Enhancing Small Language Models for Cross-Lingual Generalized Zero-Shot Classification with Soft Prompt Tuning

Fred Philippy[1,2], Siwen Guo[1], Cedric Lothritz[3], Jacques Klein[2], Tegawendé F. Bissyandé[2]

[1] Zortify Labs, Zortify S.A., Luxembourg
[2] SnT, University of Luxembourg, Luxembourg
[3] Luxembourg Institute of Science and Technology (LIST), Luxembourg
{fred, siwen}@zortify.com cedric.lothritz@list.lu
{tegawende.bissyande, jacques.klein}@uni.lu

Abstract

In NLP, Zero-Shot Classification (ZSC) has become essential for enabling models to classify text into categories unseen during training, particularly in low-resource languages and domains where labeled data is scarce. While pretrained language models (PLMs) have shown promise in ZSC, they often rely on large training datasets or external knowledge, limiting their applicability in multilingual and low-resource scenarios. Recent approaches leveraging natural language prompts reduce the dependence on large training datasets but struggle to effectively incorporate available labeled data from related classification tasks, especially when these datasets originate from different languages or distributions. Moreover, existing prompt-based methods typically rely on manually crafted prompts in a specific language, limiting their adaptability and effectiveness in cross-lingual settings. To address these challenges, we introduce RoSPrompt, a lightweight and data-efficient approach for training soft prompts that enhance cross-lingual ZSC while ensuring robust generalization across data distribution shifts. RoSPrompt is designed for small multilingual PLMs, enabling them to leverage high-resource languages to improve performance in low-resource settings without requiring extensive fine-tuning or high computational costs. We evaluate our approach on multiple multilingual PLMs across datasets covering 106 languages, demonstrating strong cross-lingual transfer performance and robust generalization capabilities over unseen classes.

1 Introduction

Zero-Shot Classification (ZSC) is a task in NLP where a model classifies inputs into classes that it has not seen during training. This task is crucial in real-world scenarios where some classes are underrepresented with little or no labeled data. Traditionally, two approaches have dominated the landscape: entailment-based and similarity-based approaches. Entailment-based approaches (Yin et al., 2019) focus on understanding relationships between sentences, particularly determining the level of entailment between the document and the potential class labels. This method requires the model to have a deep understanding of language structure and logic. On the other hand, similarity-based approaches focus on computing the similarity between the input and labels of each class, even if the model has never encountered them during training. This method often relies on embeddings or vector representations of text, allowing the model to make inferences based on how closely the input aligns with class descriptors (Schopf et al., 2023).

However, these methods face inherent drawbacks, as they depend on Natural Language Inference or Semantic Text Similarity datasets that require considerable effort to develop and are susceptible to potential biases (Pavlick and Kwiatkowski, 2019; Kalouli et al., 2023). In light of this, and with the acknowledgment of the extensive knowledge embedded in general pre-trained language models (PLMs) and the potential to extract it, a novel paradigm has arisen: prompting. Prompting reformulates a task as a cloze-style task using a natural language prompt, retrieves the model's masked or next token prediction, and maps it to the right class via a verbalizer, while requiring little to no training data. Nevertheless, traditional prompting methods are hindered not only by manual effort and inherent biases of the individuals creating the prompts and verbalizers, but also by other factors such as the order of examples in the prompt during in-context learning (Zhao et al., 2021; Lu et al., 2022).

To address this, Shin et al. (2020) developed an automated system for generating prompts and verbalizers using a limited number of training samples. Furthermore, Hu et al. (2022) introduced a technique that eliminates the need for training data by automatically creating a verbalizer using an external knowledge base. Motivated by the goal of elim-

Proceedings of the 1st Workshop on Language Models for Underserved Communities (LM4UC 2025), pages 61–75
May 4, 2025 ©2025 Association for Computational Linguistics

inating the need for any additional data, Zhao et al. (2023) proposed a method that forms a verbalizer using only the PLM's embedding space, without requiring any training data or external knowledge base. This approach, while efficient and effective in various ZSC tasks, shares a limitation with the methods of Shin et al. (2020) and Hu et al. (2022): it relies on language-specific prompts which introduce a language bias, making the method less effective in multilingual scenarios. Moreover, despite the high efficiency and appeal of methods that operate without existing data, their inability to leverage even a minimal amount of available data from a similar classification task in a high-resource language, can be seen as a significant limitation in our data-abundant world.

To address these shortcomings, we suggest to transform the language-specific hard prompts into trainable soft prompts (Lester et al., 2021), which can then be fine-tuned. However, directly adopting the conventional soft prompt tuning (SPT) setup leads to overfitting on the seen classes (§7), therefore, does not generalize under data distribution shifts. In response to this constraint, we introduce ***Robust Soft Prompts*** (`RoSPrompt`), a novel method for cross-lingual zero-shot topic classification through few-shot SPT, which exhibits robust out-of-distribution generalization and strong cross-lingual transfer performance. `RoSPrompt` not only retains the efficiency and effectiveness of leveraging the knowledge of PLMs but also enhances it by incorporating small sets of existing data. By doing so, we aim to broaden the applicability of ZSC in a multilingual context, ensuring more accurate topic classification across diverse languages and datasets.

Specifically, our approach

(a) enables the training of soft prompts, which are better suited for ZSC tasks compared to hand-crafted, natural language hard prompts.

(b) shows strong cross-lingual transfer performance after few-shot fine-tuning in English, with soft prompts significantly improving accuracy compared to hard prompts.

(c) displays significant robustness against data distribution shifts, enabling the fine-tuning of the prompt on any available topic classification data for subsequent use in diverse topic classification tasks.

(d) exhibits computational efficiency, as fewer than 1% of parameters are fine-tuned in comparison to full-model fine-tuning.

To showcase the efficacy of our proposed approach, we conduct a comprehensive evaluation using three distinct types of multilingual language models (encoder-only, decoder-only, and encoder-decoder) and three diverse datasets, encompassing 106 languages, thereby highlighting the versatility and applicability of our method in cross-lingual scenarios.

2 Background

Soft Prompt Tuning (SPT) Our approach is based on SPT (Lester et al., 2021), extending it specifically for cross-lingual zero-shot topic classification. SPT appends tunable vectors (soft prompts) to the input of a PLM, training only the soft prompts while keeping the original model weights frozen. This method demonstrates efficacy in various downstream tasks, providing a balance between model performance and resource efficiency, and is particularly effective for cross-lingual transfer (Philippy et al., 2024).

Given an input sequence $\mathbf{x}$ and the set of C potential classes $\mathcal{C}$, we define the two main components of SPT:

- A **soft prompt** $\mathbf{p}$ that is appended to $\mathbf{x}$ in order to obtain $\mathbf{x}' = [\mathbf{x}; \mathbf{p}]$, where $[\cdot;\cdot]$ is the concatenation function.
- A **verbalizer** $v : \mathcal{T} \to \mathcal{C}$ which maps the token predicted by the model to the respective class. $\mathcal{T} = \{t_1, \ldots, t_C\}$ is a subset of the model's vocabulary $\mathcal{V}$ and the token t_c "describes" the class c.

If we denote the function performed by the model as f, with its parameters θ (which are frozen during SPT), the logits over the vocabulary $\mathcal{V}$ for the next token in the sequence are given by:

$$f_\theta(\mathbf{x}') = \{z_1, \ldots, z_{|\mathcal{V}|}\}$$

The predicted class will then be

$$\hat{y} = \arg\max_{c \in \mathcal{C}} z_{t_c}$$

Nonparametric Prompting (NPPrompt) Zhao et al. (2023) demonstrated that PLMs possess significant innate capabilities for ZSC, even without task-specific fine-tuning. Their technique, NPPrompt, involves adding a natural language prompt to the input example, prompting the model to fill in the [MASK] for BERT-based models, or predict the next token for autoregressive and Seq2Seq models, which are then used for the final classification of the sample. Nevertheless, their strategy is primarily designed for English, as the prompts employed are in English. Applying their method to additional languages would necessitate the engineering of new prompts specific to those languages. Furthermore, despite the appeal of their zero-shot framework, particularly when there is a lack of fine-tuning data, it falls short by not accommodating the use of existing labeled data when it is available. Therefore, we suggest to extend their method by transforming the natural language prompt into a trainable soft prompt (Lester et al., 2021), enabling its training through any available topic classification data in the source language for subsequent zero-shot topic classification in any target language.

3 RoSPrompt

We describe our technique as a hybrid of SPT (Lester et al., 2021) and NPPrompt (Zhao et al., 2023). SPT excels in data efficiency but is sensitive to data distribution shifts, needing unique prompts for each topic classification dataset. On the other hand, NPPrompt uses one prompt for various data distributions but fails to leverage existing data. Our strategy combines their strengths, using a single, robust soft prompt for different data distributions and enhancing data utilization (Figure 1).

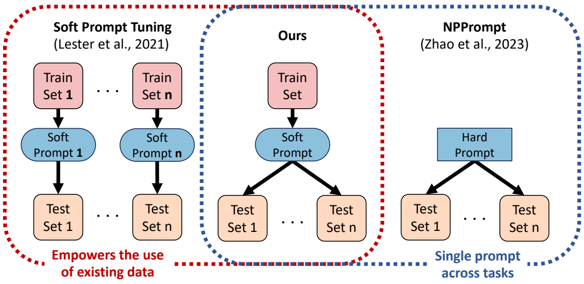


Figure 1: Conventional SPT (Lester et al., 2021), while effective in leveraging existing data, requires distinct training for each topic classification task. Conversely, NPPrompt (Zhao et al., 2023) offers versatility with a single natural language prompt for various tasks but lacks data leverage. Our method combines the strengths of both methods, enabling data utilization with a single soft prompt applicable across diverse topic classification tasks, while effectively overcoming the drawbacks of both methods.

Figure 2 provides a graphical illustration of our approach. The novelty of our method is most apparent in the training phase (§3.1), which involves three main components: **1)** the application of a multilingual verbalizer; **2)** the use of contrastive label smoothing; **3)** the adoption of a custom loss function penalty. For the inference phase of our method, we adopt the technique proposed by Zhao et al. (2023), aligning seamlessly with our goals.

3.1 Training

Below, we detail the three main components of our training approach.

1) Multilingual Class Description Tokens As mentioned before, in the standard methodology of SPT, a class c is characterized, via the verbalizer, by a single token t_c from the vocabulary $\mathcal{V}$. However, this single token might not fully capture the essence of the respective class. Moreover, it is confined to one language, leading to potential inconsistencies in multilingual settings, where the sample and the verbalizer token may be in different languages.

Therefore we propose, during training, to extend the single verbalizer token t_c to a multilingual set of verbalizer tokens $T_c = \{t_c^{(1)}, t_c^{(2)}, \ldots\}$. These augmented verbalizer tokens could be additional descriptive tokens, such as synonyms or translations of the original label token.

Our method does not mandate a uniform number of verbalizer tokens across different classes, and the manual labor involved in generating these labels is a one-time effort only required for fine-tuning the soft prompt.

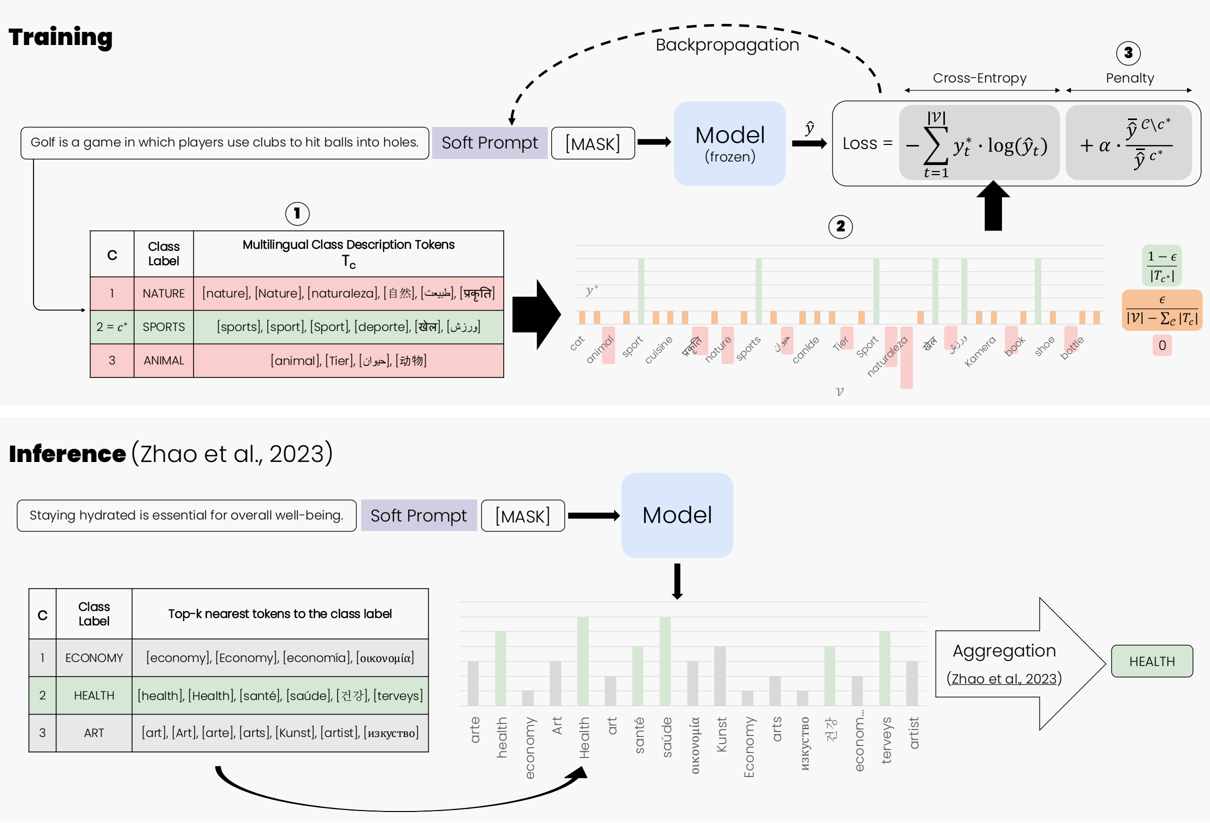
2) Contrastive Label Smoothing Conventionally, when pre-training large language models, using self-supervised tasks such as the masked language modeling or next-token prediction objective, a single token from the vocabulary is considered to be the gold truth.

Mathematically, given a token vocabulary $\mathcal{V}$, $y = \left[y_1, \ldots, y_{|\mathcal{V}|}\right]$ represents the "true" masked or next token in one-hot encoded form. When using a "hard" probability distribution, if t^* is the "true" token, $\forall t \in \mathcal{V}$,

$$y_t = 1 \times \mathbb{1}_{\{t=t^*\}}$$

for the cross-entropy loss defined as

$$CE(\hat{y}, y) = -\sum_{t=1}^{|\mathcal{V}|} y_t \times \log\left(\hat{y}_t\right)$$


Training
Backpropagation
Golf is a game in which players use clubs to hit balls into holes.
Soft Prompt
[MASK]
Model
(frozen)
$\hat{y}$
3
Cross-Entropy
Penalty
Loss = $-\sum_{t=1}^{|\mathcal{V}|} y_t^* \cdot \log(\hat{y}_t)$ $+ \alpha \cdot \frac{\bar{\hat{y}}^{\mathcal{C}\setminus c^*}}{\bar{\hat{y}}^{c^*}}$
1
2
y^*
cat
animal
sport
cuisine
प्रकृति
nature
sports
حيوان
canide
Tier
Sport
naturaleza
खेल
ورزش
Kamera
book
shoe
bottle
$\mathcal{V}$
$\frac{1-\epsilon}{|T_{c^*}|}$
$\frac{\epsilon}{|\mathcal{V}| - \sum_{\mathcal{C}} |T_c|}$
0
Inference (Zhao et al., 2023)
Staying hydrated is essential for overall well-being.
Soft Prompt
[MASK]
Model
arte
health
economy
Art
Health
art
santé
saúde
οικονομία
Kunst
Economy
arts
изкуство
건강
econom...
terveys
artist
Aggregation
(Zhao et al., 2023)
HEALTH

C	Class Label	Multilingual Class Description Tokens T_c
1	NATURE	[nature], [Nature], [naturaleza], [自然], [طبيعت], [प्रकृति]
2 = c^*	SPORTS	[sports], [sport], [Sport], [deporte], [खेल], [ورزش]
3	ANIMAL	[animal], [Tier], [حيوان], [动物]

C	Class Label	Top-k nearest tokens to the class label
1	ECONOMY	[economy], [Economy], [economía], [οικονομία]
2	HEALTH	[health], [Health], [santé], [saúde], [건강], [terveys]
3	ART	[art], [Art], [arte], [arts], [Kunst], [artist], [изкуство]

Figure 2: Visual representation of **RoSPrompt**. During training, each class is categorized by a **multilingual set of label tokens** (①). We apply **contrastive label smoothing** (②) to the probability distribution across the entire vocabulary. To further deter overfitting, we integrate a **custom penalty** (③) into the loss function. During inference, we retrieve the logits predicted by the model and use the aggregation technique proposed by Zhao et al. (2023) to make the final prediction.

where $\hat{y}$ represents the probabilities predicted by the model.

In other words, this standard method assigns a probability of 1 to the true token and 0 to the others, which might lead to overfitting as the model becomes overly confident in certain predictions. A strategy to resolve this is label smoothing (Szegedy et al., 2016), a regularization technique penalizing models for over-confident predictions and thereby mitigating overfitting. Label smoothing achieves this by shifting from a "hard" probability distribution, where only the true token gets a non-zero probability, to a "soft" distribution, where small probabilities are allocated to all or some vocabulary tokens, and the probability for the true token is reduced.

Our method employs a modified form of conventional label smoothing, which we refer to as *contrastive label smoothing*. This variation is designed to handle multiple "true" tokens for each class. Additionally, it not only prevents overconfident predictions by the model but also penalizes it for consistently favoring class label tokens over those without a class assignment. We argue that this approach leads to improved generalization over unseen classes in ZSC setups.

If $\mathcal{C}$ represents the potential classes of the training data, we denote $(T_c)_{c\in\mathcal{C}}$ as the label class token collections for each class, where T_c is the collection of verbalizer tokens of class c. If a sample belongs to class c^* we distribute the probabilities across the vocabulary, $\forall t \in \mathcal{V}$, as follows:

$$y_t = \begin{cases} \frac{1-\epsilon}{|T_{c^*}|} & \text{if } t \in T_{c^*} \\ \frac{\epsilon}{|\mathcal{V}|-\sum_{c\in\mathcal{C}}|T_c|} & \text{if } t \notin \bigcup\limits_{c\in\mathcal{C}} T_c \\ 0 & \text{otherwise} \end{cases}$$

In other words we uniformly distribute a collective probability of $1-\epsilon$ over the label tokens of the true class, i.e. T_{c^*}, and the remaining probability ϵ over all other tokens in the vocabulary **except** the label tokens of other classes.

3) Penalty In order to further penalize the soft prompt for overfitting on the seen classes during training, we additionally add a penalty to the cross-entropy loss function. We define

$$\bar{\boldsymbol{y}}^{\boldsymbol{\mathcal{C}\backslash c^*}} = \frac{\sum\limits_{c\in\mathcal{C}\backslash c^*} \sum\limits_{t\in T_c} \hat{y}_t}{\sum\limits_{c\in\mathcal{C}\backslash c^*} |T_c|} \quad \textbf{and} \quad \bar{\boldsymbol{y}}^{\boldsymbol{c^*}} = \frac{\sum\limits_{t\in T_{c^*}} \hat{y}_t}{|T_{c^*}|}$$

as the average predicted probabilities for all verbalizer tokens across all classes except the true class c^*, and for all verbalizer tokens within the class c^*, respectively.

With these definitions, we express the penalty Ω as:

$$\Omega\left(\hat{y}\right) = \frac{\bar{\hat{y}}^{\mathcal{C}\backslash c^*}}{\bar{\hat{y}}^{c^*}}$$

This penalty simply expresses the ratio of the average predicted probabilities for the true class tokens and the class tokens for all other potential classes.

Hence, the final loss function used in our approach becomes

$$L(\hat{y}, y) = CE(\hat{y}, y) + \alpha \times \Omega\left(\hat{y}\right)$$

where α is the coefficient that controls the influence of the penalty.

3.2 Inference

During inference we use the methodology proposed by Zhao et al. (2023).

The verbalizer tokens get automatically chosen by selecting the the top-k nearest tokens in the embedding space to each original English class label t_c. More specifically, for a given class c, its verbalizer tokens are given by

$$T_c = \underset{t\in\mathcal{V}}{\text{Top-k}} \left\{S(\text{emb}(t), \text{emb}(t_c))\right\}$$

where $S(\cdot)$ is the cosine similarity function.

For a given input document x, the aggregated prediction score for class c, based on the model's output logits for the next or MASK token, $\hat{y}$, is given by

$$Q(c|x) = \sum_{t\in T_c} w(t, t_c) \cdot \hat{y}_t$$

where the weight of each token in the verbalizer for a given class c is given by

$$w(t, t_c) = \frac{\exp(S(\text{emb}(t), \text{emb}(t_c)))}{\sum_{j\in T_c} \exp(S(\text{emb}(j), \text{emb}(t_c)))}$$

The final predicted class is then given by

$$\hat{c} = \arg\max_{c\in\mathcal{C}} Q(c|x)$$

This selects the class with the highest aggregated prediction probability.

4 Experimental Setup

We provide a general description of the datasets for training and evaluation, along with the models used in our experiments. Further specific details about the experimental setup can be found in Appendix A.

4.1 Datasets

For our experiments, a general English document classification dataset serves as the source data for training the soft prompts. We then evaluate these prompts on three diverse multilingual datasets, each with its own set of classes.

4.1.1 Training

As training data we use the English **DBPedia14** dataset, an ontology classification dataset, compiled from Wikipedia's most frequently used infoboxes and containing 14 distinct classes. Every class includes 40.000 samples for training and 5.000 samples for testing.

4.1.2 Evaluation

For evaluation we use 3 distinct multilingual topic classification datasets. Further details being provided in Appendix A.3.

MLSUM (Scialom et al., 2020), a multilingual news summarization dataset. We classify articles based on their summaries, using six main categories per language, although the exact categories differ slightly across languages.

MTOP (Li et al., 2021), a multilingual utterance classification dataset, featuring 11 different domains and covering 6 languages.

SIB-200 (Adelani et al., 2024), a multilingual topic classification dataset featuring 7 categories and covering more than 200 languages.

We focus on using MTOP and MLSUM to test the robustness of our method under distribution shifts, but since they are limited to high-resource languages, we leverage SIB-200 to assess cross-lingual transfer to low-resource languages, thanks to its broader language coverage

4.2 Models

We evaluate our method on three distinct models, each one based on a different architecture:

XGLM (Lin et al., 2022), a *decoder-only* model supporting 30 different languages.

mT0 (Muennighoff et al., 2023), an *encoder-decoder* model supporting 101 languages, which is a multi-task fine-tuned version of the mT5 model (Xue et al., 2021).

XLM-R (Conneau et al., 2020), an *encoder-only* model, supporting 100 languages.

More specifically, we use the `XGLM-564M`, `mT0-base` and `XLM-RoBERTa-large` variants. We describe them in more detail in Appendix A.4.

4.3 Baselines

We evaluate RoSPrompt against different baselines:

NPPrompt (Zhao et al., 2023), previously described in Section 2, using the English hard prompt "In this sentence, the topic is about [MASK]".

NPPrompt-t, a variant of NPPrompt where the English prompt is translated into the target language for inference.[1]

SPT (Lester et al., 2021), previously described in Section 2, where a soft prompt is fine-tuned on English samples using standard SPT practices and then used with NPPrompt during inference.

Zero-Shot Prompting, where we evaluate generative LLMs prompted in a zero-shot manner using a natural language instruction. Specifically, we use the 8-bit quantized variants of *Llama3.1-8B* (Dubey et al., 2024) and *Phi3.5-mini* (Abdin et al., 2024). We focus on SIB-200 for this baseline, as RoSPrompt is not designed for high-resource languages where smaller models cannot compete with large LLMs trained on extensive data. For transparency, results on MTOP and MLSUM are included in Appendix B, along with further details on this baseline.

4.4 Technical Details

Our experimental setup includes freezing all model parameters and appending a soft prompt to the initial input, as detailed in Section 2. We start by initializing the soft prompt with the embeddings of the natural language prompt from Zhao et al. (2023): "In this sentence, the topic is about". We then fine-tune this prompt using 8 randomly selected English samples from each class in DBPedia. Our methodology includes using translations of the original English label tokens into a diverse range of languages[2], and selecting words that tokenize as a single token for our multilingual label tokens. We then assess the model's performance using the trained soft prompt on all three evaluation datasets across all supported languages. During evaluation, only the original English class names are needed, with no need for further translation efforts.

To account for variability in few-shot experiments, we repeat each experiment four times using different random seeds and report the average results.

5 Results

For each of the three models, our experimental findings are presented in Table 1 for `MLSUM` and `MTOP`, across all languages. Given the extensive range of languages in SIB-200, we present average results for each major language family in Table 2, while detailed results for individual languages are shown in Appendix C (see Table 10). Overall, our methodology shows a significant advantage over NPPrompt in nearly all cases. In particular, our training method, which leverages a mere 8 samples per class from an existing topic classification dataset, generates a soft prompt that is more effective for ZSC than a natural language prompt, demonstrating robust generalization capabilities for unseen classes.

Additionally, we observe that while larger generative LLMs slightly outperform the smaller RoSprompt-enhanced LLMs on high-resource languages, they significantly underperform, often worse than the random baseline, on low-resource languages, highlighting the effectiveness of our method in such scenarios.

6 Ablation Study

To illustrate the individual contributions of each component in our training method, we carry out an ablation study. We assess the efficacy of our original method against variants lacking the loss penalty, contrastive label smoothing, and/or multilingual labels.

[1] Languages unsupported by Google Translate or with syntax that does not place the [MASK] token at the end are excluded. In Table 2, English prompt performance is used for reporting.

[2] We used the following languages as they are spoken by at least one member of our team: de, en, es, fa, fr, hi, ro, sv, uk, zh.

Model		MTOP						MLSUM			
		de	en	es	fr	hi	th	de	es	fr	ru
XGLM	**RoSPrompt**	**54.99**	**64.31**	**58.95**	**55.38**	**56.47**	47.59	**79.47**	**70.77**	**71.60**	**62.66**
	NPPrompt	48.72	55.02	47.77	47.57	52.49	**49.26**	56.30	48.83	43.92	42.97
	NPPrompt-t	26.63	55.02	42.03	19.14	-	33.27	61.22	21.68	31.97	38.24
	SPT	30.52	31.98	32.51	30.98	28.69	29.46	63.12	53.26	54.00	53.68
mT0	**RoSPrompt**	**47.65**	**53.23**	**51.48**	**48.21**	**49.42**	**46.28**	**65.24**	50.58	**48.00**	45.22
	NPPrompt	43.14	46.35	48.57	43.60	46.04	38.37	65.07	48.10	43.23	43.14
	NPPrompt-t	33.14	46.35	33.36	7.02	-	39.89	59.51	43.36	31.33	26.80
	SPT	46.22	52.31	47.87	44.19	44.42	42.98	64.64	**52.81**	47.32	**45.92**
XLM-R	**RoSPrompt**	**55.64**	**63.93**	**54.79**	**52.91**	**62.25**	**53.28**	**81.77**	**65.46**	**60.66**	53.39
	NPPrompt	36.38	46.03	35.76	34.95	47.69	39.02	62.38	50.77	52.79	**58.17**
	NPPrompt-t	35.25	46.03	35.29	28.47	-	47.05	72.95	41.89	38.18	48.37
	SPT	39.10	43.75	35.29	36.35	40.04	37.55	69.00	57.83	50.40	49.59

Table 1: Comparison of accuracy scores on the **MTOP** and **MLSUM** datasets between RoSPrompt and baselines.

Model		Afro-Asiatic	Atlantic-Congo	Austronesian	Dravidian	Indo-European	Sino-Tibetan	Turkic	Uralic
XGLM	**RoSPrompt**	**69.12**	**65.32**	**73.04**	**64.95**	**70.80**	**72.92**	**72.55**	**71.51**
	NPPrompt	60.78	61.76	59.31	58.09	61.48	58.83	62.25	62.26
	NPPrompt-t	53.92	58.82	63.73	58.09	54.41	53.68	62.25	40.69
	SPT	59.19	55.51	66.05	58.15	60.94	54.05	60.42	66.54
mT0	**RoSPrompt**	**71.69**	**71.69**	**75.61**	**75.61**	**75.75**	**74.27**	**74.39**	**73.10**
	NPPrompt	57.11	59.13	59.95	61.64	61.52	62.42	61.03	63.40
	NPPrompt-t	46.41	51.16	51.84	61.64	54.84	59.47	61.03	53.27
	SPT	65.05	66.42	67.37	70.07	69.91	72.18	68.28	69.40
XLM-R	**RoSPrompt**	**72.67**	**65.69**	**71.69**	**66.91**	**68.65**	**68.63**	**67.89**	**70.59**
	NPPrompt	57.43	56.62	63.14	64.83	64.20	63.73	65.13	65.69
	NPPrompt-t	45.26	38.24	57.25	64.83	52.05	57.84	65.13	57.03
	SPT	56.78	52.33	61.96	64.49	61.65	65.28	61.40	57.31
Llama3.1-8B		25.42	18.44	26.42	8.58	39.84	35.29	26.82	44.61
Phi-3.5-mini		42.30	38.11	57.95	7.72	55.17	54.09	46.08	65.03

Table 2: Comparison of accuracy scores on the **SIB-200** dataset between RoSPrompt and baselines.

The outcomes of this study, presented in Table 3 for MTOP across three models, indicate that all three elements are integral to our method's success. Notably, the removal of the loss penalty leads to the most significant decline in performance for XGLM and mT0, while the lack of multilingual labels has the greatest negative impact on XLM-R.

	XGLM	mT0	XLM-R
RoSPrompt	**56.28**	49.38	**57.13**
w/o penalty	30.22	31.91	51.59
w/o LS	50.05	49.71	48.18
w/o penalty & LS	29.38	41.53	51.26
w/o ML labels	50.05	**50.37**	47.40

Table 3: Ablation study results for MTOP.

This could potentially be attributed to XLM-R's enhanced code-switching capabilities (Winata et al., 2021; Zhang et al., 2023), making it more efficient at using multilingual label tokens during training compared to XGLM and mT0.

7 Generalized Zero-Shot Learning

In our initial experiments, training (*seen*) and evaluation (*unseen*) classes were distinct with merely minimal overlap. In contrast, the *Generalized Zero-Shot Learning* (GZSL) settings, which mirror real-world situations more closely, involve evaluating on a mix of both seen and unseen classes. Models in this setting often struggle with overfitting to seen classes and fail to perform well on unseen classes

(Xian et al., 2019).

Therefore, we aim to investigate whether our method is also efficient under GZSL settings. For this, we fine-tune the soft prompt on a subset of classes from a dataset, then test it on the entire set of classes. Considering the potential variability resulting from the specific choice of seen and unseen classes, we repeat this process four times for each dataset and model, each time with a different subset of seen classes. We then average the F1 scores for seen and unseen classes and present them in Table 4. These experiments are conducted with all three models, but only for the SIB-200[3] and MTOP datasets, as MLSUM does not support English, and has varying categories across languages.

		SIB-200		**MTOP**	
		Unseen	*Seen*	*Unseen*	*Seen*
XGLM	SPT	20.02	48.56	28.04	49.04
	RoSPrompt	48.68	49.60	62.41	61.50
mT0	SPT	31.32	39.44	32.26	23.49
	RoSPrompt	67.11	65.44	39.33	52.98
XLM-R	SPT	26.88	53.78	17.63	43.88
	RoSPrompt	56.64	55.68	62.54	57.96

Table 4: Comparison of average F1 scores for seen and unseen classes using standard SPT versus RoSPrompt.

For conventional SPT, there is a notable imbalance in performance between seen and unseen classes, with seen classes showing higher performance, suggesting overfitting to seen classes and poor generalization to unseen classes. However, when training the soft prompts using our method, the performance is more balanced, indicating improved generalization to unseen classes.

8 Contextualizing Our Approach

In this study, we acknowledge that comparing our approach with NPPrompt may not constitute an entirely fair comparison. RoSPrompt uses a small dataset for training, while NPPrompt directly leverages a PLM without additional fine-tuning. However, it is important to emphasize that the intent of our research is not to demonstrate RoSPrompt's performance superiority over NPPrompt. Instead, our objective is to illustrate how RoSPrompt's methodology can effectively improve cross-lingual transfer capabilities of natural language prompts. This aspect is vital as our findings indicate that merely converting hard prompts to soft prompts and then fine-tuning them using the standard SPT approach results in non-robust prompts which are ineffective for Generalized ZSC.

Additionally, while our paper focuses on topic classification, we believe that our approach could be equally effective for other types of classification tasks as well. Nonetheless, we emphasize the significance of zero-shot learning in topic classification, where classes often change more frequently over time or across domains, unlike in more stable tasks like sentiment analysis, where classes show less variation.

Furthermore, we want to emphasize the threefold efficiency of our approach: **1)** it is data efficient, requiring only a small number of labeled training samples from any comparable classification task; **2)** it is computationally efficient as fewer than 0.1% of parameters are fine-tuned compared to full-model fine-tuning, reducing training time by approximately 50% in our experiments; **3)** it is memory-efficient, as for n training processes, besides the resulting n prompts that take up a few hundred KBs at most, only one model copy is stored, in contrast to full-model fine-tuning where each model occupies several GBs of storage.

Moreover, while our method is theoretically applicable to larger models with billions of parameters, our primary target is smaller LLMs, which are often sufficient for tasks like zero-shot classification but need more focused guidance. These smaller multilingual models also excel in low-resource languages, where larger English-centric models, as we demonstrate, are less effective.

9 Conclusion

In this paper, we introduced `RoSPrompt`, a novel approach for cross-lingual zero-shot topic classification. It combines the advantages of few-shot SPT with the extensive knowledge acquired by language models in their pre-training phase. Our training method is designed for computational efficiency and incorporates three key components to enhance the standard SPT methodology, contributing to RoSPrompt's cross-lingual abilities and resilience to data distribution shifts.

[3] For computational efficiency during this experiment, we limited our evaluation to a subset of ten linguistically diverse languages (en, ru, zh, de, ar, bn, ta, ko, my, sw) instead of all supported ones.

Limitations

Our research was conducted on datasets encompassing a variety of classes and data distributions. However, the absence of multilingual datasets across entirely distinct domains limits our ability to test the method's effectiveness in distant or niche domains. Therefore, while our results are promising within the domains we studied, they may not fully represent the model's capabilities across all specific domains.

In addressing the few-shot learning nature of our approach, varied the training samples across 4 iterations for each experiment to reduce potential biases. Nonetheless, the specific selection of these samples can still influence the outcomes due to the inherent characteristics of few-shot learning. This limitation suggests that our findings could be partially influenced by the particular datasets used, and might not entirely reflect the model's performance with different or broader data samples.

Ethics Statement

In our work, we prioritized two key ethical aspects, through which we strive to contribute to the inclusive and responsible advancement of NLP technology.

Language Diversity and Equity. Our method aims to balance performance across various languages, addressing the common disparity in model effectiveness between high- and low-resource languages. By enhancing multilingual capabilities, `RoSPrompt` contributes towards more balanced performance across languages, ensuring fair and inclusive technology across diverse linguistic groups.

Environmental Responsibility. Our method is designed for computational efficiency, requiring fine-tuning of fewer than 1% of parameters compared to traditional methods. This approach not only conserves computational resources but also aligns with environmental sustainability goals by reducing the energy consumption and carbon footprint associated with training and deploying NLP models.

Acknowledgment

The author Cedric Lothritz is supported by the Luxembourg National Research Fund (FNR) PEARL program, grant agreement 16544475.

References

Marah Abdin, Sam Ade Jacobs, Ammar Ahmad Awan, Jyoti Aneja, Ahmed Awadallah, Hany Awadalla, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Harkirat Behl, et al. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*.

David Adelani, Hannah Liu, Xiaoyu Shen, Nikita Vassilyev, Jesujoba Alabi, Yanke Mao, Haonan Gao, and En-Shiun Lee. 2024. SIB-200: A Simple, Inclusive, and Big Evaluation Dataset for Topic Classification in 200+ Languages and Dialects. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 226–245, St. Julian's, Malta. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised Cross-lingual Representation Learning at Scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Shengding Hu, Ning Ding, Huadong Wang, Zhiyuan Liu, Jingang Wang, Juanzi Li, Wei Wu, and Maosong Sun. 2022. Knowledgeable Prompt-tuning: Incorporating Knowledge into Prompt Verbalizer for Text Classification. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2225–2240, Dublin, Ireland. Association for Computational Linguistics.

Aikaterini-Lida Kalouli, Hai Hu, Alexander F. Webb, Lawrence S. Moss, and Valeria De Paiva. 2023. Curing the SICK and Other NLI Maladies. *Computational Linguistics*, 49(1):199–243.

Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer, and Christian Bizer. 2015. DBpedia – A large-scale, multilingual knowledge base extracted from Wikipedia. *Semantic Web*, 6(2):167–195.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The Power of Scale for Parameter-Efficient Prompt Tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Haoran Li, Abhinav Arora, Shuohui Chen, Anchit Gupta, Sonal Gupta, and Yashar Mehdad. 2021. MTOP: A Comprehensive Multilingual Task-Oriented Semantic Parsing Benchmark. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2950–2962, Online. Association for Computational Linguistics.

Xi Victoria Lin, Todor Mihaylov, Mikel Artetxe, Tianlu Wang, Shuohui Chen, Daniel Simig, Myle Ott, Naman Goyal, Shruti Bhosale, Jingfei Du, Ramakanth Pasunuru, Sam Shleifer, Punit Singh Koura, Vishrav Chaudhary, Brian O'Horo, Jeff Wang, Luke Zettlemoyer, Zornitsa Kozareva, Mona Diab, Veselin Stoyanov, and Xian Li. 2022. Few-shot Learning with Multilingual Generative Language Models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9019–9052, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. ArXiv:1907.11692 [cs].

Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. 2022. Fantastically Ordered Prompts and Where to Find Them: Overcoming Few-Shot Prompt Order Sensitivity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8086–8098, Dublin, Ireland. Association for Computational Linguistics.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng Xin Yong, Hailey Schoelkopf, Xiangru Tang, Dragomir Radev, Alham Fikri Aji, Khalid Almubarak, Samuel Albanie, Zaid Alyafeai, Albert Webson, Edward Raff, and Colin Raffel. 2023. Crosslingual Generalization through Multitask Finetuning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15991–16111, Toronto, Canada. Association for Computational Linguistics.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No Language Left Behind: Scaling Human-Centered Machine Translation. ArXiv:2207.04672 [cs].

Ellie Pavlick and Tom Kwiatkowski. 2019. Inherent Disagreements in Human Textual Inferences. *Transactions of the Association for Computational Linguistics*, 7:677–694.

Fred Philippy, Siwen Guo, Shohreh Haddadan, Cedric Lothritz, Jacques Klein, and Tegawendé F. Bissyandé. 2024. Soft Prompt Tuning for Cross-Lingual Transfer: When Less is More. In *Proceedings of the 1st Workshop on Modular and Open Multilingual NLP (MOOMIN 2024)*, pages 7–15, St Julians, Malta. Association for Computational Linguistics.

Tim Schopf, Daniel Braun, and Florian Matthes. 2023. Evaluating Unsupervised Text Classification: Zero-shot and Similarity-based Approaches. In *Proceedings of the 2022 6th International Conference on Natural Language Processing and Information Retrieval*, NLPIR '22, pages 6–15, New York, NY, USA. Association for Computing Machinery.

Thomas Scialom, Paul-Alexis Dray, Sylvain Lamprier, Benjamin Piwowarski, and Jacopo Staiano. 2020. MLSUM: The Multilingual Summarization Corpus. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8051–8067, Online. Association for Computational Linguistics.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. 2020. AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4222–4235, Online. Association for Computational Linguistics.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the Inception Architecture for Computer Vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2818–2826, Las Vegas, NV, USA. IEEE.

Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. 2020. CCNet: Extracting High Quality Monolingual Datasets from Web Crawl Data. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4003–4012, Marseille, France. European Language Resources Association.

Genta Indra Winata, Samuel Cahyawijaya, Zihan Liu, Zhaojiang Lin, Andrea Madotto, and Pascale Fung. 2021. Are Multilingual Models Effective in Code-Switching? In *Proceedings of the Fifth Workshop on Computational Approaches to Linguistic Code-Switching*, pages 142–153, Online. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-Art Natural Language Processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Yongqin Xian, Christoph H. Lampert, Bernt Schiele, and Zeynep Akata. 2019. Zero-Shot Learning—A Comprehensive Evaluation of the Good, the Bad and the Ugly. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(9):2251–2265.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A Massively Multilingual Pre-trained Text-to-Text Transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

Wenpeng Yin, Jamaal Hay, and Dan Roth. 2019. Benchmarking Zero-shot Text Classification: Datasets, Evaluation and Entailment Approach. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3914–3923, Hong Kong, China. Association for Computational Linguistics.

Ruochen Zhang, Samuel Cahyawijaya, Jan Christian Blaise Cruz, Genta Winata, and Alham Fikri Aji. 2023. Multilingual Large Language Models Are Not (Yet) Code-Switchers. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12567–12582, Singapore. Association for Computational Linguistics.

Xuandong Zhao, Siqi Ouyang, Zhiguo Yu, Ming Wu, and Lei Li. 2023. Pre-trained Language Models Can be Fully Zero-Shot Learners. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15590–15606, Toronto, Canada. Association for Computational Linguistics.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate Before Use: Improving Few-shot Performance of Language Models. In *Proceedings of the 38th International Conference on Machine Learning*, pages 12697–12706. PMLR.

A Technical Details

Access to the code used in our research will provided in the camer-ready version.

A.1 Training

We conducted all of our experiments using the Transformers library (Wolf et al., 2020) and ran them on 4 A100 Nvidia GPUs within a few hours. We used AdamW (Loshchilov and Hutter, 2019) as an optimizer. We provide the hyperparameters used during our experiments in Table 5. Due to computational constraints, we did not perform exhaustive hyper-parameter optimization, but instead selected hyper-parameters that demonstrated satisfactory performance in preliminary experiments.

	XGLM	XLM-R	mT0
Batch size	8	8	8
Learning rate	0.01	0.01	0.3
Epochs	10	10	10
α	100	10	200
ϵ	0.2	0.1	0.8
Prompt length	8	8	9

Table 5: Hyperparameters

A.2 Evaluation

During evaluation, NPPrompt (Zhao et al., 2023) requires a parameter k, which is referred to as the *neighborhood number*. In our experimental setup, for each model and dataset type, we selected the value of k that achieved the highest average performance across the development sets of all supported languages. The specific values selected for k in the evaluation of RoSPrompt, NPPrompt (including NPPrompt-t) and SPT are presented in Tables 6, 7 and 8 respectively.

	XGLM	XLM-R	mT0
SIB-200	3	4	14
MTOP	4	2	8
MLSUM	300	5	7

Table 6: Chosen *neighborhood number* k values for **RoSPrompt**.

	XGLM	XLM-R	mT0
SIB-200	4	3	6
MTOP	3	2	5
MLSUM	5	4	6

Table 7: Chosen *neighborhood number* k values for **NPPrompt** Zhao et al. (2023) and **NPPrompt-t** (Zhao et al. (2023) with translated hard prompt).

	XGLM	XLM-R	mT0
SIB-200	2	17	5
MTOP	100	7	12
MLSUM	200	16	7

Table 8: Chosen *neighborhood number* k values for **SPT**.

Impact of Hyperparameters RoSPrompt's training methodology primarily relies on two numerical hyperparameters: the contrastive label smoothing factor, denoted as ϵ, and the penalty strength, represented by α.

In Figure 3, we illustrate RoSPrompt's performance using XGLM and mT0 on the SIB-200 dataset, using a diverse subset of languages[4], across various values for α and ϵ, while maintaining the other hyperparameter at zero each time. Generally, we find that both excessively low and high values for α and ϵ do not lead to optimal outcomes.

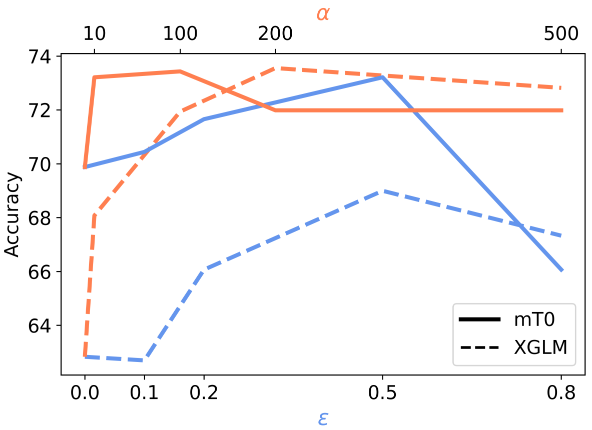


Figure 3: Average performance (accuracy) of RoSPrompt across 10 languages on SIB-200 for different values of ϵ and α.

A.3 Datasets

As source data to train the soft prompts, we use the **DBPedia14** ontology classification dataset[5] (Lehmann et al., 2015). It is a subset of the English version of DBpedia 2014[6], consisting of randomly chosen 560 000 training and 70 000 test samples equally distributed across 14 distinct classes. These classes represent the most common infobox categories on Wikipedia, including categories like Company, Artist, Athlete, Village, Animal, among others.

For evaluation we use three different multlingual datasets:

MLSUM (Scialom et al., 2020), a multilingual news summarization dataset. However, each article-summary pair is also labeled with its respective news category. Therefore, in our experiments, we use, for each article, the summary for its classification. Given the differing data sources for different languages, the categories across languages slightly differ. More specifically we use articles on society, politics, culture, sports, economy and science for Spanish, Russian and French and articles on politics, sports, economy, travel, car and education for German. This selection amounts to 8935, 612, 5950, 5315 test samples for German, Russian, French and Spanish respectively. MLSUM is licensed under the MIT License[7].

MTOP[8] (Li et al., 2021), a multilingual utterance classification dataset, featuring 11 different domains, such as alarm, reminder, recipes or weather. The dataset covers 6 languages: English, German, Spanish, French, Hindi and Thai, with respective test sample counts of 4386, 3549, 2998, 3193, 2789, and 2765. MTOP is licensed under the Creative Commons Attribution-ShareAlike 4.0 International License[9].

SIB-200[10] (Adelani et al., 2024), a multilingual topic classification dataset covering 203 languages. The dataset is derived from the FLORES-200 benchmark (NLLB Team et al., 2022) and consists of 701 training, 99 validation and 204 test samples in each language. It features 7 distinct classes: geography, politics, science/technology, travel, sports, health and entertainment. SIB-200 is licensed under the Apache License 2.0[11].

A.4 Models

In our work, we use the following models:

XGLM$_{564M}$[12] (Lin et al., 2022) is a decoder-only multilingual model supporting a diverse selection of 30 languages. Pre-trained on the CC100-XL dataset,

[4] en, ru, zh, de, ar, bn, ta, ko, my, sw

[5] https://huggingface.co/datasets/dbpedia_14

[6] https://downloads.dbpedia.org/wiki-archive/data-set-2014.html

[7] https://opensource.org/license/mit/

[8] https://huggingface.co/datasets/mteb/mtop_domain

[9] https://creativecommons.org/licenses/by-sa/4.0/

[10] https://github.com/dadelani/sib-200

[11] https://www.apache.org/licenses/LICENSE-2.0.txt

[12] https://huggingface.co/facebook/xglm-564M

	MTOP						MLSUM			
Model	**de**	**en**	**es**	**fr**	**hi**	**th**	**de**	**es**	**fr**	**ru**
Llama3.1-8B	83.26	93.50	85.32	83.15	84.69	75.26	78.13	70.21	68.27	53.92
Phi-3.5-mini	79.57	86.34	78.62	78.52	70.49	66.22	77.08	71.17	66.91	59.64

Table 9: Accuracy scores on the **MTOP** and **MLSUM** obtained through zero-shot prompting.

an expansion of CC100 (Conneau et al., 2020; Wenzek et al., 2020), it features 564 million parameters, 24 layers, a hidden dimension size of 1024, and 16 attention heads.

XLM-R$_{Large}$ [13] (Conneau et al., 2020) is an encoder-only multilingual RoBERTa-based (Liu et al., 2019) model supporting 100 languages, pre-trained on CC100 (Conneau et al., 2020; Wenzek et al., 2020) using the MLM objective. It consists of 550 million parameters, 24 hidden layers, a dimension of 1024, and 16 attention heads.

mT0$_{Base}$[14] (Muennighoff et al., 2023) is an encoder-decoder model supporting 101 languages. It is an mT5 model (Xue et al., 2021) that has been multi-task fine-tuned on the xP3 dataset[15] (Muennighoff et al., 2023). It features 584 million parameters, 12 encoder and decoder layers, 12 attention heads, and a hidden dimension size of 768.

B Additional Details on "Zero-Shot Prompting" Baseline

For this baseline, we used the 8-bit quantized versions of *Llama3.1-8B*[16] (Dubey et al., 2024) and *Phi-3.5-mini*[17] (Abdin et al., 2024), which have been designed with robust multilingual capabilities. *Llama3.1-8B* is a transformer-based language model with 8.03 billion parameters, designed for efficient text generation tasks. *Phi-3.5-mini*, a smaller variant, has 3.82 billion parameters and shares a similar transformer architecture optimized for lightweight inference. Both models were prompted using the prompt shown in Figure 4 and used with 8-bit quantization.

The results on MTOP and MLSUM are provided in Table 9.

```
I will provide text and potential
categories, and I would like you
to classify the text into one
of the given categories based on
its content.  Please ensure the
classification is accurate and
consistent.

Categories:
- Label 1
- Label 2
- ...

Text: "{Document}"

Only return the category name.
```

Figure 4: The prompt used for the **Zero-Shot LLMs** baseline with *Llama3.1-8B* and *Phi-3.5-mini*.

C Full Results for SIB-200

Table 10 presents the experimental results for each language on SIB-200, with average values per language family reported in Table 2 in Section 5.

[13] https://huggingface.co/xlm-roberta-large

[14] https://huggingface.co/bigscience/mt0-base

[15] https://huggingface.co/datasets/bigscience/xP3

[16] https://huggingface.co/meta-llama/Llama-3.1-8B-Instruct

[17] https://huggingface.co/microsoft/Phi-3.5-mini-instruct

	XGLM				**mT0**				**XLM-R**					
	RoSPrompt	***NPPrompt***	***NPPrompt-t***	***SPT***	***RoSPrompt***	***NPPrompt***	***NPPrompt-t***	***SPT***	***RoSPrompt***	***NPPrompt***	***NPPrompt-t***	***SPT***	***Llama3.1-8B***	***Phi-3.5-mini***
afr_Latn					74.02	63.24	62.75	**74.39**	**69.85**	62.75	38.73	59.07	44.12	**74.51**
als_Latn					**75.37**	62.75	56.86	70.47	**69.24**	65.69	24.02	61.15	30.88	**58.82**
amh_Ethi					**64.71**	55.39	-	63.11	**65.56**	59.80	-	62.38	2.45	**3.92**
arb_Arab	**69.12**	60.78	53.92	59.19	**71.69**	62.25	64.71	71.08	**72.67**	69.61	55.39	71.69	48.53	**72.06**
asm_Beng									**72.30**	62.75	-	59.31	**15.20**	11.76
azb_Arab					**65.07**	54.41	-	58.58	**59.80**	57.35	-	49.02	23.04	**37.25**
azj_Latn					**75.37**	63.24	-	69.73	67.52	**69.61**	-	67.16	33.33	**52.94**
bel_Cyrl					**74.02**	62.75	60.29	69.36	**67.89**	64.71	44.61	66.67	44.61	**51.47**
ben_Beng	**69.36**	61.27	61.27	59.44	**72.30**	57.35	-	69.85	**67.28**	64.22	-	59.07	**31.86**	25.00
bos_Latn									**68.14**	64.71	60.29	58.70	48.53	**60.29**
bul_Cyrl	**69.49**	63.73	57.35	63.73	**77.45**	63.24	55.88	74.02	**68.26**	66.18	59.80	67.77	51.96	**67.65**
cat_Latn	**71.81**	65.69	59.80	59.19	**79.90**	65.20	51.47	71.32	**69.61**	64.71	59.31	67.77	53.43	**75.98**
ceb_Latn					**69.12**	61.27	54.41	**69.12**					29.90	**65.20**
ces_Latn					**75.25**	62.25	50.98	71.81	**65.93**	64.71	53.92	64.83	55.88	**72.06**
cym_Latn					**63.97**	55.88	32.84	63.48	**65.69**	59.80	46.57	59.19	28.92	**49.51**
dan_Latn					**73.77**	64.22	61.76	72.06	**69.00**	67.16	38.24	64.22	52.94	**74.51**
deu_Latn	**70.22**	62.25	48.04	66.91	**75.12**	66.18	62.25	72.55	**69.36**	68.14	50.49	64.34	66.67	**80.39**
ell_Grek	**69.36**	57.84	52.45	62.75	**73.65**	58.82	65.20	69.49	69.12	**70.10**	47.06	64.83	**51.47**	43.63
eng_Latn	**73.53**	63.73	63.73	69.00	**79.41**	65.20	65.20	73.53	**68.01**	61.76	61.76	52.33	75.98	**83.33**
epo_Latn					**77.08**	67.16	40.20	73.04	**67.65**	67.16	20.59	59.44	43.14	**62.25**
est_Latn	**69.36**	61.76	22.55	65.56	**73.28**	66.18	45.59	72.30	**69.73**	66.18	47.06	58.21	36.76	**55.88**
eus_Latn	**71.20**	63.73	-	60.17	**74.51**	63.73	-	74.02	**65.81**	58.82	-	58.33	31.37	**52.45**
fin_Latn	**73.65**	62.75	58.82	67.52	**72.92**	62.25	52.45	67.28	**71.45**	66.18	59.31	60.17	47.06	**69.12**
fra_Latn	**71.08**	58.33	36.76	60.05	**77.70**	64.22	58.33	70.71	**66.18**	65.20	31.86	60.17	65.20	**78.92**
gaz_Latn									**44.24**	35.29	29.41	38.48	9.31	**25.98**
gla_Latn					**60.42**	48.53	38.24	56.13	**59.19**	54.90	41.67	53.19	13.24	**30.88**
gle_Latn					69.00	60.78	29.90	**69.24**	**63.60**	57.84	44.61	56.86	19.61	**42.16**
glg_Latn					**76.47**	67.65	47.55	**76.47**	**68.75**	64.71	52.45	68.50	52.94	**77.94**
guj_Gujr					**74.02**	65.20	-	69.24	**68.26**	60.29	-	64.09	**6.86**	2.94
hat_Latn	**65.81**	59.80	62.25	48.41	**69.73**	55.39	40.69	67.40					18.63	**54.41**
hau_Latn					60.91	51.47	42.65	**61.64**	**60.54**	59.31	45.59	51.10	23.53	**33.33**
heb_Hebr					**73.41**	59.80	51.47	68.26	**67.28**	65.20	41.67	64.83	52.45	**58.33**
hin_Deva	**67.89**	62.25	-	58.09	**72.43**	62.25	-	71.45	**71.20**	65.69	-	65.56	50.00	**55.39**
hrv_Latn									**68.26**	66.67	62.75	58.09	48.04	**64.22**
hun_Latn					**72.92**	61.76	-	68.63	**69.98**	64.71	-	53.55	50.00	**70.10**
hye_Armn					**71.69**	61.27	65.69	66.54	**70.10**	66.67	68.14	63.60	11.27	**25.00**
ibo_Latn					**71.45**	62.25	55.88	68.75					24.51	**34.80**
ind_Latn	**73.04**	59.31	63.73	66.05	**75.61**	67.16	57.84	72.92	**71.69**	67.16	63.73	67.28	52.94	**79.41**
isl_Latn					**70.96**	61.27	53.92	69.61	**67.77**	67.65	39.22	58.82	24.51	**48.53**
ita_Latn	**72.43**	63.24	48.53	63.24	**75.00**	63.24	53.43	73.16	**66.79**	65.69	44.61	63.73	64.22	**79.41**
jav_Latn									66.54	60.78	51.96	**66.67**	19.12	**61.76**
jpn_Jpan	**72.30**	62.25	-	59.68	**75.49**	62.75	-	72.18	**69.12**	64.22	-	64.22	49.51	**72.55**
kan_Knda					**72.92**	62.25	-	68.75	65.20	**66.18**	-	66.05	**8.82**	2.45
kat_Geor					**74.14**	62.25	58.33	72.30	**70.47**	66.18	55.88	65.93	5.39	**18.63**
kaz_Cyrl					**76.96**	63.24	-	71.20	**73.04**	70.10	-	69.24	28.92	**56.86**
khk_Cyrl					69.73	58.33	-	**69.85**	**65.69**	60.29	-	53.92	20.10	**34.80**
khm_Khmr					**71.57**	65.69	61.27	70.71	66.54	**66.67**	52.45	64.71	3.43	**4.90**
kir_Cyrl					**70.83**	60.78	-	69.00	**69.49**	66.67	-	61.76	22.55	**50.00**
kmr_Latn					57.35	52.94	-	**57.84**	**64.09**	59.31	-	61.40	20.10	**43.14**
kor_Hang	**68.38**	60.78	-	62.99	**71.57**	59.31	-	68.87	**69.00**	63.24	-	62.62	45.59	**73.53**
lao_Laoo					**74.39**	65.69	61.76	74.02	**68.63**	61.27	60.78	65.44	**3.92**	2.94
lit_Latn					**74.02**	64.71	64.71	69.98	66.05	**67.65**	25.98	57.35	36.76	**59.31**
ltz_Latn					66.42	55.88	44.61	**66.54**					23.53	**65.69**
lvs_Latn					**74.26**	61.76	47.55	70.22	**69.12**	62.75	24.51	48.53	38.24	**56.37**
mal_Mlym					**72.06**	58.33	-	68.26	**70.10**	66.18	-	64.83	8.33	**10.78**
mar_Deva					**71.45**	61.27	-	67.52	**67.03**	61.27	-	57.60	**35.29**	33.33
mkd_Cyrl					**76.47**	60.78	63.73	72.06	**70.83**	61.27	59.80	62.38	40.20	**63.24**
mlt_Latn					**68.26**	58.82	27.45	66.18					28.92	**65.69**
mri_Latn					56.13	47.06	26.47	**58.09**					11.76	**32.35**
mya_Mymr	**72.30**	62.75	-	61.40	**71.32**	58.33	-	70.71	**68.38**	61.76	-	**68.38**	2.45	**3.92**
nld_Latn					**76.35**	63.73	66.18	74.63	**70.34**	68.14	51.96	60.29	58.82	**79.41**
nno_Latn					**74.14**	63.73	-	69.24	**69.98**	62.75	-	66.30	40.69	**75.00**

	XGLM				mT0				XLM-R					
	RoSPrompt	*NPPrompt*	*NPPrompt-t*	*SPT*	*RoSPrompt*	*NPPrompt*	*NPPrompt-t*	*SPT*	*RoSPrompt*	*NPPrompt*	*NPPrompt-t*	*SPT*	*Llama3.1-8B*	*Phi-3.5-mini*
nob_Latn					**75.25**	62.75	58.82	70.59	**70.96**	64.71	38.73	62.62	51.47	**73.53**
npi_Deva					**73.04**	62.25	-	70.34	**69.85**	66.18	-	65.44	25.98	**45.10**
nya_Latn					**70.59**	61.76	-	69.24					15.69	**38.24**
pan_Guru					**72.30**	61.76	-	71.94					**9.31**	2.94
pbt_Arab					66.91	55.88	-	**67.65**	**65.07**	62.75	-	62.62	23.04	**38.24**
pes_Arab					**75.00**	60.29	-	68.75	**68.75**	65.69	-	61.03	45.10	**52.94**
plt_Latn					**67.28**	55.88	-	66.54	**62.99**	54.90	-	48.04	13.24	**42.65**
pol_Latn					**75.98**	62.75	54.41	74.14	**66.42**	64.71	47.55	64.95	57.84	**78.43**
por_Latn	**72.92**	64.71	50.49	66.18	**76.35**	67.16	37.25	74.02	**70.34**	66.67	57.35	68.63	60.78	**77.45**
quy_Latn	**45.71**	44.12	-	32.48									14.71	**41.18**
ron_Latn					72.43	64.22	37.25	**75.86**	**71.32**	68.63	56.86	62.01	50.00	**72.55**
rus_Cyrl	**70.22**	60.78	57.84	64.95	**75.49**	63.73	61.27	72.55	68.75	**69.12**	65.69	67.28	59.80	**75.98**
san_Deva									**64.34**	62.25	-	59.93	18.63	**41.18**
sin_Sinh					**72.18**	60.29	-	69.12	**68.14**	62.25	-	60.05	**4.41**	3.92
slk_Latn					**70.96**	62.25	31.86	69.61	67.77	**69.12**	65.20	63.85	44.61	**71.57**
slv_Latn					72.43	62.25	52.94	**73.53**	**65.93**	64.71	61.27	62.62	40.20	**67.65**
smo_Latn					60.91	50.49	49.51	**63.48**					13.24	**33.33**
sna_Latn					**67.03**	59.31	48.53	64.71					15.20	**39.71**
snd_Arab					**65.32**	58.33	43.63	63.85	**67.40**	58.82	16.67	55.15	26.47	**32.84**
som_Latn					59.44	54.90	36.76	**60.05**	**59.19**	55.39	39.71	52.21	12.75	**36.76**
sot_Latn					**70.34**	58.33	46.57	67.40					14.71	**34.80**
spa_Latn	**74.39**	60.29	44.12	55.88	**78.19**	67.65	56.86	74.02	**66.67**	65.69	46.57	68.50	68.63	**80.39**
srp_Cyrl					**76.84**	64.22	64.71	71.08	**69.36**	62.75	57.35	59.56	46.08	**60.78**
sun_Latn					**73.04**	60.29	54.90	70.83	**68.14**	67.16	57.35	67.40	17.16	**62.25**
swe_Latn					**72.92**	62.25	55.88	70.96	**71.81**	67.16	57.35	66.67	54.90	**75.49**
swh_Latn	**65.32**	61.76	58.82	55.51	**71.69**	61.76	49.51	68.87	**65.69**	62.75	50.00	55.27	28.43	**44.12**
tam_Taml	**67.65**	60.78	-	61.40	**76.10**	62.75	-	71.32	**66.05**	63.24	-	63.48	10.29	**15.20**
tel_Telu	**62.25**	55.39	-	54.90	**75.12**	63.24	-	71.94	**67.77**	63.73	-	63.60	**6.86**	2.45
tgk_Cyrl					**70.47**	60.29	57.84	66.79					23.04	**35.78**
tgl_Latn					**71.94**	63.73	59.80	68.63					41.67	**70.10**
tha_Thai	**67.65**	58.82	53.92	59.31	**73.28**	62.25	63.73	70.96	**69.12**	65.69	54.90	68.87	52.45	**54.90**
tur_Latn	**72.55**	62.25	-	60.42	**74.39**	61.27	-	71.69	**67.89**	65.20	-	63.11	42.16	**70.10**
uig_Arab									**67.16**	62.25	-	59.93	**15.20**	9.31
ukr_Cyrl					**73.65**	62.75	63.24	71.32	67.89	**69.61**	51.96	66.79	50.00	**69.61**
urd_Arab	**67.65**	56.86	-	55.27	**71.81**	59.80	-	69.00	**70.83**	61.76	-	65.69	**53.92**	40.69
uzn_Latn					**74.63**	63.24	-	69.49	**69.00**	64.71	-	59.56	22.55	**46.08**
vie_Latn	**69.61**	66.18	58.33	59.31	**72.79**	62.25	62.75	70.34	66.79	**68.14**	54.90	62.50	44.12	**69.61**
xho_Latn					**69.36**	61.27	50.49	67.89	**52.82**	50.49	26.47	49.39	16.18	**42.16**
ydd_Hebr					**60.66**	53.43	43.14	59.80	**62.25**	51.47	30.39	43.14	16.67	**18.14**
yor_Latn					55.88	49.51	40.69	**59.07**					14.22	**30.88**
zho_Hans	**73.53**	54.90	44.61	46.69	**77.21**	63.24	58.33	74.88	**68.87**	63.24	56.86	65.56	54.90	**78.92**
zho_Hant					**74.14**	65.69	61.76	70.96	**70.22**	66.18	54.90	61.89	48.53	**79.41**
zsm_Latn					**73.16**	65.69	55.88	69.36	**69.61**	65.69	58.33	60.42	38.73	**74.51**
zul_Latn					**67.77**	58.82	55.88	65.44					18.63	**40.20**

Table 10: Comparison of accuracy scores on the SIB-200 dataset between RoSPrompt and different baselines across all supported languages. For each language, the best overall result is underlined, and the best result within each column group is highlighted in **bold**.