XY-UP at SSD-2026: Hierarchical Social Support Detection in Spanish Social Media Using Classical and Transformer-Based Approaches

Ximena Denise Macias Mateos[1], Yael Emiliano Salinas Lozano[1]

[1]*Universidad Panamericana (UP), Mexico City, Mexico*

Abstract

Social support detection is an emerging natural language processing task focused on identifying expressions of encouragement, empathy, solidarity, and care in online discourse. In this work, we address social support detection in Spanish social media comments using the datasets provided by the Social Support Detection shared task (SSD-2026). Our system follows a hierarchical formulation composed of three subtasks: (1) detecting whether a comment expresses support (2,600 samples), (2) identifying whether that support is directed toward an individual or a group (590 samples), and (3) classifying the specific target group when the support is group-directed, among six categories—Black Community, LGBTQ, Nation, Religion, Women, and Other (485 samples). We compare two modeling paths. The first relies on unigram TF-IDF representations and traditional machine learning classifiers, including Logistic Regression, Support Vector Machines, Decision Trees, Random Forests, and voting ensembles. The second fine-tunes BETO [1], a pre-trained Spanish transformer model. All training experiments were evaluated using 5-fold stratified cross-validation, with macro F1-score as the main evaluation metric. In the training phase, BETO achieved macro F1-scores of 0.8446 (Task 1), 0.7977 (Task 2), and 0.7812 (Task 3), outperforming the strongest classical baselines in all subtasks. On the official test set, our system obtained macro F1-scores of 0.8750 (Task 1: Support vs. Non-Support), 0.7698 (Task 2: Individual vs. Group), and 0.5881 (Task 3: Target Group Identification). These results suggest that contextualized representations are particularly effective for capturing supportive language in Spanish social media, while fine-grained group classification remains a challenging open problem.

Keywords

social support detection, social media, Spanish NLP, text classification, BETO, TF-IDF, transformer models, hierarchical classification

1. Introduction

Social media platforms have become important spaces for interaction, emotional expression, and community support. Beyond opinion sharing or sentiment expression, many online comments communicate empathy, encouragement, solidarity, admiration, or validation toward other people and communities. Automatically identifying these supportive expressions is a relevant challenge for natural language processing because it expands the scope of text classification beyond polarity-based sentiment analysis and toward the understanding of prosocial language.

This work focuses on social support detection in Spanish, using the data released for the Social Support Detection shared task (SSD-2026) [2, 3, 4, 5, 6]. Rather than framing the problem as a single classification decision, we address it as a hierarchical task composed of three subtasks. The first subtask identifies whether a comment expresses support or not (Subtask 1; 2,600 samples). The second subtask determines whether the support is directed toward an individual or a group (Subtask 2; 590 samples). The third subtask specializes the group-directed support case by predicting the specific target group from six categories—Black Community, LGBTQ, Nation, Religion, Women, and Other (Subtask 3; 485 samples).

This paper describes the system that team XY-UP submitted to the SSD-2026 shared task. We compare two complementary approaches: a classical machine learning pipeline based on TF-IDF features and a transformer-based pipeline based on BETO [1]. For each subtask we report both *validation* results,

IberLEF 2026, September 2026, León, Spain

ximena.macias@up.edu.mx (X. D. M. Mateos); yael.salinas@up.edu.mx (Y. E. S. Lozano)

obtained through 5-fold stratified cross-validation on the released training data, and the *official test* results returned by the shared task evaluation platform on the held-out test set. The validation phase is used to select the modeling strategy for each subtask, and the selected models are then used to produce the official test submission.

The main research questions that guide this work are the following:

1. Can classical TF-IDF-based classifiers capture lexical patterns distinctive of supportive language in Spanish social media, and which family of models generalizes best across the three subtasks?
2. Does fine-tuning a pre-trained Spanish transformer model (BETO) yield consistent improvements over classical baselines, and by what margin for each subtask?
3. How well does the hierarchical pipeline generalize to unseen data, and where does performance degrade most as the classification becomes more fine-grained?

2. Related Work

Social support detection in online communities sits at the intersection of several active research directions in NLP: prosocial language analysis, sentiment and emotion detection, and the classification of hate or hope speech. This section reviews the most directly relevant lines of prior work.

Social support detection. The task of detecting social support in social media was formally introduced as an NLP problem by Ahani et al. [4], who defined three hierarchical subtasks—distinguishing supportive from non-supportive comments, identifying the target (individual vs. group), and classifying the specific group—and released a dataset of 10,000 YouTube comments annotated with these labels. Their experiments combined linguistic, psycholinguistic, and emotional features with traditional machine learning and neural network models. A follow-up study by Kolesnikova et al. [6] extended this line of research using state-of-the-art transformer models, including RoBERTa and GPT-4o, as well as K-means clustering to address class imbalance. Their best transformer-based system achieved macro F1-scores of 0.78, 0.84, and 0.80 across the three subtasks on the original English dataset, providing an important reference point for comparison. The Spanish extension of this task, which constitutes our target setting, is described by Tash et al. [5]. A complementary linguistic analysis of support categories using LIWC across Spanish and English communities is presented in [7].

Hope speech detection in Spanish. Closely related to social support is the problem of hope speech detection, which focuses on identifying encouraging and positive messages. García-Baena et al. [8] presented a comprehensive study on automated hope speech detection in Spanish, building a dataset of LGBT-related tweets collected from Twitter and evaluating traditional machine learning, deep learning, and transformer-based baselines. Their findings suggest that transformer models are especially effective for capturing the nuanced linguistic features of hopeful language in Spanish, a conclusion that aligns with our own results for supportive language. The broader multilingual perspective on hope speech is explored by additional recent work that highlights how transfer learning approaches substantially improve performance across languages.

Hate speech detection in Spanish. Although hate speech detection addresses the opposite pole of prosocial discourse, its methodology is directly relevant because many of the same modeling choices—TF-IDF baselines, fine-tuned transformer models, and Spanish-specific pretrained models—are common to both tasks. Research on Spanish hate speech detection has consistently found that monolingual transformer models outperform multilingual counterparts and classical methods [8]. The application of ALBETO, a compact Spanish BERT variant, to hate speech detection with the HatEval benchmark further corroborates this trend.

Spanish language models. Our transformer-based path relies on BETO [1], the first BERT model pre-trained exclusively on a large Spanish corpus using Whole Word Masking. Cañete et al. showed that BETO consistently outperforms Multilingual BERT on a range of Spanish NLP benchmarks, motivating its adoption in downstream classification tasks including sentiment analysis, named entity recognition, and, more recently, social media content classification. The advantage of monolingual pretraining for Spanish tasks is also documented in the requirements classification literature, where BETO surpasses classical shallow models when evaluated on out-of-domain data.

Classical vs. transformer approaches. A recurring finding in the social media classification literature is that classical TF-IDF-based models, particularly linear SVMs, remain strong baselines that are competitive with or sometimes superior to neural approaches when the training set is small [4, 6]. Our work confirms and extends this finding for Spanish: SVM-Linear is the strongest classical model in Tasks 1 and 2, while Soft Voting prevails in Task 3, and BETO improves consistently over all classical baselines but with varying margins depending on dataset size and task complexity.

3. Definitions

3.1. Social Support

For the purposes of this study, *social support* refers to textual expressions that communicate encouragement, care, empathy, admiration, validation, solidarity, or any other form of prosocial reinforcement toward a person or a community. In the context of social media, this support may appear through direct encouragement, recognition of someone's effort or suffering, defense of a group, or emotional reassurance in response to a personal or collective situation.

By contrast, a comment is considered *non-supportive* when it does not convey supportive intent. This category may include neutral statements, irrelevant content, opinions without encouragement, dismissive language, or comments that do not provide any meaningful emotional or social reinforcement.

This distinction is especially important because supportive discourse cannot be reduced to positive sentiment alone. A comment may contain positive words without actually supporting anyone, while a genuinely supportive message may depend on context, target, and communicative intent.

4. Dataset Development

4.1. Dataset Source

Regarding the dataset development, the materials used in this work were provided by the Social Support Detection shared task (SSD-2026), meaning that both the data collection and the original annotation process were external to the research team. The dataset was originally introduced and described in [4, 5, 6, 7]. We used the released Spanish training set to build and compare the classification models, and submitted predictions on the official test set for evaluation. To carry out the experiments, the information was managed in two parallel formats: a cleaned version prepared for classical modeling based on TF-IDF and another version optimized for training with transformer-based architectures.

4.2. Dataset Composition

The Spanish training dataset comprises a total of 2,600 records, from which the hierarchical subtasks are constructed. In Task 1, all 2,600 comments are utilized to classify instances as Support versus Non-Support. Task 2 focuses exclusively on comments labeled as Support, resulting in a subset of 590 records. Finally, Task 3 is limited to comments identified as Group in the previous stage, consisting of 485 records. This hierarchical reduction is a direct consequence of the second and third subtasks depending on the labels assigned in the preceding stages.

4.3. Subtasks

4.3.1. Subtask 1: Support vs. Non-Support (2,600 samples)

Subtask 1 represents the initial phase of the hierarchical classification system, functioning as the entry point for the entire pipeline. The primary objective of this task is to identify whether a comment expresses social support. Within this framework, a comment is classified as *Support* if it contains elements of encouragement, empathy, solidarity, care, validation, or any other form of prosocial reinforcement. Conversely, a comment is labeled as *Non-Support* if it fails to communicate any supportive intent. As the broadest task in the sequence, it filters the data necessary for subsequent, more specific classification stages.

4.3.2. Subtask 2: Individual vs. Group (590 samples)

The second task is applied exclusively to the 590 comments identified as supportive, with the primary objective of determining the specific target type of the support. Within this stage, a comment is classified as *Individual* if the support is directed toward a specific person or a clearly identifiable individual. Conversely, it is labeled as *Group* if the support is directed toward a collective entity, such as a social community, demographic group, or broader category of people. This task refines the set of supportive comments by distinguishing whether the expression of care is personalized or collective in nature.

4.3.3. Subtask 3: Target Group Identification (485 samples)

The third task is executed exclusively on the 485 supportive comments that have been identified as being directed toward a collective entity. The objective of this final stage is to predict the specific target group from a predefined set of six categories:

- **Black Community**: comments that express support toward Afro-descendant communities or individuals racialized as Black.
- **LGBTQ**: comments that express support toward lesbian, gay, bisexual, transgender, queer, or non-binary individuals and communities.
- **Nation**: comments that express support toward a country, nationality, or national identity.
- **Religion**: comments that express support toward a religious group, faith community, or spiritual belief.
- **Women**: comments that express support toward women as a social group or gender category.
- **Other**: comments that express group-directed support not covered by the categories above.

This granular classification step allows for a more fine-grained understanding of group-directed support by identifying the specific social or demographic community receiving the prosocial reinforcement.

4.4. Dataset Statistics

Table 1
Dataset statistics per subtask

Subtask	Labels	Samples
Task 1	Non Support / Support	2,600
Task 2	Individual / Group	590
Task 3	Black Community / LGBTQ / Nation / Other / Religion / Women	485

The three subtasks are markedly imbalanced. In Task 1, only 590 of the 2,600 comments (22.7%) are labeled as Support. In Task 2, the Group class dominates (456 comments) over the Individual class (134). The imbalance is most severe in Task 3, where the six target groups are distributed very unevenly:

Table 2 reports the per-class sample counts. LGBTQ is by far the most frequent group, accounting for almost half of the examples, whereas Black Community is represented by only 24 comments. This distribution is directly relevant to the results discussion, since the scarcity of minority groups constrains how well the model can learn them.

Table 2
Task 3 sample distribution per target group (485 comments).

Target group	Samples
LGBTQ	224
Other	82
Religion	60
Women	48
Nation	47
Black Community	24
Total	485

4.5. Samples

Table 3 shows representative examples from the dataset along with their hierarchical labels.

Table 3
Example comments with their hierarchical labels

Comment	Task 1	Task 2	Task 3
Ahora veo que la cuestión del género es más cosa como de personalidad y costumbres, ¿no?... Yo creía que era hetero ...	Support	Group	LGBTQ
claro la religion gana porque DIOS es vida fuerza y fortaleza asi que para la loca safada agatha DIOS existe	Support	Group	Religion

5. Methodology

5.1. Models

The training phase compares two different modeling paths. **Path A** focuses on classical machine learning, using unigram TF-IDF [9] vectors as text representation to evaluate various classifiers, including Logistic Regression (LR) [10], Support Vector Machines [11] with RBF (SVM-RBF) and linear (SVM-Linear) kernels, Decision Trees (DT), Random Forest Classifiers (RFC) [12], and soft and hard voting ensembles [13]; this serves as a strong lexical baseline and allows for comparison across linear, non-linear, tree-based, and ensemble methods. All classical models were implemented with scikit-learn [14]. **Path B** consists of transformer-based [15, 16] fine-tuning using BETO [1], a Spanish pre-trained transformer model that enables the use of contextualized representations instead of sparse lexical vectors, fine-tuned with the Hugging Face Transformers library [17].

5.2. Preprocessing

Rather than applying a single uniform cleaning strategy, we use a shared normalization stage and then split the data into two branches aligned with the needs of the downstream models. We first load the

raw Spanish training file, inspect it for missing values and duplicates, and remove duplicate comments (using the comment text as the key) so that repeated entries do not inflate the frequency of specific lexical patterns.

Shared normalization. The shared stage standardizes noisy social-media phenomena while preserving semantic content, and is applied in four steps. (1) *Emoji conversion*: emojis are converted into textual descriptors with the `emoji` library's `demojize` function (e.g., a flexed-biceps emoji becomes a lexicalized token). (2) *Emoticon normalization*: textual emoticons such as `:)`, `:(`, `:D`, `xd`, `<3`, and `</3` are mapped to descriptive labels (*cara_feliz*, *cara_triste*, *cara_riendo*, *corazón*, *corazón_roto*) via explicit string-replacement rules. (3) *Abbreviation and slang expansion*: Spanish abbreviations and YouTube-style slang (*xq*, *pq*, *xfa*, *tb*, *tmb*, *q*, *pa*, *ntp*, *ns*, *tqm*, *tkm*, *bro*, *info*, etc.) are expanded through a manual dictionary. (4) *General normalization*: URLs and user mentions are replaced with `[URL]` and `[USER]` placeholders; zero-width and formatting characters are removed; repeated characters are capped at three; runs of punctuation are reduced to two; laughter patterns (*jajaja*, *jejeje*, *jsjsjs*, repeated *xd*) are canonicalized; and newlines and multiple spaces are collapsed to single spaces.

Branch-specific preprocessing. For the classical TF-IDF path we apply a more aggressive normalization on top of the shared text: lowercasing, punctuation removal (preserving the `[URL]`/`[USER]` placeholders), whitespace tokenization, Spanish stopword removal with NLTK [18], and stemming with the Spanish SnowballStemmer, yielding a compact and consistent feature space. For the transformer path we keep the normalized text as-is—without lowercasing, stopword removal, punctuation stripping, or stemming—so that casing, punctuation, accents, and function words remain available as contextual cues, closer to BETO's pretraining conditions.

5.3. Model Training and Predictions

5.3.1. Classical Models

All classical models followed the same protocol across the three subtasks: a Stratified 5-Fold Cross-Validation scheme with shuffling and a fixed seed (42) to preserve class proportions and ensure reproducibility. Within each fold the training text was converted into TF-IDF vectors (unigram, `max_features=10000`, `ngram_range=(1,1)`, `sublinear_tf=True`); the vectorizer was fitted only on the training fold and applied to the test fold to prevent leakage. The compared models were Logistic Regression (`max_iter=1000`), SVM-RBF, SVM-Linear (`LinearSVC`, `max_iter=2000`), Decision Tree, Random Forest (`n_estimators=100`), and soft/hard voting ensembles combining LR, linear SVM, and RFC. All other hyperparameters were left at their scikit-learn defaults, and no additional class-imbalance handling was applied. Models were evaluated with macro precision/recall/F1, weighted F1, and accuracy, with macro F1 as the main selection criterion.

5.3.2. Classical Training: Task 1 — Support vs. Non-Support

In Task 1, all 2,600 training instances were used. Each model learned to discriminate between Support and Non-Support using the TF-IDF representation of the input comments. Since this task is the entry point of the pipeline, performance here is particularly important: errors at this stage prevent downstream classification in Tasks 2 and 3.

Among the classical models, SVM-Linear obtained the highest macro F1-score, indicating that a linear decision boundary over sparse lexical features was highly effective for detecting supportive language. Voting ensembles also performed strongly, particularly in weighted F1 and accuracy, suggesting that the lexical cues for supportive discourse are relatively stable and can be captured by several complementary classifiers. Table 4 reports the full results.

Table 4
Task 1 — Classical model results (5-fold cross-validation, mean ± std). Best macro F1 in bold.

Model	Macro Prec.	Macro Rec.	Macro F1	Weighted F1	Acc.
LR	0.8617 (±0.0256)	0.6949 (±0.0221)	0.7332 (±0.0262)	0.8305 (±0.0159)	0.8523 (±0.0126)
SVM-RBF	0.8610 (±0.0230)	0.7127 (±0.0266)	0.7508 (±0.0286)	0.8402 (±0.0172)	0.8585 (±0.0135)
SVM-Linear	0.8223 (±0.0228)	0.7679 (±0.0219)	**0.7894** (±0.0221)	0.8579 (±0.0145)	0.8642 (±0.0134)
DT	0.7549 (±0.0111)	0.7476 (±0.0115)	0.7506 (±0.0093)	0.8261 (±0.0066)	0.8273 (±0.0074)
RFC	0.8428 (±0.0264)	0.7247 (±0.0269)	0.7594 (±0.0287)	0.8437 (±0.0177)	0.8585 (±0.0147)
Soft Voting	0.8542 (±0.0260)	0.7427 (±0.0176)	0.7778 (±0.0199)	0.8547 (±0.0129)	0.8669 (±0.0118)
Hard Voting	0.8571 (±0.0245)	0.7297 (±0.0222)	0.7667 (±0.0241)	0.8487 (±0.0149)	0.8635 (±0.0126)

5.3.3. Classical Training: Task 2 — Individual vs. Group

Task 2 was trained on the 590 supportive comments, distinguishing Individual from Group support. This is a subtler problem than Task 1 since both classes are supportive and share affective and lexical patterns. SVM-Linear again achieved the highest macro F1 (Table 5); Soft Voting matched its accuracy but with a lower, less balanced macro F1.

Table 5
Task 2 — Classical model results (5-fold cross-validation, mean ± std). Best macro F1 in bold.

Model	Macro Prec.	Macro Rec.	Macro F1	Weighted F1	Acc.
LR	0.8339 (±0.0574)	0.5635 (±0.0314)	0.5546 (±0.0520)	0.7346 (±0.0249)	0.7983 (±0.0125)
SVM-RBF	0.8281 (±0.0662)	0.5637 (±0.0248)	0.5561 (±0.0407)	0.7352 (±0.0203)	0.7983 (±0.0112)
SVM-Linear	0.7959 (±0.0729)	0.6915 (±0.0376)	**0.7172** (±0.0395)	0.8155 (±0.0249)	0.8322 (±0.0230)
DT	0.6678 (±0.0456)	0.6562 (±0.0291)	0.6606 (±0.0360)	0.7638 (±0.0291)	0.7661 (±0.0337)
RFC	0.7930 (±0.1000)	0.6313 (±0.0410)	0.6555 (±0.0539)	0.7842 (±0.0350)	0.8169 (±0.0306)
Soft Voting	0.8393 (±0.0681)	0.6545 (±0.0223)	0.6860 (±0.0300)	0.8028 (±0.0214)	0.8322 (±0.0203)
Hard Voting	0.8167 (±0.1033)	0.6183 (±0.0392)	0.6396 (±0.0539)	0.7775 (±0.0345)	0.8169 (±0.0287)

5.3.4. Classical Training: Task 3 — Target Group Identification

Task 3 was trained on the 485 Group-directed comments, the most fine-grained stage, where the model predicts one of six target groups (Black Community, LGBTQ, Nation, Other, Religion, Women), using the same per-fold TF-IDF procedure. Soft Voting produced the best macro F1 (0.7474 ±0.0146), with SVM-Linear close behind (0.7383 ±0.0377); full results are in Table 6.

Table 6
Task 3 — Classical model results (5-fold cross-validation, mean ± std). Best macro F1 in bold.

Model	Macro Prec.	Macro Rec.	Macro F1	Weighted F1	Acc.
LR	0.6973 (±0.0991)	0.5365 (±0.0212)	0.5713 (±0.0320)	0.6939 (±0.0190)	0.7320 (±0.0146)
SVM-RBF	0.6556 (±0.0406)	0.4835 (±0.0179)	0.5188 (±0.0211)	0.6523 (±0.0162)	0.6969 (±0.0140)
SVM-Linear	0.8094 (±0.0411)	0.7069 (±0.0297)	0.7383 (±0.0377)	0.7952 (±0.0238)	0.8062 (±0.0230)
DT	0.6529 (±0.0277)	0.6209 (±0.0257)	0.6249 (±0.0227)	0.6769 (±0.0308)	0.6825 (±0.0294)
RFC	0.8035 (±0.0300)	0.6410 (±0.0324)	0.6808 (±0.0321)	0.7493 (±0.0260)	0.7732 (±0.0226)
Soft Voting	0.8144 (±0.0138)	0.7148 (±0.0200)	**0.7474** (±0.0146)	0.7982 (±0.0114)	0.8082 (±0.0105)
Hard Voting	0.8334 (±0.0498)	0.6159 (±0.0143)	0.6695 (±0.0191)	0.7419 (±0.0097)	0.7649 (±0.0101)

5.3.5. Transformer-Based Training and Predictions

BETO Training Procedure. For the transformer-based path, we fine-tuned BETO (`dccuchile/bert-base-spanish-wwm-cased`) [1] independently for each subtask, using contextualized representations instead of sparse TF-IDF vectors. We preserved the same gold-filtered hierarchy and the same Stratified 5-Fold Cross-Validation protocol as the classical path, keeping the two setups directly comparable.

Hyperparameters. Since the code is not released, we report the full fine-tuning configuration so that the experiments can be reproduced. All three subtasks used the same hyperparameters, summarized in Table 7. BETO was fine-tuned for 8 epochs with the AdamW optimizer, a learning rate of 2×10^{-5}, a per-device batch size of 16 with gradient accumulation of 2 (effective batch size of 32), a maximum sequence length of 128 tokens, weight decay of 0.01, a warmup ratio of 0.15, and label smoothing of 0.1. Mixed-precision (`fp16`) training was enabled when a GPU was available. At the end of every epoch the model was evaluated on the held-out fold, and the checkpoint with the best macro F1 was retained (`load_best_model_at_end` with `metric_for_best_model` set to macro F1 and early stopping patience of 2). A fixed random seed of 42 was used for NumPy, PyTorch, the cross-validation splits, and the trainer to ensure reproducibility.

Table 7
BETO fine-tuning hyperparameters (identical for all subtasks).

Hyperparameter	Value
Base model	`bert-base-spanish-wwm-cased`
Optimizer	AdamW
Learning rate	2×10^{-5}
Epochs	8
Batch size (per device)	16
Gradient accumulation	2 (effective 32)
Max sequence length	128
Weight decay	0.01
Warmup ratio	0.15
Label smoothing	0.1
Early stopping patience	2
Mixed precision	fp16 (if GPU)
Random seed	42

Class imbalance. Given the strong class imbalance documented in Table 2, the BETO path explicitly compensates for it during training. We replaced the standard cross-entropy loss with a class-weighted cross-entropy, where the per-class weights were computed with scikit-learn's `compute_class_weight` in `balanced` mode (inversely proportional to class frequency) and combined with label smoothing of 0.1. No resampling, oversampling, or synthetic data generation (e.g., SMOTE or K-means-based augmentation) was applied. The classical TF-IDF path did not use any additional imbalance-handling strategy beyond the macro-averaged selection criterion.

Final prediction strategy. Cross-validation was used only for model selection. To generate the official test submission, we did *not* ensemble or average the five folds, nor retrain on the full training set; instead, for each subtask we kept the single fold checkpoint that obtained the best macro F1 during cross-validation and used it to predict the test comments. The way these per-subtask predictions are combined into the final hierarchical submission is described in Section 5.5.

Cross-validation results. BETO was fine-tuned on the full 2,600-comment set for Task 1, on the 590 supportive comments for Task 2, and on the 485 group-directed comments for Task 3, following the

same gold-filtered hierarchy as the classical path. Table 8 summarizes the cross-validation results across all three subtasks. BETO reaches a mean macro F1 of 0.8446 (±0.0049) on Task 1, 0.7977 (±0.0185) on Task 2, and 0.7812 (±0.0421) on Task 3, improving over the best classical baseline in every subtask.

Table 8
BETO cross-validation results (5-fold, mean ± std).

Metric	Task 1	Task 2	Task 3
Macro Precision	0.8581 (±0.0206)	0.8094 (±0.0338)	0.8449 (±0.0227)
Macro Recall	0.8361 (±0.0151)	0.7917 (±0.0191)	0.7637 (±0.0482)
Macro F1	**0.8446** (±0.0049)	**0.7977** (±0.0185)	**0.7812** (±0.0421)
Weighted F1	0.8923 (±0.0043)	0.8594 (±0.0135)	0.8449 (±0.0224)
Accuracy	0.8938 (±0.0061)	0.8610 (±0.0157)	0.8515 (±0.0191)

5.4. Best-Performing Model per Task

To summarize model selection across both paths, Table 9 compares the strongest classical baseline against BETO for each subtask. In all three cases, the transformer-based path achieved the highest macro F1-score, although the margin varied depending on the complexity of the task.

Table 9
Best classical model vs. BETO per subtask (macro F1, 5-fold CV).

Task	Best Classical	Macro F1	BETO Macro F1
Task 1	SVM-Linear	0.7894	**0.8446**
Task 2	SVM-Linear	0.7172	**0.7977**
Task 3	Soft Voting	0.7474	**0.7812**

5.5. Official Test Phase Results

Evaluation setup. It is important to distinguish how the subtasks are organized during training from how they are evaluated on the official test set, as the two differ. During *training and cross-validation*, each subtask is learned independently on a *gold-filtered* subset: Task 2 is trained only on comments whose gold Task 1 label is Support, and Task 3 only on comments whose gold Task 2 label is Group. This isolates the difficulty of each subtask and yields the validation scores reported above. For the *official test submission*, however, gold labels are not available, so predictions are produced through an *end-to-end routed pipeline*: the Task 1 model first predicts Support vs. Non-Support; only comments predicted as Support are routed to the Task 2 model; and only comments predicted as Group (by Task 2) are routed to the Task 3 model. Consequently, the official Task 2 and Task 3 scores already incorporate any error propagated from the upstream stages, and should be read as end-to-end pipeline results rather than as isolated per-subtask results.

After submitting our predictions to the shared task evaluation platform, the system was scored on the official held-out test set. The results obtained during the testing phase are reported in Table 10.

The system achieved strong performance on Task 1, with a macro F1-score of 0.8750 on the test set, which is even above the cross-validation estimate of 0.8446 and suggests good generalization. Task 2 also yielded competitive results (macro F1 = 0.7698), close to the cross-validation score of 0.7977. Task 3, however, showed a more pronounced drop from cross-validation (0.7812) to test (0.5881), indicating

Table 10
Official test-set results for the three subtasks.

Task	Macro Prec.	Macro Rec.	Macro F1	Weighted F1	Acc.
Task 1: Support vs. Non-Support	0.8716	0.8785	**0.8750**	0.9113	0.9109
Task 2: Individual vs. Group	0.7909	0.7545	**0.7698**	0.8420	0.8471
Task 3: Target Group Identification	0.6868	0.5714	**0.5881**	0.6898	0.6116

that the fine-grained group classification remains the most challenging subtask in the pipeline and that the model does not generalize as well to unseen group-directed comments.

6. Error Analysis

A detailed error analysis ideally requires inspecting the confusion matrices produced by the best-performing model (BETO) for each subtask. The official test-set gold labels are not released, so a confusion matrix on the official test set cannot be computed. The only confusion matrix we are able to report is for Task 1, computed on a held-out validation fold (520 comments) from the training data; the remaining observations are based on the aggregate precision/recall figures returned by the official scoring and are therefore stated as hypotheses rather than confirmed error patterns.

Task 1 — Support vs. Non-Support. This subtask shows the strongest performance on both cross-validation (macro F1 = 0.8446) and the official test set (macro F1 = 0.8750). On the validation fold (Table 11), the model classifies the majority Non-Support class very accurately (recall 0.943) but recovers the minority Support class less reliably (recall 0.686), i.e., it misses some genuinely supportive comments. This is consistent with the observation that the boundary between polite or agreeable language and genuine support can be difficult to capture from surface form alone. On the official test set, recall is marginally higher than precision (0.8785 vs. 0.8716).

Table 11
Task 1 confusion matrix on a held-out validation fold (520 comments). Rows are gold labels, columns are predictions.

Gold \ Pred.	Non-Support	Support
Non-Support	379	23
Support	37	81

Task 2 — Individual vs. Group. The test-set results for Task 2 (macro F1 = 0.7698) fall below the cross-validation estimate (0.7977), with the gap driven primarily by a drop in recall relative to precision (0.7545 vs. 0.7909). The lower macro recall indicates that at least one of the two classes is recovered less reliably than the macro-averaged precision would suggest. Because we do not have an official confusion matrix for this subtask, we refrain from attributing the gap to a specific class. A plausible contributing factor is that individual- and group-directed support share much affective vocabulary, making the distinction hard to learn; in addition, part of the gap is expected from error propagation, since the official Task 2 predictions are conditioned on the Task 1 stage (Section 5.5).

Task 3 — Target Group Identification. The most pronounced generalization gap appears in Task 3, where macro F1 drops from 0.7812 (cross-validation) to 0.5881 (test set). The low macro recall (0.5714) against a higher macro precision (0.6868), together with the gap between weighted F1 (0.6898) and macro F1 (0.5881), indicates that performance is uneven across the six classes and concentrated on the better-represented ones. The class distribution (Table 2) is highly skewed: a single class, LGBTQ,

accounts for almost half of the 485 training examples, while minority groups such as Black Community (24) and Nation (47) are strongly underrepresented in each fold, leaving the model poorly calibrated for them. Without an official confusion matrix we cannot identify exactly which groups are confused, so we present this as the most likely explanation rather than a verified pattern. The drop is also amplified by error propagation from the upstream Task 1 and Task 2 stages.

7. Limitations

Several limitations of the present work should be acknowledged. First, the training dataset for Tasks 2 and 3 is relatively small (590 and 485 samples, respectively), which restricts the statistical power of cross-validation estimates and limits the diversity of examples available to fine-tune BETO. This scarcity is particularly harmful for the minority categories in Task 3, such as *Black Community* (24 samples) and *Nation* (47 samples), which are strongly underrepresented across all five folds.

Second, as detailed in Section 5.5, the official Task 2 and Task 3 results are produced by an end-to-end routed pipeline, so classification errors in upstream tasks cascade into downstream ones: a comment misclassified as Non-Support in Task 1 is never passed to Task 2 or Task 3 during inference. This means the reported test scores for Tasks 2 and 3 are conditioned on the correctness of the preceding stages and conflate two sources of error—the subtask classifier itself and the propagated upstream error. A formal decomposition that isolates these two contributions was not conducted.

Third, our classical path relies exclusively on unigram TF-IDF representations. Higher-order n-grams, character-level features, psycholinguistic lexicons, or sentiment scores—used in related work on social support and hope speech detection—were not explored and could provide complementary signals, particularly for the fine-grained group classification in Task 3.

Fourth, the experiments do not include data augmentation strategies such as K-means-based oversampling [6] or back-translation, which have been shown to help in imbalanced classification settings similar to the one addressed here.

Finally, this work focuses exclusively on Spanish. This is a deliberate scope decision: SSD-2026 is part of IberLEF, whose focus is the languages of the Iberian Peninsula, and our system targets the Spanish dataset of the shared task, for which a strong monolingual model (BETO) is available. We therefore did not run English experiments; although the original social support task was introduced on an English dataset [4, 6], the Spanish setting [5] is the relevant target here. Generalization to other Iberian languages or dialectal variants was not tested and remains an open question for future work.

8. Conclusions and Future Work

This paper presented a hierarchical system for social support detection in Spanish social media comments, addressing three subtasks that correspond to progressively more fine-grained classification problems. We compared two complementary modeling paths: a classical pipeline based on unigram TF-IDF representations and a transformer-based pipeline using BETO [1], the Spanish pre-trained BERT model.

The experiments answer the three research questions posed in the Introduction as follows. With respect to RQ1, classical TF-IDF models do capture lexical cues for supportive language, and SVM-Linear consistently emerges as the strongest classical baseline in Tasks 1 and 2, while Soft Voting leads in Task 3. With respect to RQ2, BETO improves over all classical baselines in every subtask during cross-validation, with macro F1 gains of 5.52 points (Task 1), 8.05 points (Task 2), and 3.38 points (Task 3), confirming the advantage of contextualized representations for this problem. With respect to RQ3, the pipeline generalizes well on Tasks 1 and 2, where test-set macro F1 scores (0.8750 and 0.7698, respectively) are close to or above cross-validation estimates, but shows a pronounced drop on Task 3 (from 0.7812 to 0.5881), pointing to fine-grained group classification as the main open challenge.

These results are broadly consistent with prior work on social support detection [4, 6] and hope speech detection in Spanish [8], which also find that transformer-based models outperform classical approaches and that multiclass settings with small training sets are the most challenging configuration.

Future work. Several directions are worth pursuing in follow-up studies.

- **Data augmentation.** Techniques such as back-translation, synonym replacement, or K-means-based oversampling [6] could mitigate the class imbalance problem in Tasks 2 and 3 and improve generalization for underrepresented group categories.
- **Richer feature spaces.** Incorporating psycholinguistic features, sentiment lexicons, or character-level representations alongside TF-IDF—as explored by Ahani et al. [4]—may close part of the gap between classical and transformer models.
- **Larger and more recent language models.** Fine-tuning larger Spanish or multilingual models (e.g., RoBERTa-large trained on Spanish corpora, or GPT-4-based classifiers) could push performance further, especially for the fine-grained multiclass task.
- **End-to-end error propagation analysis.** Quantifying how upstream classification errors in Task 1 affect downstream performance in Tasks 2 and 3 would allow for a more precise characterization of the pipeline's weak points and could motivate joint or multi-task learning formulations.
- **Cross-lingual transfer.** Evaluating the system on related languages (e.g., Catalan, Portuguese, or other Iberian variants) would test whether the representations learned for Spanish social support generalize across related varieties.

Acknowledgments

We thank the organizers of the SSD-2026 shared task at IberLEF 2026 for providing the Spanish social support detection dataset and the evaluation platform used in this work.

9. Declaration on Generative AI

During the preparation of this work the authors used Claude (Anthropic) to verify compliance of the manuscript format with the submission guidelines provided by the organizers. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

References

[1] J. Cañete, G. Chaperon, R. Fuentes, J.-H. Ho, H. Kang, J. Pérez, Spanish pre-trained BERT model and evaluation data, in: PML4DC at ICLR 2020, 2020.

[2] M. Shahiki Tash, L. Ramos, Z. Ahani, A. Y. Barrera Animas, F. Balouchzahi, O. Kolesnikova, G. Sidorov, A. Gelbukh, R. Monroy, F. Vargas, Overview of SSD-2026 at IberLEF 2026: Social support detection, in: Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2026), co-located with the 42nd Conference of the Spanish Society for Natural Language Processing (SEPLN 2026), CEUR-WS.org, 2026.

[3] A. Bonet-Jover, J. Á. González-Barba, L. Chiruzzo, Overview of IberLEF 2026: Natural language processing challenges for Spanish and other Iberian languages, in: Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2026), co-located with the 42nd Conference of the Spanish Society for Natural Language Processing (SEPLN 2026), CEUR-WS.org, 2026.

[4] Z. Ahani, M. Shahiki Tash, F. Balouchzahi, L. Ramos, G. Sidorov, A. Gelbukh, Social support detection from social media texts, PLOS ONE 21 (2026) e0337476. URL: https://doi.org/10.1371/journal.pone.0337476. doi:10.1371/journal.pone.0337476.

[5] M. Shahiki Tash, L. Ramos, Z. Ahani, R. Monroy, H. Calvo, G. Sidorov, Online social support detection in spanish social media texts, arXiv preprint arXiv:2502.09640 (2025). URL: https://arxiv.org/abs/2502.09640. doi:10.48550/arXiv.2502.09640.

[6] O. Kolesnikova, M. Shahiki Tash, Z. Ahani, A. Agrawal, R. Monroy, G. Sidorov, Advanced machine learning techniques for social support detection on social media, Heliyon 11 (2025). doi:10.1016/j.heliyon.2025.e43437.

[7] M. Shahiki-Tash, Z. Ahani, L. Ramos, O. Kolesnikova, A. Y. Barrera-Animas, Linguistic analysis in different types of support using LIWC for Spanish and English online communities, Research Square (preprint) (2026). URL: https://doi.org/10.21203/rs.3.rs-9557287/v1. doi:10.21203/rs.3.rs-9557287/v1.

[8] D. García-Baena, M. Á. García-Cumbreras, S. M. Jiménez-Zafra, J. A. García-Díaz, R. Valencia-García, Hope speech detection in Spanish: The LGBT case, Language Resources and Evaluation 57 (2023) 1487–1514. doi:10.1007/s10579-023-09638-3.

[9] K. Spärck Jones, A statistical interpretation of term specificity and its application in retrieval, Journal of Documentation 28 (1972) 11–21. doi:10.1108/eb026526.

[10] D. W. Hosmer, S. Lemeshow, R. X. Sturdivant, Applied Logistic Regression, 3 ed., Wiley, 2013.

[11] C. Cortes, V. Vapnik, Support-vector networks, Machine Learning 20 (1995) 273–297. doi:10.1007/BF00994018.

[12] L. Breiman, Random forests, Machine Learning 45 (2001) 5–32. doi:10.1023/A:1010933404324.

[13] T. G. Dietterich, Ensemble methods in machine learning, in: Multiple Classifier Systems, volume 1857 of *Lecture Notes in Computer Science*, Springer, 2000, pp. 1–15. doi:10.1007/3-540-45014-9_1.

[14] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, É. Duchesnay, Scikit-learn: Machine learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.

[15] 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), volume 30, 2017.

[16] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of NAACL-HLT 2019, 2019, pp. 4171–4186.

[17] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, et al., Transformers: State-of-the-art natural language processing, in: Proceedings of EMNLP 2020: System Demonstrations, 2020, pp. 38–45.

[18] S. Bird, E. Klein, E. Loper, Natural Language Processing with Python, O'Reilly Media, 2009.