Fine-Tuned Transformers vs. LLM Prompting for Social Support Detection: A Comparative Study

Claudio Diego Vazquez Nava[1]

[1]Universidad Panamericana, Mexico City, Mexico

Abstract

This paper presents a comparative study of fine-tuning and LLM prompting strategies for detecting social support in a bilingual setting (English and Spanish). We evaluated a fine-tuned version of XLM-RoBERTa-base with a prompting strategy comprising zero-shot, few-shot, chain-of-thought, and task-decomposition prompts on the SSD-2026 shared-task dataset. Our experiments address 3 hierarchical subtasks for social support detection. (1) Binary support detection, (2) target identification (individual vs. group), and (3) community classification. On the official test set, fine-tuning obtained the best macro-F1 on all three subtasks (0.815, 0.777, and 0.730). However, prompting remained competitive on some low-sample classes, such as Black Community, where few-shot prompting exceeded fine-tuning; we hypothesize that prompting may benefit from the broad knowledge of LLMs in these classes, although our single-model comparison does not allow us to conclude decisively. This observation ties with the findings of Brown et al. [1], where prompting can be competitive in low-resource tasks. Task decomposition and chain-of-thought produced unparseable outputs and are analyzed only for reliability, not in the main comparison. The total LLM inference cost was $5.97 compared to 30 minutes of free GPU training for fine-tuned XLM-RoBERTa. Our results suggest that classification paradigms should be determined by data availability and deployment constraints.

Keywords

social support detection, NLP, transformer fine-tuning, LLM prompting, multilingual

1. Introduction

Texts from social networks have been a main focus of NLP analysis due to their high variability and their role in describing human social interactions. In the majority of cases, this analysis has been about sentiment. But analyzing prosocial language has become an important task in NLP [2]. This analysis offers a deeper understanding of empathy, solidarity, and expressions of support.

To contribute to the literature on social support, our work presents a series of tasks in a hierarchical way: (1) classification of a text as supportive or non-supportive; (2) identification of the target from a supportive text (group or individual); (3) classification of which community is supported (Nation, Other, LGBTQ, Black Community, Religion, Women). The data considered for these tasks is a bilingual mix of English and Spanish.

Text classification has been led by two dominant paradigms in recent years. The first, fine-tuning, modifies an existing pre-trained transformer for a specific task by training on this model. The second prompting leverages the extensive knowledge of large language models, using zero-shot or few-shot methods to obtain the answer without a training phase. Both approaches entail trade-offs in terms of cost, data requirements, and time [3].

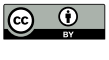
To extend existing literature [4], we compare the two current paradigms for text classification and apply them to the detection of social support. The comparison is performed by fine-tuning XLM-RoBERTa-base [5] and evaluating it with prompting strategies including zero-shot, few-shot, chain-of-thought, and decomposition, using Claude Haiku 4.5.

The main contributions of this paper are:

1. A comparison between fine-tuning vs. LLM prompting applied to social support detection.
2. An analysis of the trade-offs of each paradigm.

IberLEF 2026, September 2026, León, Spain

0243416@up.edu.mx (C. D. V. Nava)



3. Performance analysis between models in minority classes.
4. Documentation of output errors in LLM prompting.

2. Related work

2.1. Social support and prosocial language detection

Previous work on social support has described early datasets for detecting emotional support [6], in which the authors found that cancer patients with greater online support stay longer in online cancer groups. This work highlights the importance and necessity of detecting social support.

Furthermore, in subsequent years, research expanded to a broader social media context, analyzing solidarity and help-seeking within short-form text formats. [7].

Moreover, recent work on this task has extended the analysis by applying traditional and advanced machine learning techniques to detect social support [8], expanded the scope to include Spanish social media texts [9], and analyzed the linguistic characteristics of support texts in English and Spanish [10].

Sentiment and social support detection has evolved from lexicon-based methods and traditional machine learning [11], to deep learning models [12] and transformer architectures, which have increased performance.

2.2. Transformer fine-tuning for text classification

In the field of fine-tuning for text classification, pre-trained transformers such as BERT [13], and RoBERTa [14] have become the standard [15]. Fine-tuning takes the pre-trained model weights and further fine-tunes them.

2.3. LLM prompting for classification

The newer classification paradigm, prompting, surged after improvements to LLMs made classification viable. Brown et al. report that GPT-3 achieves strong performance across many NLP tasks, even with zero-shot and few-shot prompting. [1]. Wei et al. show that chain-of-thought prompting improves performance, even surpassing fine-tuned GPT-3 on the GSM8K benchmark [16]. However, recent work has shown that CoT does not improve classification tasks, as it is better suited for reasoning than for pattern matching [17]. The reliability of LLM outputs continues to be researched, given problems such as hallucinations, inconsistent results and inconsistent outputs [18].

2.4. Comparative studies

Recent studies compared both approaches for classification. The general finding is that fine-tuned models outperform prompting, given enough labeled data (above 500 examples) [3]. Prompting still excels in low-resource environments (with no training or deployment resources) and in zero-shot scenarios (with no training data available). However, most studies focus on English-only texts. Our work contributes to the literature by analyzing a bilingual setting with hierarchical subtasks, class imbalance, and limited training data.

3. Dataset

The training data consists of 7,998 English comments and 2,600 Spanish comments [2]. Table 1 shows the class distribution for each task. All tasks present unbalanced data. Task 3 presents data scarcity for some groups (Women, Religion, and Black Community), with fewer than 150 samples per group.

Important characteristics of the dataset: (1) Subtask 1 is imbalanced by approximately 77% non-supportive comments in both languages; (2) English data comprises about three-quarters of all data.

Table 1
Class distribution in the training data by language.

Subtask	Class	EN	ES
Task 1	Non-Supportive	6,203	2,010
	Supportive	1,795	590
Task 2	Individual	345	134
	Group	1,450	456
Task 3	Nation	786	47
	Other	416	82
	LGBTQ	123	224
	Black Community	91	24
	Women	19	48
	Religion	15	60

(3) The insufficiency of the data in Task 3 for *Women*, *Black Community*, and *Religion* constitutes a challenge to supervised learning.

We applied a stratified 80/20 train–validation split per language, preserving class ratios. Using the same random seed (42) allows a fair comparison.

4. Methodology

We compare two paradigms:

1. Full fine-tuning a multilingual transformer (XLM-RoBERTa) without parameter freezing.
2. Prompting a large language model (Haiku 4.5) with different strategies. For both English and Spanish datasets, prompt remains the same.

4.1. Fine-tuned XLM-RoBERTa

We adhere to the findings of Conneau et al. [5], who showed that multilingual transformers achieve strong cross-lingual transfer even with limited data. We fine-tuned XLM-RoBERTa-base, a multilingual transformer pre-trained on 100 languages. We created three separate models, one for each task.

Training details.

We performed full fine-tuning of XLM-RoBERTa-base with no frozen layers. So all of the transformer's parameters were updated.

Input tokens were tokenized with a maximum length of 128. Longer or shorter comments were cut or padded, respectively.

The hyperparameters we set:

1. AdamW optimizer
2. Learning rate 2×10^{-5}
3. Linear warmup over 10%
4. Weight decay of 0.01
5. Gradient clipping at 1.0
6. 128 max tokens

Given the severe imbalance of the dataset, we employ class-weighted cross-entropy, penalizing higher errors for minority classes.
The training was set as follows:

1. Task 1: 4 epochs with batch size 32

2. Task 2: 5 epochs with batch size 32
3. Task 3: 8 epochs with batch size 16

During training the model was evaluated with the validation split after every epoch. Only the best model checkpoint (by macro-F1 score) is reported. Therefore epochs reported above denote the maximum number of epochs.

The training was done on a Tesla T4 GPU in Google Colab.

4.2. LLM prompting with Claude Haiku 4.5

The LLM prompt was performed using different prompt strategies using Claude Haiku 4.5, the latest cost-optimized model from Anthropic. The reason is cost and time.

1. **Zero-shot:** No examples are provided. Only the task and the permitted labels.
2. **Few-shot (k = 3 and k = 5):** Includes k random labeled examples *per class*. Task 1 includes 2 * k examples, task 2 includes 2 * k examples and task 3 includes 6 * k examples. These samples are picked only once and only samples within the range of [20-200] characters are selected. Samples are chosen from the training subset.
3. **Chain-of-thought (CoT):** Expands the zero-shot prompt to include a step-by-step instruction. Output tokens are limited to 150.
4. **Task decomposition** (Task 3 only): A single prompt chains all three sub-tasks hierarchically. First, asks to determine support, then the target, and finally the group.

The resulting outputs for the chain-of-thought were mapped by keyword matching. First, look at the last word; if no label was found, use the full response. To avoid erroneous matches, we prioritized longer, more specific keys over shorter ones. This prevents "non-supportive" responses from being mapped as "supportive". All unparseable responses are assigned to the majority class.

All API calls use temperature 0.0 for deterministic outputs. The maximum output token length was limited to 150 for chain-of-thought and 200 for task-decomposition.

4.3. Prompt per task

Task 1: Support or non support

> You are a social media analyst. Determine whether the following comment expresses social support (encouragement, care, admiration, help, or solidarity toward someone). A comment that promotes violence or harm is NOT supportive. Respond with ONLY one word: 'Supportive' or 'Non-Supportive'.

Task 2: Community or Individual

> You are a social media analyst. The following comment has been identified as expressing social support. Determine whether the support is directed at a specific individual person or at a group/collective/community. Respond with ONLY one word: 'Individual' or 'Group'.

Task 3: Target Community

> You are a social media analyst. The following comment expresses social support directed at a group or community. Identify which community is being supported. Respond with ONLY one of these exact labels: 'Black Community', 'LGBTQ', 'Nation', 'Other', 'Religion', 'Women'.

4.4. Prompt per strategy

Chain of Thought A chain-of-thought suffix is added at the end of each task prompt

> Think step by step before answering. First analyze the comment's tone, language, and intent. Then give your final answer on the LAST line as ONLY the label.

Task decomposition A single prompt performs all three tasks in sequence. Without using previous prompts

> You are a social media analyst. Analyze the following comment step by step: Step 1: Is this comment supportive (encouragement, care, solidarity)? A comment promoting violence is NOT supportive. Step 2: If supportive, is the support directed at an Individual or a Group? Step 3: If directed at a group, which community? Choose from: " 'Black Community', 'LGBTQ', 'Nation', 'Other', 'Religion', 'Women'. Respond in EXACTLY this format (3 lines, nothing else):Step 1: [Supportive/Non-Supportive]Step 2: [Individual/Group/N/A]Step 3: [community label/N/A]

5. Experimental setup

To keep a fair comparison, both paradigms are evaluated on the same 20% validation split, ensuring that performance metrics are calculated on the same samples and under the same conditions. For few-shot prompts, the examples were extracted from the training split, preventing leakage into the evaluation.

Our primary evaluation metric reported is the macro-averaged F1-score. Providing equal weight to all classes was necessary due to the imbalance in the set. We report additional metrics, including accuracy, weighted F1-score, precision, and recall, each per language and per task. For all reported metrics, Task 2 and Task 3 are evaluated hierarchically, as they cannot be evaluated unless the previous task is positive. All experiments used a fixed random seed (42) to ensure reproducibility.

We list the python libraries versions used for all the experiments:

1. Python 3.12.13
2. torch 2.11.0+cu128 | CUDA 12.8
3. transformers 5.12.1
4. datasets 4.0.0
5. accelerate 1.14.0
6. scikit-learn 1.6.1
7. anthropic 0.115.1
8. numpy 2.0.2
9. pandas 2.2.2

6. Results

6.1. Main results

Table 2 presents macro-F1 scores across all methods and subtasks. Fine-tuning noticeably outperforms LLM prompting in all tasks.

Table 3 presents macro-F1 scores for task 3. Noticeably, 3 of the classes in task 3 are outperformed by LLM prompting. But only 2 of these classes are severely underrepresented (fewer than 150 samples each). The other is the highest represented class in the task.

Table 4 presents the amount of unparseable responses. Only chain of thoughts and decomposition present unparseable results (outputs that do not give a single label as result), which shows evidence of the unreliability of step-by-step reasoning for classification.

Table 2
Macro-F1 comparison across methods and subtasks on the validation split. Best result per subtask in bold.

Method	Task 1	Task 2	Task 3
Fine-tuned XLM-R	**0.809**	**0.836**	**0.881**
Zero-shot	0.679	0.779	0.694
Few-shot (k=3)	0.695	0.779	0.664
Few-shot (k=5)	0.664	0.728	0.751
Chain-of-thought	0.551	0.548	0.526
Decomposition	—	—	0.552

Table 3
Task 3 per-class F1 scores on the validation split. Bold indicates best per class. † marks classes where the LLM exceeds fine-tuning.

Class	Fine-tuned	Zero-shot	Few-3	Few-5	CoT
Black Community	0.863	0.880†	0.852	**0.885**†	0.444
LGBTQ	**0.976**	0.900	0.947	0.962	0.880
Nation	0.875	0.748	0.714	**0.876**†	0.712
Other	**0.772**	0.440	0.443	0.496	0.120
Religion	**0.933**	0.306	0.218	0.438	0.258
Women	0.867	**0.889**†	0.813	0.849	0.741

Table 4
Number of unparseable LLM responses per strategy and task.

Strategy	**Task 1** (n=2120)	**Task 2** (n=477)	**Task 3** (n=382)
Zero-shot	0	0	0
Few-shot (k=3)	0	0	0
Few-shot (k=5)	0	0	0
Chain-of-thought	1,349 (63.6%)	24 (5.0%)	91 (23.8%)
Decomposition	—	—	94 (24.6%)

6.2. Fine-tuning: detailed results

Table 5 shows the fine-tuned model's performance broken down by language. All of them show remarkable results.

Table 5
Fine-tuned XLM-RoBERTa: per-language validation results.

Task	Lang	N	Accuracy	Macro-F1
Task 1	EN	1,600	0.854	0.796
Task 1	ES	520	0.887	0.851
Task 2	EN	359	0.895	0.842
Task 2	ES	118	0.881	0.817
Task 3	EN	290	0.864	0.840
Task 3	ES	92	0.891	0.874

Although English has substantially more data, Spanish still outperforms English on Tasks 1 and 3. However, there is no basis for conclusive explanation of these results. The Spanish validation set is small, which makes these observations noisy. Furthermore, rather than an intrinsic difference between languages, the gap may reflect properties of the annotations. We leave a systematic investigation for

future work.

6.3. Task 3: per-class analysis

Table 6
Per-language comparison on the validation split: fine-tuned vs. best LLM strategy.

Task	Lang	FT Macro-F1	Best LLM	LLM Macro-F1	Gap
Task 1	EN	0.796	Few-shot 3	0.661	−0.135
Task 1	ES	0.851	Few-shot 3	0.805	−0.046
Task 2	EN	0.842	Few-shot 3	0.810	−0.032
Task 2	ES	0.817	Zero-shot	0.869	+0.052
Task 3	EN	0.840	Few-shot 5	0.727	−0.113
Task 3	ES	0.874	Few-shot 5	0.838	−0.036

As expected, fine-tuned models outperform LLM prompting in almost every task. The only exception is Spanish in task 2. Where LLM showed a slight improvement of 0.052, which may be attributable to the limited Spanish training data for Task 2 (about 472 examples, i.e., 80% of the 590 instances in Table 1).

6.4. Official test-set results

The tables above report the results of our internal validation split. We now report results on the official shared-task test set [19] [20]. The metrics report is limited to the fine-tuned model and the three prompting strategies that produced fully parseable responses (zero-shot and few-shot with k = 3, 5); both chain-of-thought and task decomposition are omitted given their unparseable outputs (Table 4) and low performance in the validation split (Table 2). Both fine-tuning and prompting strategies are evaluated hierarchically; task 2 and task 3 depend on the results of the previous task; therefore, an error in the previous task carries over to the next task.

Table 7
Macro-F1 on the official test set (Overall). Best result per subtask in bold.

Method	Task 1	Task 2	Task 3
Fine-tuned XLM-R	**0.815**	**0.777**	**0.730**
Zero-shot	0.693	0.622	0.549
Few-shot (k=3)	0.707	0.664	0.560
Few-shot (k=5)	0.703	0.692	0.608

On the test dataset, fine-tuning still achieves the best overall macro-F1 across all three subtasks (0.815, 0.777, 0.730). Few-shot prompting still remains the best prompting variant, and the performance gap to fine-tuning is smallest on Task 1 and increases as the task becomes more difficult. Table 8 breaks the results down by language.

Fine-tuning achieves the best macro-F1 score on all Spanish subtasks and on English Tasks 1–2. English Task 3 is the only task where few-shot ($k = 5$) slightly exceeds fine-tuning (0.523 vs. 0.470). Both fine-tuning and prompting strategies achieve higher scores in Spanish across all three subtasks. As previously stated on the validation split, there is no basis for a conclusive explanation of this advantage by language. We leave a systematic investigation for future work. Table 9 shows the per-class breakdown for Task 3.

On Task 3, prompting outperforms fine-tuning only on the Black Community, where few-shot ($k = 5$) is best overall (0.681 vs. 0.609). Fine-tuning remains the leading strategy across all categories, including the low-sample categories, Women and Religion. These per-class supports are small (for example, 29

Table 8
Macro-F1 on the official test set by language. Best result per subtask and language in bold.

Method	Lang	Task 1	Task 2	Task 3
Fine-tuned XLM-R	EN	**0.800**	**0.750**	0.470
Zero-shot	EN	0.681	0.531	0.387
Few-shot (k=3)	EN	0.693	0.614	0.422
Few-shot (k=5)	EN	0.689	0.638	**0.523**
Fine-tuned XLM-R	ES	**0.860**	**0.835**	**0.820**
Zero-shot	ES	0.722	0.788	0.670
Few-shot (k=3)	ES	0.747	0.745	0.653
Few-shot (k=5)	ES	0.742	0.762	0.632

Table 9
Task 3 per-class F1 on the official test set. Bold indicates best per class. † marks classes where a prompting strategy exceeds fine-tuning.

Class	Fine-tuned	Zero-shot	Few-3	Few-5
Black Community	0.609	0.619	0.651	**0.681**†
LGBTQ	**0.854**	0.818	0.813	0.844
Nation	**0.818**	0.640	0.697	0.784
Other	**0.590**	0.351	0.323	0.420
Religion	**0.774**	0.226	0.292	0.300
Women	**0.733**	0.640	0.583	0.621

test examples for Black Community and 17 for Women), and we did not perform statistical significance testing, so these differences should be read as indicative rather than conclusive.

Table 10
Official SSD-2026 leaderboard placement of our submission, by subtask and language (macro-F1). "Best" and "Median" are computed over all participating systems.

Subtask	Lang	Ours	Rank	Best	Median
Task 1	EN	0.800	19 / 40	0.832	0.798
Task 1	ES	0.860	8 / 41	0.890	0.838
Task 2	EN	0.750	26 / 39	0.870	0.759
Task 2	ES	0.835	20 / 40	0.920	0.813
Task 3	EN	0.470	20 / 38	0.875	0.465
Task 3	ES	0.820	11 / 39	0.950	0.611

(8th of 41)

Table 10 compares our best submission metrics (fine-tuned model) with the other participants in the SSD-2026 shared task. Our best metrics are within the Spanish subset, ranking in the upper quartile for Task 1 (8th of 41) and Task 3 (11th of 39). Notably in English subtasks, our score remains close to the median, even when our macro-F1 score is below 0.50. This in turn reflects a difficult subtask for all participants rather than a failure from our system.

6.5. API token usage

API token usage is reported only for the validation split, as the test dataset does not include chain-of-thought and task decomposition calls.

The total cost of the evaluation was of $5.97. It stands out that consistently the most expensive strategy was Chain-of-Thought. With a range of 20x-36x the number of output tokens compared to

Table 11
API token usage and cost for Claude Haiku 4.5 ($1/MTok input, $5/MTok output).

Task	Strategy	Input tokens	Output tokens	Cost (USD)
Task 1	Zero-shot	231,170	15,666	$0.31
Task 1	Few-shot 3	642,450	15,430	$0.72
Task 1	Few-shot 5	924,410	15,216	$1.00
Task 1	CoT	309,610	313,344	$1.88
Task 2	Zero-shot	49,110	1,908	$0.06
Task 2	Few-shot 3	123,522	1,908	$0.13
Task 2	Few-shot 5	194,118	1,908	$0.20
Task 2	CoT	66,759	69,411	$0.41
Task 3	Zero-shot	43,904	1,908	$0.05
Task 3	Few-shot 3	290,294	1,757	$0.30
Task 3	Few-shot 5	428,578	1,755	$0.44
Task 3	CoT	58,038	55,619	$0.34
Task 3	Decomp.	81,722	9,119	$0.13
TOTAL		**3,443,685**	**504,949**	**$5.97**

zero-shot.

The number of output tokens in CoT may explain why it produces so many parsing errors. With 313,344 output tokens divided by 2,120 calls, the average is 148 tokens. The maximum token output limit of 150 tokens could have truncated the final response.

7. Discussion

7.1. Why fine-tuning wins overall

On the official test set, fine-tuning achieves the best overall macro-F1 on all three subtasks (Table 7). The advantage of fine-tuning stems from having prior label data that helps create decision boundaries. Fine-tuning adapts to solve only the specific task it was trained on, whereas an LLM has a broader understanding.

7.2. Where LLMs compete: world knowledge on minority classes

On the test set, prompting exceeds fine-tuning on the Black Community class, where all three prompting strategies exceed the fine-tuned model and few-shot ($k = 5$) is best overall (0.681 vs. 0.609, Table 9). We hypothesize that this can occur given that Black Community is widely recognized social category, in which the LLM's pretraining data can provide previous context; nonetheless, due to the limitations of the experiment, a single LLM and few classes, we cannot establish this causally. Furthermore, fine-tuning leads the performance on Women and Religion, both being low-sample classes in the test set, further contradicting the hypothesis. Meanwhile, the category Other favors fine-tuning, given its ambiguous definition; fine-tuning can clarify what is considered other.

These observations suggest a tentative heuristic to be tested and researched in future work: *prompting can be competitive on low-sample classes when the class concept is broadly present in its pretraining data, while fine-tuning is the best option on task-specific classes.*

7.3. Error propagation in the hierarchical cascade

Both Task 2 and Task 3 are predicted only when the preceding task is positive; in turn, errors propagate down the hierarchy, which leads to errors in Task 2 and Task 3 without a proper prediction of the comment. To maintain consistency, both fine-tuning and prompting paradigms are affected similarly, which

explains the systematic decrease in the macro-F1 score across subsequent tasks. A non-hierarchical design, would isolate per-task ability.

7.4. Chain-of-thought

We observed that chain-of-thought responses were systematically close to the 150 maximum token output limit (Table 4). This implies that the majority of CoT responses were truncated, leaving 63.6% unparseable. Therefore, chain-of-thought comparison was restricted, and we excluded CoT from our comparison conclusions. We note that an ideal evaluation would require no limit on the maximum token output.

7.5. Cross-lingual considerations

The multilingual fine-tuned XLM-RoBERTa model performed similarly between the two languages, despite the unbalanced set. This validates the choice of a multilingual model and aligns with the findings of Conneau et al. [5], as it allows pre-trained data to transfer well between both languages.

Claude Haiku 4.5 is also a multilingual model, which enabled good overall performance. On the test set, prompting scored higher in Spanish than in English across all 3 tasks (Table 8). We again emphasize that there is no strong evidence that explains this phenomenon. And we suggest a systematic investigation as future work.

7.6. Practical implications

Both methods provided good results and trade-offs.

Fine-tuning requires a one-time training with a T4 GPU and classifies inputs faster than an LLM API call. But requires data and a dedicated deployment space.

LLM prompting requires no training data but incurs per-inference costs and may lead to output parsing issues. It also has higher per-inference time costs.

For systems that require consistent, high-volume classification, fine-tuning is the better option. Finding sufficient training data will offset the caveats of LLM prompting, such as long-term high costs and slow classification.

8. Limitations

Limitations should be considered when interpreting our results. First, the chain-of-thought strategy was compromised as the majority of results were truncated; a higher limit is needed for a proper comparison. Second, we evaluated a single LLM (Claude Haiku 4.5) and a single fine-tuned model (XLM-RoBERTa-base), so the results cannot be generalized to other models. Third, all comparisons are conducted without statistical significance testing; therefore, our results remain non-definitive. Fourth, the macro-F1 score is lower on the test set than on the validation split for both fine-tuning and prompting experiments, indicating a gap. Finally, this study is a comparative, system-description evaluation of two existing paradigms rather than a proposal of a new method.

9. Conclusion

Our work compares a fine-tuned transformer with LLM prompting strategies. On the official test set, our fine-tuned XLM-RoBERTa model achieved the best macro-F1 across all three subtasks (0.815, 0.777, and 0.730), outperforming Claude Haiku 4.5 overall. At the same time, prompting remained competitive on a low-sample class, Black Community, where few-shot prompting exceeded fine-tuning.

These findings suggest that both paradigms are well suited for different kinds of tasks. Fine-tuned models should be the choice when classification speed is important and training data is available. While

LLM prompting may be preferable in scenarios with low data and well-known topics, it should be the best option if the classification volume is low.

From a monetary perspective, LLM prompting is reasonable: the total cost for classifying 10,598 comments using five prompting strategies was only $5.97. In contrast, fine-tuning incurred no direct cost given the free access to a T4 GPU via Google Colab; however, this approach requires labeled data, which can be time-consuming to obtain.

Furthermore, the results of Table 4 indicate that LLM prompting is a reliable strategy for classification, as long as the task remains simple and does not require multi-step reasoning.

Based on these findings, we propose the following future research:

(1) Developing and evaluating hybrid approaches, such as fine-tuning with LLM labeled data or LLM prompting for verification after fine-tuning.

(2) Investigate LLM prompting strategies per language. Since this study used a single English prompt to classify both English and Spanish texts, future research should explore language-specific prompts to quantify the impact of same language input and output.

(3) Conduct a language-focused analysis of the annotations to find the cause why Spanish results consistently outperformed English, despite the use of the same model and the same English prompt. Including annotation quality, consistency of language across the dataset, and linguistic characteristics.

Acknowledgments

We thank the organizers of the Social Support Detection shared task at IberLEF 2026 [19] [20] for providing the dataset and evaluation framework.

Declaration on Generative AI

During the preparation of this work, the author used Claude Opus 4.8 in order to: write and debug the experimental code; clarify and format the wording of the classification prompts (with an original draft made by the author) used in the LLM-prompting experiments; format the tables in LaTeX with the original dataset data and author's own results. The author did not use generative AI to conclude the study. After using this tool, the author reviewed and edited all content as needed and takes full responsibility for the publication's content.

References

[1] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al., Language models are few-shot learners, in: Advances in Neural Information Processing Systems (NeurIPS), volume 33, 2020, pp. 1877–1901.

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

[3] M. J. J. Bucher, M. Martini, Fine-tuned 'small' llms (still) significantly outperform zero-shot generative ai models in text classification, arXiv preprint arXiv:2406.08660 (2024).

[4] A. Kermani, V. Perez-Rosas, V. Metsis, A systematic evaluation of LLM strategies for mental health text analysis: Fine-tuning vs. prompt engineering vs. RAG, in: Proceedings of the 10th Workshop on Computational Linguistics and Clinical Psychology (CLPsych 2025), Association for Computational Linguistics, Albuquerque, New Mexico, 2025, pp. 172–180. URL: https://aclanthology.org/2025.clpsych-1.14/.

[5] 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, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL), 2020, pp. 8440–8451.

[6] Y.-C. Wang, R. Kraut, J. M. Levine, To stay or leave? The relationship of emotional and informational support to commitment in online health support groups, in: Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work (CSCW), ACM, 2012, pp. 833–842.

[7] H. Purohit, G. Dong, V. Shalin, K. Thirunarayan, A. Sheth, Intent classification of short-text on social media, in: 2015 IEEE International Conference on Smart City/SocialCom/SustainCom (SmartCity), 2015.

[8] 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.

[9] 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).

[10] M. Shahiki Tash, Z. Ahani, L. Ramos, et al., Linguistic analysis in different types of support using LIWC for spanish and english online communities, Research Square preprint (2025). doi:10.21203/rs.3.rs-9557287/v1.

[11] P. Biyani, C. Caragea, P. Mitra, J. Yen, Identifying emotional and informational support in online health communities, in: Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, 2014, pp. 827–836.

[12] S. Buechel, A. Buffone, B. Slaff, L. Ungar, J. Sedoc, Modeling empathy and distress in reaction to news stories, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2018, pp. 4758–4765.

[13] 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 (NAACL-HLT), 2019, pp. 4171–4186.

[14] 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).

[15] L. Galke, A. Scherp, Are we really making much progress in text classification? a comparative review, arXiv preprint arXiv:2204.03954 (2025).

[16] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q. V. Le, D. Zhou, Chain-of-thought prompting elicits reasoning in large language models, in: Advances in Neural Information Processing Systems (NeurIPS), volume 35, 2022.

[17] Z. Sprague, F. Yin, J. D. Rodriguez, D. Jiang, M. Wadhwa, P. Singhal, X. Zhao, X. Ye, K. Mahowald, G. Durrett, To cot or not to cot? chain-of-thought helps mainly on math and symbolic reasoning, in: International Conference on Learning Representations (ICLR), 2025.

[18] L. Huang, W. Yu, W. Ma, W. Zhong, Z. Feng, H. Wang, Q. Chen, W. Peng, X. Feng, B. Qin, T. Liu, A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions, arXiv preprint arXiv:2311.05232 (2023).

[19] 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.

[20] M. Shahiki Tash, L. Ramos, A. Zahra, 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.