLM-RoBERTa + RFE(300) + Logistic Regression	0.9250	0.6522	0.7886
XLM-RoBERTa + RFE(300) + Linear SVM	0.9428	0.7384	0.8406
XLM-RoBERTa + RFE(300) + Ridge Classifier	0.9371	0.7052	0.8211
XLM-RoBERTa + RFE(800) + Linear SVM	**0.9608**	**0.8019**	**0.8813**

Note: The base representation is the same as in Table 3. Feature selection is performed using RFE (Recursive Feature Elimination); RFE(k) denotes selecting k features.

4.2. Test Set

The results presented in this section were obtained using the optimal configuration identified during the development phase. The final model relies on contextual representations extracted via XLM-RoBERTa from both the text and user biography, aggregating mean-pooled and max-pooled vectors. To optimize performance, Recursive Feature Elimination (RFE) was applied to select the top 800 features. A Linear SVM classifier (regularization parameter $C = 1$) was then trained on the entire available labeled dataset to generate predictions for the blind test set.

Since the test set labels were not available, the evaluation was conducted exclusively based on the metrics provided by the challenge organizers following the submission of predictions.

Table 5
Classification report results for the Italian dataset.

Category	Precision	Recall	F1-Score
Class 0	0.9117	0.9163	0.9140
Class 1	0.6397	0.6258	0.6327
Macro Avg	0.7757	0.7711	0.7733

Table 6
Classification report results for the Spanish dataset.

Category	Precision	Recall	F1-Score
Class 0	0.9050	0.8249	0.8631
Class 1	0.3409	0.5113	0.4090
Macro Avg	0.6229	0.6681	0.6361

The results show a clear difference between the two languages: for observations with Italian text, performance appears generally more stable, whereas for Spanish text, a substantial drop is observed, mainly attributable to greater difficulty in recognizing the minority class.

In both cases, however, performance on the test set is significantly lower than that observed on the available labeled data. This gap suggests that the model overfitted the training set, consequently reducing its ability to generalize to unseen data.

5. Conclusion

In the context of the MultiPRIDE challenge (EVALITA 2026), the test set was unlabeled; consequently, the final evaluation was derived exclusively from the metrics provided by the organizers based on the submitted predictions. Given the class imbalance, Macro F1 was used as the primary metric. The objective of this section is to summarize the experimental process and justify the choice of the final combination of data representation and model.

In the initial phase, a baseline based on TF-IDF and Logistic Regression was defined and subsequently compared with approaches leveraging contextual embeddings extracted from pre-trained models combined with various classifiers (Logistic Regression, Linear SVM, and Ridge Classifier). Considering exclusively the text, and thus without using contextual information (user biography), these setups did not show improvements over the baseline; however, they were useful primarily for guiding the choice of data representation for the current task.

Subsequently, a more comprehensive representation was introduced, obtained by concatenating text and biography representations (mean and max pooling) and integrating features derived from feature engineering. In this configuration, however, the high overall dimensionality amplified noise and redundancy, negatively affecting both training times and performance, which did not surpass the baseline.

Improvement was observed with the introduction of feature selection via Recursive Feature Elimination (RFE): even with an initial selection of features, results superior to the baseline were achieved, indicating that the extended representation was informative but penalized by the presence of many non-useful features. Comparisons between classifiers in the RFE configuration highlighted the Linear SVM as the most effective choice; once the classifier was fixed, further experiments showed that the absolute best configuration is obtained by selecting 800 features.

Finally, a drop in performance is observed on the test set compared to estimates obtained via cross-validation on the available labeled data, signaling weak generalization and thus overfitting. Furthermore, a difference between languages emerges: results are generally more stable on Italian texts, while on Spanish texts the decline is more marked, especially for the minority class. Since test set labels are not available, an in-depth error analysis cannot be performed; however, the gap between validation and test results may be linked to differences between training and test data, to an overly aggressive optimization of Macro F1 on the available data, which may reduce robustness on unseen data, or to the fact that the multilingual model was unable to achieve an equally high level of generalization across different languages, specializing better in one than in the others.

5.1. Identified Limitations

A first limitation concerns the quantity of available data. After removing duplicates, the training set contained 2,983 total samples distributed across the three languages (Italian, Spanish, and English). Consequently, the number of examples per single language was relatively low, with a potential impact on the model's ability to generalize.

A second limitation is related to finding the configuration that maximized the Macro F1 score on the available data. Although useful for quickly comparing many solutions, this approach may have favored choices that were too "fitted" to the training set, increasing the risk of overfitting and reducing robustness on unseen data.

Furthermore, the evaluation was conducted by considering all languages together, without systematically analyzing performance language-by-language during model selection. This made it harder to detect potential imbalances between languages in advance, which instead emerged clearly during the testing phase (for example, the more marked drop in performance for Spanish).

Finally, the presence of English in the training phase may have introduced meaning shifts or information loss: the model was trained in a multilingual setting on three languages, but the final evaluation concerned only two languages. Additionally, the English texts did not include the biography. This misalignment (same model, but different information depending on the language) could have added noise and made learning less effective, especially for linear classifiers.

5.2. Future Developments

First of all, it would be useful to directly address the identified limitations: increasing evaluation robustness (e.g., by analyzing performance per language during cross-validation), reducing the risk of overfitting (using more conservative selection and tuning strategies), and managing differences between languages and biography availability more consistently.

A natural second step is to test other multilingual models for extracting contextual embeddings, verifying whether different representations prove to be more stable across languages and more informative for the minority class.

Finally, an interesting approach to explore is zero-shot classification. In this scenario, the model is not directly trained on the task with labeled examples but attempts to classify texts using a class description (or a prompt) and the knowledge already acquired during pre-training. This can be useful when labeled data is scarce or when the goal is to improve cross-lingual generalization without relying too heavily on the specific training set.

Code and Data Availability

For reproducibility purposes, the source code is publicly available at: https://github.com/FedericoDingeo/Multipride-at-EVALITA-2026-Project/. To obtain access to the data, we invite the reader to refer to the guidelines published in [5].

Declaration on Generative AI

During the preparation of this work, the authors employed ChatGPT, Gemini, and Grammarly, to assist with text revision, primarily for grammar and spell checking, paraphrasing, and enhancing the clarity and style of the English writing. Specifically, these tools were utilized to refine the phrasing of certain passages and concepts, and to assist in translating ideas into clearer English formulations. All outputs produced by these tools were carefully reviewed and, where necessary, edited by the authors, who take full responsibility for the final content of the paper.

References

[1] Y. Goldberg, Neural Network Methods for Natural Language Processing, Morgan & Claypool Publishers, 2017.

[2] R. Brontsema, A queer revolution: Reconceptualizing the debate over linguistic reclamation, Colorado Research in Linguistics 17 (2004) 1–17. URL: https://journals.colorado.edu/index.php/cril/article/view/238.

[3] L. Draetta, C. Ferrando, M. Cuccarini, L. James, V. Patti, ReCLAIM project: Exploring Italian slurs reappropriation with large language models, in: F. Dell'Orletta, A. Lenci, S. Montemagni, R. Sprugnoli (Eds.), Proceedings of the Tenth Italian Conference on Computational Linguistics (CLiC-it 2024), CEUR Workshop Proceedings, Pisa, Italy, 2024, pp. 335–342. URL: https://aclanthology.org/2024.clicit-1.40/.

[4] L. Dixon, J. Li, J. Sorensen, N. Thain, L. Vasserman, Measuring and mitigating unintended bias in text classification, in: Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society, 2018, pp. 67–73. URL: https://doi.org/10.1145/3278721.3278729.

[5] C. Ferrando, L. Draetta, M. Madeddu, M. Sosto, V. Patti, P. Rosso, C. Bosco, J. Mata, E. Gualda, Multipride at evalita 2026: Overview of the multilingual automatic detection of slur reclamation in the lgbtq+ context task, in: Proceedings of the Ninth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2026), CEUR.org, Bari, Italy, 2026.

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

[7] F. Barbieri, J. Camacho-Collados, L. Espinosa Anke, L. Neves, TweetEval: Unified benchmark and comparative evaluation for tweet classification, in: Findings of the Association for Computational Linguistics: EMNLP 2020, Association for Computational Linguistics, Online, 2020, pp. 1644–1650. URL: https://aclanthology.org/2020.findings-emnlp.148. doi:10.18653/v1/2020.findings-emnlp.148.

[8] J. Hartmann, Emotion english distilroberta-base, 2022. URL: https://huggingface.co/j-hartmann/emotion-english-distilroberta-base/.

[9] 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). URL: https://arxiv.org/abs/1907.11692.

[10] D. Demszky, D. Movshovitz-Attias, J. Ko, A. Cowen, G. Monterso, S. Ravi, GoEmotions: A dataset of fine-grained emotions, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 4040–4054. URL: https://aclanthology.org/2020.acl-main.372/.

[11] D. R. Cox, The regression analysis of binary sequences, Journal of the Royal Statistical Society: Series B (Methodological) 20 (1958) 215–232.

[12] K. Spärck Jones, A statistical interpretation of term specificity and its application in retrieval, Journal of documentation 28 (1972) 11–21.

[13] C. Cortes, V. Vapnik, Support-vector networks, Machine learning 20 (1995) 273–297.

[14] A. E. Hoerl, R. W. Kennard, Ridge regression: Biased estimation for nonorthogonal problems, Technometrics 12 (1970) 55–67.

[15] M. Kuhn, K. Johnson, Applied Predictive Modeling, Springer, New York, 2013.