Label-Aware Automatic Verbalizer for Few-Shot Text Classification in Mid-To-Low Resource Languages

Thanakorn Thaminkaew[1], **Piyawat Lertvittayakumjorn**[2], **Peerapon Vateekul**[1]*
[1] Department of Computer Engineering, Faculty of Engineering,
Chulalongkorn University, Thailand
[2] Google, United States
6472031921@student.chula.ac.th, piyawat@google.com, peerapon.v@chula.ac.th

Abstract

Prompt-based learning has shown its effectiveness in few-shot text classification. A key factor in its success is a verbalizer, which translates output from a language model into a predicted class. Notably, the simplest and widely acknowledged verbalizer employs manual labels to represent the classes. However, manual selection may not yield the optimal words for a given language model, potentially leading to subpar classification performance, especially in mid-to-low resource languages with weaker language models. Therefore, we propose Label-Aware Automatic Verbalizer (LAAV), effectively augmenting manual labels for improved few-shot classification results. Specifically, we utilize the label name along with the conjunction "and" to induce the model to generate more effective words for the verbalizer. Experimental results on four mid-to-low resource Southeast Asian languages demonstrate that LAAV significantly outperforms existing verbalizers.

1 Introduction

In recent years, we have seen many promising applications of *prompt-based learning* for text classification (Schick and Schütze, 2021b; Wang et al., 2022b; Zhang et al., 2022; Hu et al., 2022). While the traditional approach trains or fine-tunes a machine learning model to directly predict a class for an input text, the prompt-based approach fits the input text into a *template* that has some slots to be filled. Next, it asks a language model (LM)[1] to fill in the slots and then translates what the model filled to be a predicted class (Liu et al., 2023). To predict sentiment in a movie review like "Great movie!" as positive or negative, we may prompt a masked LM with "Great movie! It was `[MASK]`." The model may predict the word "fun" for the `[MASK]` token, and we can apply a function, so-called a *verbalizer*, to map "fun" to the positive class.

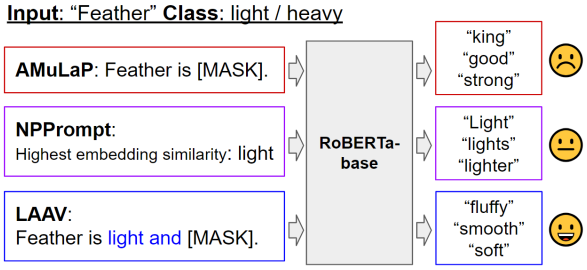

Figure 1: Comparing LAAV with AMuLaP and NPPrompt in the search for class representative tokens. This example can be applied to other languages.

Certainly, the success of a prompt-based text classifier heavily relies on its verbalizer. Schick and Schütze (2021a) proposed PET, which manually chooses a word to represent each class. During inference, it compares the likelihood of those words at the `[MASK]` token (as predicted by the LM) to find the most probable class. In contrast, Wang et al. (2022a) proposed AMuLaP, which represents each class with a set of words, automatically derived from those predicted by the LM for training examples. Zhao et al. (2023) proposed NPPrompt, which represents each class using a set of tokens with the highest embedding similarity to the manual class label. Its performance, therefore, relies solely on the LM's embedding space. Additionally, there is no guarantee that the chosen words will be relevant to the classes of interest, potentially affecting the classifier's performance. This issue disproportionately impacts mid-to-low resource languages, where the LM may have received less comprehensive training data (Hangya et al., 2022; Conneau et al., 2019).

In Figure 1 (top), to predict whether an object "Feather" is light with a prompt "Feather is `[MASK]`.", the LM suggests "king", "good", and "strong", which are irrelevant to the task but used by AMuLaP to construct the verbalizer. Meanwhile, as shown in Figure 1 (middle), NPPrompt

* Corresponding author

[1]Generally, masked LMs are preferred for classification tasks due to their close alignment with the pre-training task (Liu et al., 2023).

Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop), pages 195–203

suggests "Light", "lights", and "lighter", which are variations related to the class "light" but hardly provide additional information about the class.

With the smaller size of LMs, particularly for mid-to-low languages, predicting relevant words becomes more challenging. (Nguyen and Nguyen, 2020). In this paper, we propose **LAAV** (**L**abel-**A**ware **A**utomatic **V**erbalizer), integrating PET and AMuLaP by exploiting the class labels to induce the model to generate more relevant words for the verbalizer. As shown in Figure 1 (bottom), we could construct a better verbalizer by asking "Feather is **light and** `[MASK]`." Now, the LM suggests "fluffy", "smooth", and "soft", which are closely connected to the light class and can be used to construct an effective verbalizer. The contributions of this paper are as follows.

- We propose LAAV– a simple yet effective technique to create a reliable verbalizer for prompt-based text classification (Section 3).
- We conduct few-shot classification experiments on four datasets from four mid-to-low resource languages (Section 4), showing LAAV outperforms baselines (Section 5.1).
- We carry out an additional analysis to determine the best choice of conjunction for retrieving more related words (Section 5.2).

2 Background & Related Work

2.1 Few-shot Text Classification

Various strategies address few-shot scenarios in text classification. Meta-learning uses labeled examples from auxiliary tasks to train a model for quick adaptation to new tasks with only a few examples (Li et al., 2020; Yin, 2020). Semi-supervised or weakly-supervised approaches use extensive unlabeled data with limited labeled data to enhance the model's performance (Li et al., 2018; Duarte and Berton, 2023). In-context learning (ICL) includes a few labeled examples within a prompt for querying large pre-trained LMs to get the classification (Brown et al., 2020; Lin et al., 2021). Our paper adopts the prompt-based learning approach, which involves template design, verbalizer, and model fine-tuning. This approach has proven efficient in model training (Zhao et al., 2023; Schick and Schütze, 2021a) and is beneficial for few-shot classification in mid-to-low resource languages, where auxiliary tasks, unlabeled data, and large pre-trained LMs are limited.

2.2 Verbalizers for Prompt-Based Learning

The easiest way to construct a verbalizer is to manually select a representative word for each class, as in PET (Schick and Schütze, 2021a). However, manual selection could be laborious and does not ensure optimal word choice for the chosen LM. Hambardzumyan et al. (2021) introduced trainable continuous tokens, known as a soft verbalizer, for automating class representations. However, these tokens may not represent actual words, hindering model debugging and improvement.

Meanwhile, our study, along with others, favors discrete verbalizers due to their interpretability. Schick et al. (2020) searched for the best word to represent each class by maximizing the likelihood of the training data. AMuLaP (Wang et al., 2022a) does the same but represents each class by multiple words to reduce the effects of noise in the data. NPPrompt (Zhao et al., 2023) utilizes a set of tokens that have the closest embedding similarity to the manual label to represent each class. However, its effectiveness is strongly dependent on the quality of the LM's embedding space, which may not be effective for mid-to-low resource languages or suitable for classification task. Additionally, it overlooks the input text, potentially leading to problems with polysemous words. Since our work is based on AMuLaP, the next section explores its details.

2.3 AMuLaP

For a text classification task aiming to classify an input text x to a class $y \in Y$, AMuLaP represents each class y_i with a set of k tokens, denoted as $\mathcal{S}(y_i)$. These tokens are selected from the sub-word vocabulary $\mathcal{V}_M$ of the language model M it prompts. To construct $\mathcal{S}(y_i)$, it applies a template T to all training examples x of which the ground truth label is y_i. One example is $T(x) = [x]\ It\ was$ `[MASK]` for the classification task in the Introduction. Then it lets M predict the probability of each $v \in \mathcal{V}_M$ for the `[MASK]` of these $T(x)$s. The score of token v for class y_i is

$$s(v, y_i) = \sum_{(x,y_i)\in D} p_M(\texttt{[MASK]} = v|T(x)) \quad (1)$$

where D is the training set and p_M is the probability predicted by M. $\mathcal{S}(y_i)$ is then defined as a set of k tokens with the highest $s(v, y_i)$.

To ensure that each token v is assigned to only one class, AMuLaP calculates its score for every $y \in Y$ and assigns it to the class y_i where

$y_i = \arg\max_{y \in Y} s(v, y)$. After that, the LM is fine-tuned on D using the cross-entropy loss. Specifically, the log-probability of class y_i for an input x is

$$L(y_i|x) = \frac{1}{k} \sum_{v \in \mathcal{S}(y_i)} \log p_M(\texttt{[MASK]} = v|T(x)) \quad (2)$$

The cross-entropy loss will be calculated from $L(y_i|x)$ for all $y_i \in Y$ and all $x \in D$ as

$$loss = - \sum_{(x,y) \in D} \sum_{y_i \in Y} I(y, y_i) \cdot L(y|x) \quad (3)$$

where $I(y, y_i) = 1$ if $y = y_i$; otherwise, 0.

Finally, during validation and testing, the predicted label $\hat{y}$ for an input x is simply $\arg\max_{y_i \in Y} L(y_i|x)$.

3 Label-Aware Automatic Verbalizer

As illustrated in Figure 1, the words in $\mathcal{S}(y_i)$, selected by AMuLaP, could be unrelated to their corresponding class. So, when constructing $\mathcal{S}(y_i)$, our method LAAV integrates the label name of y_i into the template T, using a conjunction. This helps induce M to predict words that are related to y_i. Our choice for the conjunction is "and" because it serves to connect words or phrases with the same grammatical category and similar meaning. Also, "and" is one of the most widely used conjunctions in many languages (Davies, 2011). As a result, our LAAV template for creating $\mathcal{S}(y_i)$ is

$$T_{y_i}(x) = [x]\ It\ was\ [y_i]\ \textbf{and}\ \texttt{[MASK]}$$

Note that we will explore other conjunction options in Section 5.2. Now, the score of token v for class y_i for LAAV will be

$$s(v, y_i) = \sum_{(x,y_i) \in D} p_M(\texttt{[MASK]} = v|T_{y_i}(x)) \quad (4)$$

Since the objective of the LAAV template T_{y_i} is solely for seeking better representative words for each class, we use the original template T without the conjunction during training and inference.

4 Experiments

4.1 Datasets and Pre-trained Models

We conducted experiments on four datasets from four Southeast Asia languages. These include sentiment analysis datasets: SmSA (Indonesian) (Wilie et al., 2020a), Students' Feedback (Vietnamese) (Van Nguyen et al., 2018), Wisesight sentiment (Thai) (Suriyawongkul et al., 2019), and Shopee Reviews (Tagalog) (Riego, 2023). The LAAV templates, the class labels, and other details of each dataset are reported in Appendix A.

The pre-trained LMs used in this paper are the base versions of IndoBERT (Wilie et al., 2020b), Tagalog RoBERTa (Cruz and Cheng, 2021), WangchanBERTa (Lowphansirikul et al., 2021), and PhoBERT (Nguyen and Nguyen, 2020) for Indonesian, Tagalog, Thai, and Vietnamese, respectively. Additionally, we employed SeaLLM-7B-v2.5 (Nguyen et al., 2023), an open-source large language model (LLM) designed for Southeast Asia languages, for an in-context learning (ICL) baseline.

4.2 Implementation Details

In a few-shot scenario, we randomly selected 1, 2, 4, or 8 samples per class for both the training and validation splits. Since we do not have a sizable development set for optimizing hyperparameters, we depend on related work to guide us in selecting the appropriate hyperparameters. All text inputs were limited to 500 characters. During training, we used Adam optimizer (Kingma and Ba, 2014) with a learning rate of 1e-5 to optimize the loss function. To prevent overfitting, we employed early stopping, limiting training to a maximum of 100 epochs. This process was repeated five times with different seeds for robustness. We set k = 32 for all experiments, with the best determination of k detailed in Appendix B. Our models were implemented using PyTorch (Paszke et al., 2019) and the OpenPrompt (Ding et al., 2021) libraries, and trained on a Tesla P100 PCIe 16 GB.

4.3 Baselines

We evaluated our method by comparing it to **Traditional Fine-tuning** (i.e., plugging a linear classification layer of top of the [CLS] embedding of the LM and fine-tuning the whole model) and six recent methods including five verbalizer methods and one LLM-ICL method: (1) **PET** manually selecting a token to represent each class (Schick and

Sample Size	1	2	4	8
SmSA (Indonesian)				
Traditional FT	42.5 (7.1)	43.9 (3.6)	48.1 (7.4)	52.2 (6.6)
PET	34.5 (9.8)	39.8 (7.5)	49.1 (8.4)	53.0 (7.0)
WARP$_V$	37.5 (9.1)	43.9 (5.8)	50.9 (7.2)	52.2 (5.2)
PETAL	35.5 (8.8)	44.1 (6.9)	53.8 (6.2)	52.1 (8.2)
AMuLaP	38.7 (10.4)	44.5 (4.9)	58.9 (4.6)	58.3 (4.4)
NPPrompt	22.6 (6.2)	41.7 (7.1)	50.7 (6.4)	51.6 (8.4)
LLM-ICL	**49.4 (2.4)**	**54.1 (8.0)**	50.5 (1.6)	51.9 (0.9)
LAAV (ours)	45.3 (9.9)*	46.7 (4.7)	**61.1 (7.6)***	**58.5 (10.9)***
Shopee Reviews (Tagalog)				
Traditional FT	17.3 (4.5)	21.7 (3.9)	24.4 (3.8)	28.1 (5.0)
PET	18.3 (2.4)	20.6 (1.9)	22.8 (1.2)	24.0 (1.8)
WARP$_V$	18.6 (2.4)	23.0 (1.3)	25.1 (2.1)	28.1 (2.7)
PETAL	17.8 (4.0)	26.9 (1.5)	26.8 (3.8)	30.2 (1.6)
AMuLaP	21.4 (6.0)	27.2 (3.5)	28.9 (5.8)	32.4 (3.3)
NPPrompt	13.9 (7.0)	18.0 (6.5)	17.9 (7.4)	26.9 (5.0)
LLM-ICL	**28.1 (0.7)**	28.7 (1.4)	28.1 (1.3)	28.8 (1.2)
LAAV (ours)	25.5 (5.0)*	**30.5 (1.3)***	**31.6 (3.7)***	**32.6 (2.8)***
Wisesight sentiment (Thai)				
Traditional FT	20.7 (4.3)	24.2 (5.5)	28.2 (4.2)	29.6 (5.4)
PET	23.8 (4.4)	31.0 (7.2)	34.5 (6.5)	41.0 (5.5)
WARP$_V$	23.4 (5.7)	27.2 (5.9)	30.8 (4.2)	37.7 (2.8)
PETAL	20.5 (2.0)	26.5 (7.6)	30.8 (4.4)	37.1 (2.8)
AMuLaP	21.1 (5.4)	28.0 (10.6)	32.3 (5.6)	37.4 (8.9)
NPPrompt	25.3 (2.3)	26.2 (9.1)	31.0 (7.8)	37.0 (4.6)
LLM-ICL	17.7 (2.0)	19.1 (1.3)	21.4 (2.6)	23.2 (1.9)
LAAV (ours)	**25.9 (5.9)**	**31.5 (7.6)**	**38.1 (4.5)**	**42.1 (5.8)**
Students' Feedback (Vietnamese)				
Traditional FT	39.5 (7.1)	47.3 (8.7)	51.2 (10.1)	62.6 (1.6)
PET	49.3 (13.3)	60.7 (2.1)	65.5 (3.0)	68.7 (2.8)
WARP$_V$	23.3 (3.5)	47.8 (7.6)	51.4 (8.3)	57.2 (2.6)
PETAL	21.1 (9.2)	38.3 (6.8)	49.1 (8.9)	57.7 (4.3)
AMuLaP	38.7 (13.6)	47.0 (10.9)	55.6 (11.2)	64.6 (2.1)
NPPrompt	25.5 (6.1)	39.5 (11.8)	37.0 (17.4)	40.0 (17.2)
LLM-ICL	41.5 (0.7)	41.5 (0.8)	41.5 (0.9)	41.9 (1.3)
LAAV (ours)	**53.6 (10.7)**	**61.7 (3.8)**	**67.9 (2.8)***	**69.5 (1.9)**

Table 1: Macro F1 results along with their standard deviations (in parentheses) tested on four datasets. The best results are marked in **bold**. An asterisk (*) indicates that our method, LAAV, demonstrates a statistically significant improvement over the strongest baseline, PET, based on paired t-tests, as shown in Appendix D.

Schütze, 2021a), (2) the verbalizer of WARP, denoted as **WARP$_V$**, representing each class with a trained continuous vector (Hambardzumyan et al., 2021), (3) **PETAL** searching for the most suitable representative token (Schick et al., 2020), and (4) **AMuLaP** searching for multiple suitable representative tokens using an unmodified template (Wang et al., 2022a). (5) **NPPrompt** using a set of tokens with the highest embedding similarity to the manual label as representative tokens (Zhao et al., 2023). (6) **LLM-ICL**: Unlike other baselines that involve fine-tuning, we augmented the prompt template with examples for each few-shot learning scenario, enabling ICL (Brown et al., 2020). Refer to Appendix C for the adapted prompt template suitable for LLM. We employed the OpenPrompt library for WARP$_V$ (SoftVerbalizer) and PETAL (AutomaticVerbalizer), while implementing other baselines manually in PyTorch.

5 Results and Additional Analyses

5.1 Comparison to the Baselines

Table 1 shows the results of our method compared to the baselines. The LLM-ICL method shows promise in extreme few-shot settings but struggles with additional examples. PET, however, is the strongest baseline across all datasets and sample sizes, highlighting the effectiveness of using label names as representative tokens. Nevertheless, fine-tuning LMs through prompt-based learning, as demonstrated by our proposed method LAAV, continues to show adaptability and efficacy across various learning contexts. For example, in the 4-shot settings, LAAV consistently outperforms other baselines, achieving a 5.7% absolute improvement in Macro F1 scores over PET and a 6.7% improvement over AMuLaP across four datasets. This highlights LAAV's superior performance, notably in selecting top representative words.

For instance, Table 2, presents the top 3 (out of 32) representative tokens for the Wisesight sentiment dataset as selected and ranked by different verbalizers. AMuLaP sometimes selects tokens seemingly unrelated to classes, such as associating "constructive" and "psychology" with the "negative" class, while associating "philosophy" and "theory" with the "positive" class. In contrast, NPPrompt uses PLM embeddings to choose words closely aligned with label meanings, although some selections, like the top 3 tokens for the "question" class, can be repetitive. LAAV tends to select words closely related to the label names; for example, "selfish", "terrible", and "rude" are top tokens for the "negative" class. This illustrates how incorporating label names with "and" can generate more effective verbalizations.

5.2 Choices of conjunction

While we used "and" as the conjunction of LAAV templates so far, this section aims to explore whether there are other promising conjunction choices we missed. Hence, we designed the following conjunction search process. First, we used AMuLaP to find the initial $\mathcal{S}(y_i)$ of each class. Then, we applied the template

$$T^S_{y_i}(x) = [x]\ It\ was\ [y_i]\ \texttt{[MASK]}\ [v]$$

for all $v \in \mathcal{S}(y_i)$, to every training examples x labeled y_i. Basically, $T^S_{y_i}$ asks the LM to predict

Class	Model	Top-3 Words
ลบ (negative)	AMuLaP	สร้างสรรค์ (constructive), จิตวิทยา (psychology), ธุรกิจ (business)
	NPPrompt	ลบ (negative), บวก (positive), ปิด (close)
	LAAV	เห็นแก่ตัว (selfish), แย่มาก (terrible), หยาบคาย (rude)
กลาง (neutral)	AMuLaP	สัญลักษณ์ (symbol), พาณิชย์ (commerce), ประจักษ์ (obvious)
	NPPrompt	กลาง (neutral), ตรงกลาง (middle), กลางๆ (neutral)
	LAAV	กลาง (neutral), กลางๆ (neutral), ลึก (deep)
บวก (positive)	AMuLaP	วิชาการ (academic), ปรัชญา (philosophy), ทฤษฎี (theory)
	NPPrompt	บวก (positive), บวกกับ (in addition to), ลบ (negative)
	LAAV	ชัดเจน (clear), สร้างสรรค์ (constructive), ถูกต้อง (correct)
คำถาม (question)	AMuLaP	ค่ะ (yes), นะคะ (please), ส่วนตัว (personal)
	NPPrompt	คำถาม (question), คำถามที่ (question that), มีคำถาม (have question)
	LAAV	เหตุผล (reason), ประสบการณ์ (experience), คำถาม (question)

Table 2: Comparison of the top-3 words in 4-shot settings to represent each class in Wisesight sentiment dataset.

Dataset	Top Translated Words	Automatic	"and"
SmSA	exchange, dough, mopped	42.7 (8.3)	**45.3 (9.9)**
Shopee Reviews	already, in, just	20.6 (3.2)	**25.5 (5.0)**
Wisesight sentiment	really, very, yes	24.8 (3.8)	**25.9 (5.9)**
Students' Feedback	of, for, and	43.7 (6.5)	**53.6 (10.7)**

Table 3: Comparison of Macro F1 results between automatic search and "and" conjunction in 1-shot setting. The best results are marked in **bold**.

a token that can well connect y_i to v, having the potential to be the conjunction in LAAV template.

Table 3 shows the top three English-translated words from language-specific LMs, selected by the highest token score using Equation 1 with the template $T^S_{y_i}(x)$ instead of the original $T(x)$. Conjunctions in the Students' Feedback dataset exhibits coherence, attributed to LMs favoring adjectives for effective conjunctions. Ultimately, **"and"** consistently yields the best results across datasets, supporting our initial LAAV template design.

6 Conclusion

Our method, LAAV, constructs a better verbalizer by exploiting class labels to collect more relevant words. As shown in the experiments, LAAV outperforms other existing verbalizers in few-shot text classification across four languages, even surpassing LLM with in-context learning. Our comprehensive analysis highlights "and" as a particularly effective conjunction for retrieving words that exhibit high discriminative power crucial for enhancing text classification performance.

Limitations

We only focused on improving the selection of words to represent each label with a fixed prompt template. Applying a tunable continuous template or a more specific discrete template may also reduce the ambiguity of the input and further improve the prompt-based learning results. In addition, with limited resources, we decided to explore experiments using the base version of the LMs. Fine-tuning larger LMs using parameter-efficient techniques may lead to different results. Nevertheless, parameter-efficient techniques such as Low-Rank Adaptation (Hu et al., 2021) can be implemented on top of the prompt-based learning approach presented in this paper.

Ethics Statement

Our approach involves fine-tuning LMs through prompt-based learning, utilizing openly accessible datasets and models from the Hugging Face Hub. To ensure reliability and neutrality, we conducted five runs with varied seeds for each experiment. Detailed information on model parameters and computing infrastructure is openly disclosed to promote reproducibility. While our method does not introduce new ethical concerns beyond those associated with LMs, we acknowledge the potential for biases. Users are advised to use our method cautiously and thoroughly assess model outputs before deploying them in real-world applications.

Acknowledgements

The authors would like to thank Kornraphop Kawintiranon for his constructive feedback on the early draft of this paper. The authors would also like to thank the anonymous reviewers for their helpful comments.

References

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Jan Christian Blaise Cruz and Charibeth Cheng. 2021. Improving large-scale language models and resources for filipino. *arXiv preprint arXiv:2111.06053*.

Mark Davies. 2011. Word frequency data from the corpus of contemporary american english (coca).

Ning Ding, Shengding Hu, Weilin Zhao, Yulin Chen, Zhiyuan Liu, Hai-Tao Zheng, and Maosong Sun. 2021. Openprompt: An open-source framework for prompt-learning. *arXiv preprint arXiv:2111.01998*.

José Marcio Duarte and Lilian Berton. 2023. A review of semi-supervised learning for text classification. *Artificial Intelligence Review*, pages 1–69.

Karen Hambardzumyan, Hrant Khachatrian, and Jonathan May. 2021. WARP: Word-level Adversarial ReProgramming. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4921–4933, Online. Association for Computational Linguistics.

Viktor Hangya, Hossain Shaikh Saadi, and Alexander Fraser. 2022. Improving low-resource languages in pre-trained multilingual language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11993–12006, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

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.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Jing Li, Billy Chiu, Shanshan Feng, and Hao Wang. 2020. Few-shot named entity recognition via meta-learning. *IEEE Transactions on Knowledge and Data Engineering*, 34(9):4245–4256.

Penghua Li, Fen Zhao, Yuanyuan Li, and Ziqin Zhu. 2018. Law text classification using semi-supervised convolutional neural networks. In *2018 Chinese control and decision conference (CCDC)*, pages 309–313. IEEE.

Xi Victoria Lin, Todor Mihaylov, Mikel Artetxe, Tianlu Wang, Shuohui Chen, Daniel Simig, Myle Ott, Naman Goyal, Shruti Bhosale, Jingfei Du, et al. 2021. Few-shot learning with multilingual language models. *arXiv preprint arXiv:2112.10668*.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Comput. Surv.*, 55(9).

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 preprint arXiv:1907.11692*.

Lalita Lowphansirikul, Charin Polpanumas, Nawat Jantrakulchai, and Sarana Nutanong. 2021. Wangchanberta: Pretraining transformer-based thai language models.

Meta. 2024. Introducing meta llama 3: The most capable openly available llm to date. *Meta AI Blog*.

Dat Quoc Nguyen and Anh Tuan Nguyen. 2020. PhoBERT: Pre-trained language models for Vietnamese. *Findings of EMNLP*.

Xuan-Phi Nguyen, Wenxuan Zhang, Xin Li, Mahani Aljunied, Qingyu Tan, Liying Cheng, Guanzheng Chen, Yue Deng, Sen Yang, Chaoqun Liu, et al. 2023. Seallms–large language models for southeast asia. *arXiv preprint arXiv:2312.00738*.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

Neil Riego. 2023. shopee-reviews-tl-stars (revision d096f40).

Timo Schick, Helmut Schmid, and Hinrich Schütze. 2020. Automatically identifying words that can serve as labels for few-shot text classification. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5569–5578, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Timo Schick and Hinrich Schütze. 2021a. Exploiting cloze-questions for few-shot text classification and natural language inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 255–269, Online. Association for Computational Linguistics.

Timo Schick and Hinrich Schütze. 2021b. It's not just size that matters: Small language models are also few-shot learners. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2339–2352, Online. Association for Computational Linguistics.

Arthit Suriyawongkul, Ekapol Chuangsuwanich, Pattarawat Chormai, and Charin Polpanumas. 2019. Pythainlp/wisesight-sentiment: First release.

Kiet Van Nguyen, Vu Duc Nguyen, Phu XV Nguyen, Tham TH Truong, and Ngan Luu-Thuy Nguyen. 2018. Uit-vsfc: Vietnamese students' feedback corpus for sentiment analysis. In *2018 10th international conference on knowledge and systems engineering (KSE)*, pages 19–24. IEEE.

Han Wang, Canwen Xu, and Julian McAuley. 2022a. Automatic multi-label prompting: Simple and interpretable few-shot classification. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5483–5492, Seattle, United States. Association for Computational Linguistics.

Jianing Wang, Chengyu Wang, Fuli Luo, Chuanqi Tan, Minghui Qiu, Fei Yang, Qiuhui Shi, Songfang Huang, and Ming Gao. 2022b. Towards unified prompt tuning for few-shot text classification. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 524–536, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Bryan Wilie, Karissa Vincentio, Genta Indra Winata, Samuel Cahyawijaya, Xiaohong Li, Zhi Yuan Lim, Sidik Soleman, Rahmad Mahendra, Pascale Fung, Syafri Bahar, and Ayu Purwarianti. 2020a. IndoNLU: Benchmark and resources for evaluating Indonesian natural language understanding. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 843–857, Suzhou, China. Association for Computational Linguistics.

Bryan Wilie, Karissa Vincentio, Genta Indra Winata, Samuel Cahyawijaya, Xiaohong Li, Zhi Yuan Lim, Sidik Soleman, Rahmad Mahendra, Pascale Fung, Syafri Bahar, et al. 2020b. Indonlu: Benchmark and resources for evaluating indonesian natural language understanding. *arXiv preprint arXiv:2009.05387*.

Wenpeng Yin. 2020. Meta-learning for few-shot natural language processing: A survey. *arXiv preprint arXiv:2007.09604*.

Haoxing Zhang, Xiaofeng Zhang, Haibo Huang, and Lei Yu. 2022. Prompt-based meta-learning for few-shot text classification. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1342–1357, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28.

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.

A Dataset Details

Table 4 presents the dataset statistics alongside their respective templates (LAAV and AMuLaP), labels, and translated label names. Note that Shopee Reviews originally has five classes [1,..,5] which were manually mapped to textual labels ["very bad", ..., "excellent"]. Our templates in each language are based on the same initial template, which we first created in English and then translated using Google Translate.

All datasets referenced are publicly accessible via the URLs provided below.

- SmSA: https://github.com/IndoNLP/indonlu/tree/master/dataset/smsa_doc-sentiment-prosa
- Shopee Reviews: https://huggingface.co/datasets/scaredmeow/shopee-reviews-tl-stars
- Wisesight sentiment: https://huggingface.co/datasets/wisesight_sentiment
- Students' Feedback: https://huggingface.co/datasets/uit-nlp/vietnamese_students_feedback

SmSA (Indonesian)	Label	[negatif, netral, positif] => [negative, neutral, positive]
	LAAV Template	" komentar ini adalah + [y]+ "dan" + [MASK]."
	AMuLaP / Training Template	" komentar ini adalah [MASK]."
Shopee Reviews (Tagalog)	Label	[napakasama, masama, karaniwan, mahusay, napakahusay] => [very bad, bad, average, good, excellent]
	LAAV Template	" ito ay + [y] + "at" + <mask> reivew."
	AMuLaP / Training Template	" ito ay <mask> reivew."
Wisesight Sentiment (Thai)	Label	[ลบ, กลาง, บวก, คำถาม] => [negative, neutral, positive, question]
	LAAV Template	"เป็นความเห็นเชิง + [y] + "และ" + <mask>"
	AMuLaP / Training Template	"เป็นความเห็นเชิง<mask>"
Students' Feedback (Vietnamese)	Label	[tiêu cực, trung lập, tích cực] => [negative, neutral, positive]
	LAAV Template	" Nó là + [y] + "và" + <mask>."
	AMuLaP / Training Template	" Nó là <mask>."

Table 4: Details of the datasets along with their templates and labels.

Sample Size	1	2	4	8
SmSA (Indonesian)				
1	41.7 (2.1)	40.9 (6.5)	59.9 (10.0)	58.6 (5.6)
4	44.2 (7.4)	46.6 (11.2)	58.0 (8.8)	58.9 (6.9)
8	41.1 (10.3)	45.8 (6.6)	59.4 (9.3)	55.9 (10.5)
16	41.9 (11.5)	43.9 (8.5)	61.0 (6.7)	57.6 (10.0)
24	44.2 (10.3)	46.3 (4.9)	**61.1 (6.0)**	**59.1 (7.6)**
32	**45.3 (9.9)**	**46.7 (4.7)**	**61.1 (7.6)**	58.5 (10.9)
40	45.2 (9.3)	**46.7 (4.1)**	60.9 (7.3)	58.3 (12.0)
Shopee Reviews (Tagalog)				
1	19.1 (2.1)	26.6 (1.1)	25.7 (4.7)	30.6 (3.1)
4	22.9 (3.6)	26.6 (4.2)	29.4 (2.8)	32.8 (2.0)
8	24.9 (3.5)	28.9 (2.2)	30.7 (3.7)	32.9 (2.6)
16	24.7 (3.0)	29.4 (2.6)	31.4 (3.5)	**33.3 (2.2)**
24	24.8 (5.1)	29.7 (2.4)	31.1 (3.4)	33.0 (2.4)
32	**25.5 (5.0)**	**30.5 (1.3)**	**31.6 (3.7)**	32.6 (2.8)
40	23.1 (6.8)	30.2 (1.2)	31.5 (3.5)	32.0 (3.1)
Wisesight sentiment (Thai)				
1	23.0 (4.9)	29.8 (7.4)	32.9 (5.8)	38.1 (4.4)
4	25.8 (3.3)	29.9 (8.8)	34.4 (5.5)	42.0 (3.7)
8	25.7 (4.3)	34.1 (7.8)	37.5 (4.5)	41.3 (5.5)
16	**25.9 (4.8)**	33.9 (6.0)	36.4 (6.0)	40.1 (5.6)
24	24.3 (5.2)	**34.3 (5.0)**	35.1 (4.7)	41.9 (6.1)
32	**25.9 (5.9)**	31.5 (7.6)	**38.1 (4.5)**	**42.1 (5.8)**
40	**25.9 (5.8)**	34.2 (7.4)	38.0 (5.6)	37.4 (9.0)
Students' Feedback (Vietnamese)				
1	39.7 (10.5)	50.7 (8.5)	64.1 (4.2)	64.7 (3.4)
4	50.4 (11.5)	55.0 (4.4)	64.3 (0.9)	68.4 (3.9)
8	47.8 (11.5)	60.0 (3.8)	65.6 (3.0)	68.6 (2.7)
16	49.2 (12.5)	60.0 (4.5)	67.0 (3.3)	68.8 (2.4)
24	50.3 (11.5)	62.0 (3.4)	67.9 (3.5)	69.1 (1.7)
32	**53.6 (10.7)**	**61.7 (3.8)**	67.9 (2.8)	**69.5 (1.9)**
40	52.5 (9.2)	61.5 (2.6)	**68.1 (3.0)**	69.2 (2.1)

Table 5: Macro-F1 results along with their standard deviation in the parentheses tested on four datasets when using LAAV with a different number of tokens to represent each label varying from 1, 4, 8, 16, 24, 32, and 40. The best results are marked in **bold**.

B Number of Representative Tokens (k)

In Table 5, we investigated the impact of varying the number of representative tokens assigned to each label, denoted as k, since it influences the overall accuracy of the verbalizer. Our findings show a positive correlation between a higher number of tokens used per label and an increase in Macro-F1 score, with the optimal result at 32 tokens. As a practical suggestion, when dealing with a new dataset, we advise experimenting with a range of k values, as different k values result in variations in accuracy.

C Prompt Template Used for LLM-ICL

In Table 7, we adapted the template used in prompted fine-tuning experiments in the "Instruction" section. Then, we used the same training samples to construct in-context learning examples in the "Example" section. Finally, we included test samples in the "Question" section. Please note that the order of the in-context learning (ICL) examples will be random for every test sample.

Sample Size	1	2	4	8
SmSA (Indonesian)				
PET	34.5 (9.8)	39.8 (7.5)	49.1 (8.4)	53.0 (7.0)
LAAV	45.3 (9.9)	46.7 (4.7)	61.1 (7.6)	58.5 (10.9)
p-value	**0.0093**	0.1177	**0.0172**	**0.3758**
Shopee Reviews (Tagalog)				
PET	18.3 (2.4)	20.6 (1.9)	22.8 (1.2)	24.0 (1.8)
LAAV	25.5 (5.0)	30.5 (1.3)	31.6 (3.7)	32.6 (2.8)
p-value	**0.0080**	**0.0006**	**0.0027**	**0.0009**
Wisesight sentiment (Thai)				
PET	23.8 (4.4)	31.0 (7.2)	34.5 (6.5)	41.0 (5.5)
LAAV	25.9 (5.9)	31.5 (7.6)	38.1 (4.5)	42.1 (5.8)
p-value	0.5285	0.8966	0.2134	0.3253
Students' Feedback (Vietnamese)				
PET	49.3 (13.3)	60.7 (2.1)	65.5 (3.0)	68.7 (2.8)
LAAV	53.6 (10.7)	61.7 (3.8)	67.9 (2.8)	69.5 (1.9)
p-value	0.6499	0.7170	**0.0396**	0.4818

Table 6: Macro F1 results with their standard deviations (in parentheses) tested on four datasets, along with p-values from significance paired t-test results between our method, LAAV, and the strongest baseline, PET. Results that pass the significance paired t-tests with a p-value < 0.05 are marked in **bold**.

D Significance Tests

Table 6 presents the results of the significance tests (paired t-tests) between our method, LAAV, and the strongest baseline, PET.

The results indicate that LAAV achieves statistically significant improvements in Macro F1 scores over PET in the SmSA and Shopee Reviews datasets. However, in the Wisesight sentiment and Students' Feedback datasets, the Macro F1 scores of LAAV and PET are similar, and the differences are not statistically significant.

E Comparison on English Benchmark

While the main focus of this paper is on mid-to-low resource languages, evaluating our approaches against English benchmarks is beneficial. In this section, we chose AG's News (Zhang et al., 2015), a news classification dataset with four classes: world, sports, business, and technology. This dataset serves as a benchmark in several baseline models (Schick and Schütze, 2021a; Schick et al., 2020; Zhao et al., 2023). We conducted our experiments using the same process described in Section 4 and used RoBERTa-base (Liu et al., 2019) for its LM. Additionally, we employed Meta-Llama-3-8B (Meta, 2024), an open-source LLM, for an ICL baseline.

Table 8 presents the results of our method compared to baselines on the AG's News dataset. Our approach, LAAV, consistently outperforms other baselines. Specifically, in the 1-shot setting, our

Original Template	It was [MASK].
LLM Template	###Instruction Classify the following texts into the following categories: [label] ###Example [sample 1] + [template] + "? " + [label 1] [sample 2] + [template] + "? " + [label 2] } Random order ... ###Question [test sample 1] + [template] + "? "
LLM Template: Thai (1-shot learning)	###Instruction จำแนกข้อความต่อไปนี้เป็นหมวดหมู่ต่อไปนี้ "ลบ" "กลาง" "บวก" "คำถาม" ###Example ทำไมชั้นกินสปายแล้วปวดท้องเป็นความเห็นเชิง? ลบ ลดเหลือเท่าไรเป็นความเห็นเชิง? คำถาม นาวาร่าแดงวิ่งไหมครับเป็นความเห็นเชิง? กลาง หิวๆอยากกินเป็นความเห็นเชิง? บวก ###Question มันมีซุปดำด้วย อยากลองเป็นความเห็นเชิง?

Table 7: Details of the prompt template used for the LLM, and its application to the Wisesight sentiment dataset in a 1-shot setting. The same template was translated and applied to other datasets and settings.

Sample Size	1	2	4	8
AG's News (English)				
Traditional FT	52.6 (6.8)	72.1 (2.8)	75.6 (4.9)	81.7 (2.4)
PET	66.9 (10.5)	76.1 (6.5)	79.1 (5.1)	83.8 (1.7)
$WARP_V$	58.6 (3.0)	63.9 (7.6)	70.4 (5.6)	75.4 (3.1)
PETAL	44.0 (16.3)	66.7 (8.2)	68.1 (7.2)	79.0 (1.8)
AMuLaP	53.2 (5.1)	63.6 (7.8)	71.6 (5.9)	78.3 (2.6)
NPPrompt	44.7 (30.9)	57.5 (19.7)	79.9 (2.1)	82.7 (2.9)
LLM-ICL	65.3 (0.8)	66.3 (1.2)	64.9 (1.2)	55.7 (3.0)
LAAV (ours)	**73.0 (3.9)**	**77.5 (1.9)**	**81.1 (1.2)**	**84.1 (1.5)**

Table 8: Macro F1 results along with their standard deviations (in parentheses). The best results are marked in **bold**.

model enhances Macro F1 scores by 6.1% compared to the strongest baseline, PET. This demonstrates that while our method primarily targets improvement in mid-to-low resource languages, it is also promising in high-resource languages within the few-shot classification scenario. However, it is noteworthy that English datasets in general may not inherently require few-shot learning due to the abundance of available training examples.