Urdu-GLUE: A Comprehensive Benchmark and Dynamic Prompt-Based Fine-Tuning for Urdu Language Understanding

Anonymous ACL submission

Abstract

Language understanding benchmarks have driven significant progress in Natural Language Processing (NLP). However, most benchmarks focus on high-resource languages such as English, therefore leaving low-resource languages underserved. Despite being spoken by over 246 million people worldwide, Urdu lacks comprehensive evaluation resources. To address this gap, we introduce Urdu-GLUE, the first comprehensive benchmark for Urdu language understanding. Our comprehensive benchmark comprises ten diverse tasks, including single-sentence classification, similarity and paraphrase detection, natural language inference, question answering, and sequence labeling. To cover all the tasks mentioned in the benchmark, we created four new datasets: (1) U-CoLA for grammatical acceptability, (2) U-WNLI for Winograd-style coreference, (3) U-STS-B for semantic similarity, and (4) U-XNLI, a preprocessed XNLI dataset. To ensure quality, three native Urdu speakers fluent in English manually verified each dataset. To address the low-resource status of the Urdu language, we also introduced ADAPT (Adaptive Dynamic Prompt Template), the first dynamic prompt-based fine-tuning strategy for encoder-based models. ADAPT systematically explores various prompt templates during training and automatically identifies the most effective for inference. We evaluated multiple fine-tuning (FT) strategies, including standard FT, prompt-based FT, LoRA, QLoRA, and ADAPT, across three experimental settings, i.e., zero-shot, 16-shot, and $80/20$ split. Our experiments demonstrate that prompt-based FT methods consistently outperform standard FT in few-shot settings. Our findings provide practical insights for low-resource NLP research. To facilitate future work, we publicly[1] release all datasets, and code.

[1]https://anonymous.4open.science/r/Urdu-Glue-7D78/README.md

1 Introduction

Language understanding benchmarks have played an essential role in advancing Natural Language Processing (NLP) research. The General Language Understanding Evaluation (GLUE) benchmark, introduced by Wang et al. (2018), provides diverse tasks that test various aspects of linguistic capability. These tasks include natural language inference (NLI), semantic similarity, syntactic acceptability, and question answering. Following GLUE's work, SuperGLUE (Wang et al., 2019) introduced more challenging tasks as models approached human-level performance on the original benchmark. The NLP research community has adapted GLUE to other languages, including Chinese (CLUE) (Xu et al., 2020), Russian SuperGLUE (Shavrina et al., 2020), Korean (KLUE) (Park et al., 2021), and Indian major languages (IndicGLUE) (Kakwani et al., 2020). Recently, GermanGLUE (Pfister and Hotho, 2024) filled the gap for the German language. These benchmarks enable systematic comparison across languages and drive the development of language-specific models.

Despite these advances, Urdu the 11th most spoken language with over 246 million speakers [2], lacks a comprehensive benchmark. Recently, Arif et al. (2024) and Tahir et al. (2025) have evaluated Urdu language models on several Urdu tasks. Adeeba et al. (2025) introduced UrBLiMP, a benchmark evaluating linguistic competence through minimal pairs. These studies provide valuable insights; however, they focus on isolated tasks or model comparisons rather than general Urdu language understanding. We introduce **Urdu-GLUE**, the first comprehensive benchmark for evaluating language understanding in Urdu. We created four new datasets through manual translation (by local Urdu language experts,

[2]https://www.ethnologue.com/insights/ethnologue200/

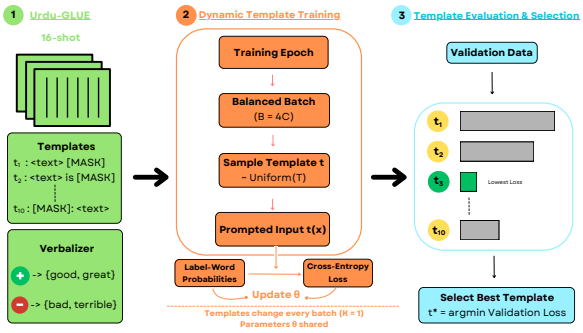

Figure 1: Overview of ADAPT (Adaptive Dynamic Prompt Template) technique.

fluent in English) from English sources (U-CoLA, U-WNLI, U-STS-B, U-XNLI), with each translation independently verified by three native Urdu speakers. This ensures high-quality, gold-standard translation/annotation. We also incorporate existing Urdu datasets for the remaining tasks to cover the general language understanding. We expand the traditional GLUE framework by including sequence labeling tasks (POS tagging and NER), recognizing their importance for comprehensive evaluation in morphologically rich languages like Urdu. The remaining six tasks utilize existing Urdu datasets (detailed in Appendix A.1). Our benchmark follows the GLUE framework and encompasses ten diverse tasks across multiple categories. Beyond evaluation, we also address the practical challenges of working with Urdu as a low-resource language.

Urdu's status compounds the challenge as a low-resource language in the NLP research community. Despite its large speaker population, the Urdu language lacks the extensive labeled datasets, pre-trained models, and available evaluation resources. This scarcity makes it essential to develop training strategies that work well with limited training data. This challenge is relevant not only for Urdu but for hundreds of other underserved languages worldwide. To fill this gap (specifically for Urdu), we introduce the Adaptive Dynamic Prompt Template (ADAPT), a novel dynamic prompt-based FT strategy as shown in Figure 1. Unlike previous prompt-based approaches, which use a single fixed template, ADAPT leverages multiple manually designed candidate prompt templates. It dynamically selects templates during training to identify the optimal prompt formulation for each task. This approach addresses a critical limitation in low-resource settings. Model performance can be susceptible to prompt formulation with limited data, and finding the right prompt through manual trial and error is expensive. To the best of our knowledge, this is the first work to apply dynamic prompt-based fine-tuning to encoder-based models, i.e., mBERT(Devlin et al., 2019) and XLM-RoBERTa(Conneau et al., 2018).

We conduct comprehensive experiments across three experimental setups: (1) zero-shot, 16-shot, and 80/20 splits. We compare standard fine-tuning vs. parameter-efficient methods (LoRA(Hu et al., 2022) and QLoRA(Dettmers et al., 2024)) on 80/20 splits. For zero-shot and 16-shots, we compared (1) prompt-based fine-tuning with fixed templates, (2) our proposed ADAPT technique, and (3) standard fine-tuning. We employ two multilingual, pre-trained language models (PLMs) bert-base-multilingual-cased (mBERT) and xlm-roberta-base (XLM-R). Our experimental results reveal several important findings. In zero-shot settings, multilingual models demonstrate reasonable performance across tasks, validating their cross-lingual transfer capabilities. However, in the 16-shot settings, our prompt-based approaches (both fixed-template prompt-based fine-tuning and ADAPT) consistently outperform standard fine-tuning across most tasks. This finding is particularly significant for low-resource languages like Urdu, where collecting large labeled datasets is expensive and time-consuming. The ability to achieve effective performance with just 16 labeled examples per class opens new possibilities for rapid NLP system development in underserved languages. In the 80/20 setting with more training data, parameter-efficient methods like LoRA and QLoRA provide competitive performance with significantly reduced computational costs, though standard fine-tuning generally achieves the highest absolute performance.

1.1 Contributions

Our contributions are fourfold:

1. We introduce the first comprehensive Urdu-GLUE benchmark (ten tasks) for Urdu.

2. We manually translate four GLUE datasets for Urdu (U-CoLA, U-WNLI, U-STS-B, and U-XNLI), with each dataset further verified by three native Urdu speakers fluent in English to ensure gold-standard quality.

3. We introduce ADAPT, the first dynamic

prompt-based FT strategy for encoder-based models.

4. Finally, we systematically compare standard FT, parameter-efficient methods (LoRA, QLoRA), prompt-based FT, and ADAPT across three experimental setups.

Our work establishes Urdu-GLUE as a benchmark for the Urdu NLP community and demonstrates that dynamic prompt selection significantly improves model performance with limited data.

2 Related Work

The GLUE benchmark (Wang et al., 2018) established a standard for evaluating language understanding across multiple tasks. Following GLUE's work, SuperGLUE (Wang et al., 2019) introduced more challenging tasks to further push the capabilities of models. The GLUE framework has been adapted to multiple languages including, Chinese (CLUE) (Xu et al., 2020), Russian (Shavrina et al., 2020), Korean (KLUE) (Park et al., 2021), Indian languages (IndicGLUE) (Kakwani et al., 2020), and German (Pfister and Hotho, 2024).

Despite having more than 246 million speakers, Urdu remains underserved in NLP research. Recent works by Arif et al. (2024) compared generalist models against specialist fine-tuned models. Tahir et al. (2025) evaluated seven LLMs on 17 Urdu tasks in zero-shot settings. Adeeba et al. (2025) introduced UrBLiMP for evaluating linguistic competence through minimal pairs in Urdu. However, these efforts focus on specific tasks or model comparisons. A comprehensive, standardized benchmark comparable to GLUE for Urdu is currently lacking, which this work aims to address.

Prompt-based learning has shown effective performance in few-shot settings (Ullah et al., 2023). Schick and Schütze (2021) introduced PET (Pattern-Exploiting Training), which converts classification tasks into cloze-style questions. Gao et al. (2021) proposed LM-BFF, demonstrating that prompt-based fine-tuning with demonstrations improves few-shot performance. Liu et al. (2023) provided a comprehensive survey of prompt-based methods. Most prompt-based approaches focus on decoder-only models, such as GPT. Recent work has explored prompting for encoder models. Gu et al. (2022) introduced PPT for masked language models. However, these methods typically use fixed prompt templates. Our ADAPT method extends this by dynamically selecting optimal prompts during training.

Parameter-efficient methods such as LoRA (Hu et al., 2022) and QLoRA (Dettmers et al., 2024) reduce computational costs while maintaining performance. These methods have shown competitive performance on high-resource languages. However, their effectiveness on low-resource languages like Urdu in few-shot settings remains underexplored. Our work provides systematic evaluation of these methods for Urdu. Few-shot learning aims to achieve better performance with minimal labeled data. Brown et al. (2020) demonstrated that large language models can perform tasks with just a few examples through in-context learning. Perez et al. (2021) showed that prompt-based methods are particularly effective in few-shot scenarios.

For low-resource languages, few-shot learning is critical. Ponti et al. (2020) studied cross-lingual transfer in few-shot settings. Winata et al. (2021) explored multilingual few-shot learning. Our work contributes to this area by systematically comparing different fine-tuning strategies in extreme few-shot settings for Urdu.

3 Urdu-GLUE Benchmark

In this section, we discuss the Urdu-GLUE benchmark. This benchmark comprises ten diverse tasks across four categories: single-sentence, similarity and paraphrase, inference, and sequence labeling tasks. We created four new datasets to address the resource gaps, and the remaining six tasks utilize publicly available Urdu datasets. Table 1 provides a summary of the newly translated datasets.

3.1 Dataset Descriptions

Urdu Corpus of Linguistic Acceptability (U-CoLA): We created U-CoLA by manually translating the English CoLA dataset (Warstadt et al., 2019). Grammatically correct sentences were translated into natural Urdu, while ungrammatical sentences were translated into intentionally ungrammatical Urdu that preserves the syntactic violation. For example, subject-verb agreement errors in English were reflected in corresponding errors in Urdu, which accounts for Urdu's gender and number agreement system.

U-WNLI (Urdu Winograd Natural Language Inference). We translated the English WNLI (Wang et al., 2018) dataset to create

U-WNLI. Urdu's distinct pronoun forms based on formality, gender, and social distance posed unique challenges. We carefully translated each instance to preserve resolution difficulty while ensuring linguistic naturalness.

U-STS-B (Urdu Semantic Textual Similarity). The U-STS-B dataset created by translating the English STS-B (Wang et al., 2018) dataset. The annotators maintained relative similarity relationships between pairs, ensuring that highly similar pairs in English remained highly similar in Urdu. U-STS-B contains $8,515$ sentence pairs, with continuous similarity scores [0-5].

U-XNLI (Preprocessed Cross-lingual NLI). The XNLI includes Urdu (Conneau et al., 2018), however, the dataset mixed standard Urdu script with Roman Urdu. Roman Urdu lacks standardized orthography, with the same word appearing in multiple spellings (e.g., in Roman Urdu: "acha", "achha", "achaa", in Urdu: اچھا) for "Good" in English. We systematically converted all Roman Urdu instances to standard Perso-Arabic script and validated semantic equivalence and label consistency through manual verification.

3.2 Translation Protocol and Challenges

We employed a rigorous process for creating our datasets. Initially, we utilize IndicTrans2 (Gala et al., 2023) to translate the source datasets. However, the translations exhibited significant issues, including grammatical errors, word-by-word translations (lacking fluency), and incorrect handling of complex syntactic structures. U-CoLA was particularly difficult due to the need to preserve grammatical violations. The translation model struggled with complex grammatical constructions, embedded clauses, and non-canonical word orders. It often distorted complex sentences, either (1) unintentionally correcting ungrammatical inputs, or (2) introducing new errors in acceptable sentences. Tense inconsistencies were common near the past-present boundary, and named entities were frequently mistranslated or transliterated. Sentences with multiple entities posed difficulties in preserving participant roles and argument structure.

U-WNLI presented challenges with longer and structurally complex sentences. The model tended toward literal word-by-word translation without capturing sentence-level semantics. This was detrimental because correct inference depends on accurately resolving pronouns and entity references across clauses. Inferential relationships were frequently lost, requiring substantial correction to restore logical coherence. U-STS-B required preserving fine-grained semantic similarity relationships. Minor translation deviations significantly altered perceived similarity between sentence pairs. The model occasionally failed to maintain nuanced semantic overlap, and tense inconsistencies further affected alignment. U-XNLI was easy, because the mian task was to convert Roman Urdu to standard Perso-Arabic script (Urdu). However it require careful verification, as dense or syntactically complex sentence pairs occasionally exhibited weakened inference cues after conversion. These limitations made the automatically translated instances unsuitable for creating a language understanding benchmark. Therefore, three native Urdu speakers, all fluent in English, independently translated each instance. They verified semantic equivalence, label consistency, and linguistic naturalness by comparing each English sentence with its Urdu translation.

Dataset	Task Type	Instances	Classes
U-CoLA	Acceptability	$10,025$	Binary
U-XNLI	NLI	7500	3-class
U-WNLI	Winograd NLI	859	Binary
U-STS-B	Similarity	$8,515$	Reg.

Table 1: Summary of four newly created datasets for Urdu-GLUE benchmark. Reg. shows regression and the scores ranging from 0 (completely different) to 5 (identical).

4 Methodology

In this section, we discuss our proposed ADAPT technique. We utilize two multilingual encoder-based pretrained language models (mBERT and XLM-R). These models provide effective multilingual transfer capabilities for various languages, including Urdu. We design three experimental settings (zero-shot, 16-shot, and $80/20$ split) to evaluate model performance under varying data availability. For the $80/20$ split, we evaluate three setups, (1) standard fine-tuning, (2) LoRA (Hu et al., 2022), and (3) QLoRA (Dettmers et al., 2024). LoRA introduces trainable low-rank matrices while freezing pre-trained weights, and QLoRA extends this by quantizing the base model. Both methods reduce the number of trainable parameters and memory requirements. We evaluate LoRA and QLoRA in the $80/20$ split (only) to assess parameter-efficient and memory-efficient

Algorithm 1 ADAPT

Require: Training set $\mathcal{D}_{train}$, validation set $\mathcal{D}_{val}$, template set $\mathcal{T} = \{t_1, \ldots, t_{10}\}$, verbalizer $\mathcal{V}$
Ensure: Fine-tuned model θ^*, optimal template t^*

Phase 1: Dynamic Template Training
Initialize θ from pre-trained MLM
for epoch $e = 1$ to E **do**
  for each batch $\mathcal{B} \subset \mathcal{D}_{train}$ **do**
    Sample $t^{(s)} \sim \text{Uniform}(\mathcal{T})$
    Construct prompted inputs using $t^{(s)}$
    Compute loss $\mathcal{L}$ as:
    $\mathcal{L} = -\frac{1}{|\mathcal{B}|} \sum_{(x_i, y_i) \in \mathcal{B}} \log P(y_i \mid x_i, t^{(s)})$
    Update $\theta \leftarrow \theta - \alpha \nabla_\theta \mathcal{L}$
  end for
  Evaluate on $\mathcal{D}_{val}$ for early stopping
end for
Phase 2: Template Evaluation
for each $t_k \in \mathcal{T}$ **do**
  Compute $\mathcal{L}_k$ on $\mathcal{D}_{val}$ using template t_k
end for
Phase 3: Optimal Template Selection
$t^* \leftarrow_{t_k \in \mathcal{T}} \mathcal{L}_k$
return θ^*, t^*

alternatives to standard fine-tuning, which is particularly relevant for low-resource languages. For zero-shot and 16-shot settings, we compare ADAPT against standard FT and prompt-based FT. Standard FT serves as our primary baseline setting. We implement prompt-based FT with a single manually designed template per task. The model is fine-tuned to predict labels through cloze-style prompts rather than standard classification heads. This baseline is used only in zero-shot and 16-shot settings.

4.1 ADAPT: Adaptive Dynamic Prompt Template

We introduce ADAPT, a novel dynamic prompt-based fine-tuning strategy designed for encoder-based models in low-resource settings. Unlike fixed-template methods (Ullah et al., 2025), ADAPT explores multiple candidate templates (detailed in Appendix A.2) during training and automatically identifies the optimal formulation for each task, as shown in the Algorithm 1. In low-resource settings, model performance is susceptible to the formulation of the prompt. A slight variation in prompt design can have a significant impact on performance. Manually searching for the best prompt through trial and error is expensive when labeled data is scarce. ADAPT addresses this by efficiently exploring the prompt space during training. Let the labeled dataset be denoted by

$$\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}, \tag{1}$$

where $y_i \in \mathcal{Y}$ and $|\mathcal{Y}| = C$ is the number of classes. From $\mathcal{D}$, we construct a few-shot training subset $\mathcal{D}_{fs}$ by sampling exactly 16 instances per class:

$$\mathcal{D}_{fs} = \bigcup_{c \in \mathcal{Y}} \{(x_j, c)\}_{j=1}^{16}, \tag{2}$$

resulting in $|\mathcal{D}_{fs}| = 16 \times C$. Using $\mathcal{D}_{fs}$ as input, we manually create a set of candidate templates 10 and verbalizers (2 per class). The template set is:

$$\mathcal{T} = \{t_1, t_2, \ldots, t_{10}\} \tag{3}$$

where each template t_k maps an input example x to a masked sequence suitable for MLM prediction. In parallel, a verbalizer $\mathcal{V}$ is constructed such that each class $c \in \mathcal{Y}$ is associated with exactly two label words (e.g. for the classification task):

$$\mathcal{V}(c) = \{w_{c1}, w_{c2}\} \tag{4}$$

For a specific task, the verbalizers remained fixed across all epochs and templates to ensure that performance differences arose from training dynamics rather than variations in verbalizers. To guarantee balanced supervision, training batches are constructed using a balanced batch sampling strategy. Let the batch size be B, constrained such that:

$$B \equiv 0 \pmod{C} \tag{5}$$

Each batch $\mathcal{B} \subset \mathcal{D}_{fs}$ contains exactly B/C examples from each class, ensuring uniform class representation within every batch and preventing template-specific bias caused by class imbalance. The batch size is further chosen to control template exposure per epoch. Since $|\mathcal{D}_{fs}| = 16 \times C$, we define the batch size as:

$$B = \frac{16C}{4} = 4C \tag{6}$$

which partitions the few-shot dataset into exactly four balanced batches per epoch. Consequently, each epoch consists of four training steps, with each step processing a distinct subset of $\mathcal{D}_{fs}$.

The core mechanism of ADAPT is dynamic template selection during training. Let T denote

the total number of epochs, with $T = 20$, and let K denote the number of training steps before changing the template. In ADAPT, we set $K = 1$, meaning that the template is resampled after every batch. This choice maximizes template diversity and prevents the model from overfitting to specific prompt patterns. With only 16 examples split into 4 batches per epoch, using $K > 1$ would mean some templates barely appear during training. While exploring different values of K could provide additional insights, we consider this analysis to be outside the scope of this work and leave it for future research. At training step s, the active template $t^{(s)}$ is drawn uniformly at random:

$$t^{(s)} \sim \text{Uniform}(\mathcal{T}) \tag{7}$$

Thus, within a single epoch, the model is trained using four distinct (possibly repeating) templates, and across epochs, the ordering and frequency of templates vary stochastically. Uniform template sampling ensures the model sees all templates equally during training, therefore preventing bias toward specific templates. Given an input-label pair (x, y), a template t, and a verbalizer $\mathcal{V}$, the model produces a probability distribution over label words at the masked position. The probability of class y is computed as:

$$P(y \mid x, t) = \sum_{w \in \mathcal{V}(y)} P_\theta([\text{MASK}] = w \mid t(x)) \tag{8}$$

where θ denotes the parameters of the MLM. Training minimizes the cross-entropy loss over a batch $\mathcal{B}$:

$$\mathcal{L}(\theta) = -\frac{1}{|\mathcal{B}|} \sum_{(x_i, y_i) \in \mathcal{B}} \log P(y_i \mid x_i, t^{(s)}) \tag{9}$$

Importantly, all templates share the same model parameters θ, and gradient updates are accumulated across batches instantiated with different templates. This enforces learning of representations that are robust to template variation rather than specialized to a single formulation. By repeatedly exposing the model to different templates at every training step, ADAPT effectively optimizes the expected loss over the template distribution:

$$\mathbb{E}_{t \sim \mathcal{T}}[\mathcal{L}(\theta; t)] \tag{10}$$

Instead of minimizing the loss for a single fixed template. This training objective reduces template-induced variance, ensuring that no individual template becomes a limiting factor in learning. The resulting model experiment involves using different templates and selecting the one with the maximum loss reduction. This allows the model to learn from diverse prompt formulations and prevents overfitting to a single template structure. After training, we evaluate each template's performance on the validation set. For each template, we compute validation loss and task-specific metrics. We select the template that achieves the lowest validation loss as the optimal template for inference.

For instance, in classification tasks, we convert the problem into a masked language modeling objective. Given an input x and template t, we construct a prompted input $t(x)$ with a `[MASK]` token at the label position. The model predicts the probability distribution over label words:

$$P(y|x) = P(\text{label\_word}(y)|\texttt{[MASK]} \text{ in } t(x))$$

where label_word(y) maps each class label to a corresponding word (e.g., "positive" -> "good"). We train all models for 20 epochs with early stopping based on validation loss. We use the AdamW optimizer with a learning rate of $2e-5$. All results are calculated using a fixed seed of 42.

5 Results and Discussion

In this section, we present our experimental results across three data setups. The primary goal is to build effective NLP systems for Urdu when labeled data is limited.

5.1 Standard FT vs. LoRA vs. QLoRA

Table 2 presents the results obtained when the model is trained on 80% (a substantial portion) of the data. Standard FT generally achieves better performance. For mBERT, we see better results on tasks like U-CoLA (93.9%), SST-M (91.0%), and sequence labeling (93.9% POS, 84.7% NER). These results demonstrate that when data is available, standard FT methods work well for Urdu. However, parameter-efficient methods LoRA and QLoRA with mBERT struggle on certain tasks, i.e., the performance drop on U-CoLA from 93.9% to around 68% and 67%, respectively. Sequence labeling tasks also suffer, with POS accuracy falling from 93.9% to just 44.8% with LoRA and 52% with QLoRA. Surprisingly, these methods perform better on inference tasks like U-XNLI (LoRA 61.7% and QLoRA 69.4%) and BoolQ (LoRA 69.4% and QLoRA 71.1%), and surpass

Method	Single-Sentence			Sim. & Para.		Inference			Seq. Label.	
	UCoLA	SST-2	SST-M	Par.	USTS-B	UXNLI	Bl-Q	UWNLI	POS	NER
bert-base-multilingual-cased (mBERT)										
Stand. FT	**93.9**	**67.8**	**91.0**	78.6	**85.7**	54.8	32.4	50.8	**93.9**	**84.7**
LoRA	68.5	54.2	76.8	78.5	68.0	61.7	69.4	52.1	44.8	56.2
QLoRA	67.8	61.4	77.1	**79.5**	71.0	**69.4**	**71.1**	**54.6**	52.3	61.9
xlm-roberta-base (XLM-R)										
Stand. FT	70.5	**74.5**	**91.7**	**89.3**	**89.1**	45.6	54.4	50.8	**95.6**	**88.5**
LoRA	**74.2**	65.5	83.1	85.4	83.0	**76.5**	**78.2**	**58.3**	56.4	67.4
QLoRA	73.9	65.2	82.8	85.1	79.0	76.2	77.9	58.1	55.9	66.8

Table 2: Performance comparison of Standard fine-tuning (Stand. FT), LoRA, and QLoRA on $80/20$ split across Urdu-GLUE tasks. Results are reported as accuracy (%) except for U-STS-B (Pearson correlation). (Par: Paraphrase detection, Bl-Q: Bool Questions)

standard fine-tuning on both tasks with scores of 54.8% and 32.4%, respectively.

The results change with the XLM-R model, where the LoRA and QLoRA remain much better than standard fine-tuning. While they still trail on tasks like SST-M and sequence labeling, the gaps are smaller. These methods are better on inference tasks, achieving 76.5% on U-XNLI compared to standard fine-tuning 45.6%. This suggests that XLM-R representations may be more amenable to parameter-efficient adaptation, at least for certain task types in the Urdu language. It is worth mentioning that we strategically evaluate each method in settings where it is most applicable, e.g., prompt-based methods for extremely low-resource scenarios (zero-shot, 16-shot) and parameter-efficient methods for resource-constrained fine-tuning with sufficient data ($80/20$ split). Standard fine-tuning provides a consistent baseline across all settings (zero-shot, 16-shot, and $80/20$ split).

5.2 Performance with Zero-Shot and 16-Shots

Without any task-specific training, PLMs performance varies dramatically across different tasks. XLM-R shows better zero-shot transfer than mBERT, particularly on POS tagging, where it achieves 85.3% compared to mBERT 15.0% as shown in Table 3. This gap highlights the importance of pre-training data and model architecture for cross-lingual transfer. In comparison to standard fine-tuning, prompt-based methods in zero-shot settings are not as effective. This is obvious because without task-specific training, the model hasn't learned to follow prompt patterns effectively. The main question is: "What happens when we have just a handful of examples?" The 16-shot results reveal some interesting findings in response to the mentioned question. With just 16 examples per class, ADAPT consistently delivers substantial improvements over both standard FT and fixed-template prompting. For instance, on the paraphrase detection task with XLM-R, ADAPT achieves 86.1% accuracy compared to standard fine-tuning 25.0%. For mBERT on the same task, the gain is similarly dramatic, 72.2% (ADAPT) versus 42.86% (standard FT).

It is important to mention the inherent limitation of MLM, requiring discrete label words. For U-STS-B task, we discretize continuous scores into five bins: [0-1): "unrelated", [1-2): "distant", [2-3): "similar", [3-4): "closely", [4-5]: "identical". For evaluation, we convert predicted bins to continuous scores using bin midpoints and compute Pearson correlation for consistency with GLUE benchmarks. We acknowledge this discretization loses fine-grained supervision and may not be directly comparable to regression-based methods. However, ADAPT achieves 49.4% and standard FT achieves 30.27% on mBERT. These results suggest that dynamic template selection helps the model better capture semantic relationships when limited data is available. ADAPT also improves performance on several other tasks, e.g. for Winograd-style inference (U-WNLI), we see consistent gains as 59.2% (ADAPT) versus 54.93% (standard FT) with mBERT, and 52.1% (ADAPT) versus 45.07% (standard FT) with XLM-R. On U-CoLA, XLM-R with ADAPT reaches 74.1% compared to 65.29% with standard fine-tuning. Sequence labeling presents an interesting results. With XLM-R, ADAPT achieves 89.9% on POS

Method	Single-Sentence			Sim. & Para.		Inference			Seq. Label.	
	UCoLA	**SST-2**	**SST-M**	**Par.**	**USTS-B**	**UXNLI**	**Bl-Q**	**UWNLI**	**POS**	**NER**
Zero-shot: bert-base-multilingual-cased (mBERT)										
Stand. FT	69.1	53.3	72.2	46.4	09.1	35.1	28.5	60.6	15.0	14.0
Prompt-FT	46.69	47.96	30.28	46.43	18.67	33.98	24.60	46.48	11.72	24.57
Zero-shot: xlm-roberta-base (XLM-R)										
Stand. FT	71.2	55.8	75.5	53.6	37.3	38.1	29.8	56.3	85.3	53.4
Prompt-FT	68.65	65.65	56.25	46.43	19.60	33.49	26.11	53.52	18.82	28.22
16-shot: bert-base-multilingual-cased (mBERT)										
Stand. FT	53.60	52.04	**67.04**	42.86	30.27	34.78	27.32	54.93	46.88	**33.48**
Prompt-FT	**62.61**	50.34	63.73	21.43	35.40	35.18	27.22	50.70	**57.64**	16.63
ADAPT	59.2	**53.3**	41.7	**72.2**	**49.4**	**37.9**	**31.3**	**59.2**	41.6	20.0
16-shot: xlm-roberta-base (XLM-R)										
Stand. FT	65.29	73.13	**84.60**	25.00	27.27	35.34	**30.54**	45.07	52.42	27.25
Prompt-FT	61.65	**77.55**	83.16	50.00	**33.07**	**38.15**	26.41	50.70	62.67	**29.29**
ADAPT	**74.1**	50.3	64.6	**86.1**	19.6	35.5	15.4	**52.1**	**89.9**	21.1

Table 3: Performance comparison on zero-shot and 16-shot settings. ADAPT refers to our proposed Adaptive Dynamic Prompt Template method. Bold indicates best performance per model and setting. Results are reported as accuracy (%) except for U-STS-B (Pearson correlation). Par: Paraphrase detection, Bl-Q: Bool Questions)

tagging, better than standard fine-tuning's 52.42%. This suggests that even for structured prediction tasks, dynamic prompting can be highly effective in few-shot scenarios.

ADAPT performs better on semantically complex tasks like paraphrase detection and textual similarity, where prompt formulation matters most. Both mBERT and XLM-R benefit from ADAPT in 16-shot settings, indicating robustness across models. Our results reveals that (1) with abundant data ($80/20$ split), standard fine-tuning works best, (2) with limited data (16-shot), ADAPT provides substantial improvements. Comparing with fixed-template prompt-based fine-tuning, we noticed that ADAPT consistently outperforms. These findings extend beyond Urdu to other (relevant) low-resource languages, suggesting that the approach is robust across various tasks. ADAPT appears most valuable for semantically complex tasks where prompt formulation significantly affects performance. For simpler tasks or when more data becomes available, traditional fine-tuning may be more practical.

6 Conclusion

We introduced Urdu-GLUE, the first comprehensive benchmark for Urdu language understanding. Urdu-GLUE comprises ten diverse tasks across single-sentence classification, similarity and paraphrase detection, inference, and sequence labeling. We created four new datasets, (1) U-CoLA, (2) U-WNLI, (3) U-STS-B, and U-XNLI (preprocessed). To address the challenge of limited labeled data in low-resource languages (specifically for Urdu language), we proposed ADAPT, a novel dynamic prompt-based fine-tuning strategy for encoder models. ADAPT systematically explores multiple candidate templates during training and automatically identifies the optimal formulation for each task.

Our comprehensive experiments across three data regimes reveal clear patterns. In the extreme few-shot setting (16 examples per class), ADAPT consistently outperforms standard fine-tuning, achieving up to 61% percentage points of improvement on semantic tasks. These gains demonstrate that thoughtful prompt engineering combined with dynamic template selection can dramatically improve model performance when labeled data is scarce. Our findings have practical implications for low-resource language understanding. Prompt-based methods should be prioritized when working with limited data, particularly for semantically complex tasks. We publicly release all datasets and code to facilitate future research and establish Urdu-GLUE as a benchmark for the Urdu NLP community.

Limitations

Our work has several limitations. Four of ten tasks/datasets are translated from English, which may introduce artifacts despite careful manual verification by native speakers. While we maintained translation quality and aimed for natural Urdu phrasing, some syntactic patterns may still reflect English structures. Cultural nuances posed some challenges. WNLI pronoun resolution cannot fully reflect Urdu's formality distinctions e.g. (آپ, تم, تو) all translate to **"you"** in English. We preserved such distinctions where possible, but complete cultural adaptation remains difficult when translating standardized benchmarks. For U-STS-B, we discretize continuous scores into bins for prompt-based methods, which loses fine-grained supervision. While ADAPT shows strong results for Urdu, validation on other low-resource languages would strengthen generalizability claims.

Ethical considerations

All datasets utilized in this research work are either manually translated from public English sources/datasets or existing Urdu datasets, no private data was collected. By creating resources for Urdu (246 M speakers), we aim to bridge the gap between high-resource and low-resource languages. We release all resources publicly to ensure equitable access for researchers, particularly in Urdu-speaking regions.

Acknowledgments

We thank the three native Urdu speakers for their careful translation and verification. We also thank the anonymous reviewers for their valuable feedback.

References

Farah Adeeba, Brian Dillon, Hassan Sajjad, and Rajesh Bhatt. 2025. Urblimp: A benchmark for evaluating the linguistic competence of large language models in urdu. *arXiv preprint arXiv:2508.01006*.

Samee Arif, Abdul Hameed Azeemi, Agha Ali Raza, and Awais Athar. 2024. Generalists vs. specialists: Evaluating large language models for Urdu. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 7263–7280, Miami, Florida, USA. Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. XNLI: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2475–2485, Brussels, Belgium. Association for Computational Linguistics.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2024. QLoRA: Efficient finetuning of quantized LLMs. *Advances in Neural Information Processing Systems*, 36.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Jay Gala, Pranjal A. Chitale, Raghavan AK, Varun Gumma, Sumanth Doddapaneni, Aswanth Kumar, Janki Nawale, Anupama Sujatha, Ratish Puduppully, Vivek Raghavan, Pratyush Kumar, Mitesh M. Khapra, Raj Dabre, and Anoop Kunchukuttan. 2023. Indictrans2: Towards high-quality and accessible machine translation models for all 22 scheduled indian languages. *Preprint*, arXiv:2305.16307.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2021. Making pre-trained language models better few-shot learners. 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 3816–3830, Online. Association for Computational Linguistics.

Yuxian Gu, Xu Han, Zhiyuan Liu, and Minlie Huang. 2022. PPT: Pre-trained prompt tuning for few-shot learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8410–8423, Dublin, Ireland. Association for Computational Linguistics.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *Proceedings of the 10th International Conference on Learning Representations (ICLR)*.

Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, Gokul N.C., Avik Bhattacharyya, Mitesh M. Khapra, and Pratyush Kumar. 2020. IndicNLPSuite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for Indian languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4948–4961, Online. Association for Computational Linguistics.

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 Computing Surveys*, 55(9):1–35.

Sungjoon Park, Jihyung Moon, Sungdong Kim, Won Ik Cho, Ji Yoon Han, Jangwon Park, Chisung Song, Junseong Kim, Youngsook Song, Taehwan Oh, Joohong Lee, Juhyun Oh, Sungwon Lyu, Younghoon Jeong, Inkwon Lee, Sangwoo Seo, Dongjun Lee, Hyunwoo Kim, Myeonghwa Lee, and 14 others. 2021. Klue: Korean language understanding evaluation. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1.

Ethan Perez, Douwe Kiela, and Kyunghyun Cho. 2021. True few-shot learning with language models. In *Advances in Neural Information Processing Systems*, volume 34, pages 11054–11070. Curran Associates, Inc.

Jan Pfister and Andreas Hotho. 2024. SuperGLEBer: German language understanding evaluation benchmark. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7904–7923, Mexico City, Mexico. Association for Computational Linguistics.

Edoardo Maria Ponti, Goran Glavaš, Olga Majewska, Qianchu Liu, Ivan Vulić, and Anna Korhonen. 2020. XCOPA: A multilingual dataset for causal commonsense reasoning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2362–2376, Online. Association for Computational Linguistics.

Timo Schick and Hinrich Schütze. 2021. 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.

Tatiana Shavrina, Alena Fenogenova, Emelyanov Anton, Denis Shevelev, Ekaterina Artemova, Valentin Malykh, Vladislav Mikhailov, Maria Tikhonova, Andrey Chertok, and Andrey Evlampiev. 2020. RussianSuperGLUE: A Russian language understanding evaluation benchmark. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4717–4726, Online. Association for Computational Linguistics.

Munief Hassan Tahir, Sana Shams, Layba Fiaz, Farah Adeeba, and Sarmad Hussain. 2025. Benchmarking the performance of pre-trained llms across urdu nlp tasks. In *Proceedings of the First Workshop on Challenges in Processing South Asian Languages (CHiPSAL 2025)*, pages 17–34.

Faizad Ullah, Ubaid Azam, Ali Faheem, Faisal Kamiran, and Asim Karim. 2023. Comparing prompt-based and standard fine-tuning for Urdu text classification. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 6747–6754, Singapore. Association for Computational Linguistics.

Faizad Ullah, Safiullah Faizullah, Imdad Ullah Khan, Turki Alghamdi, Toqeer Ali Syed, Ahmad B Alkhodre, Muhammad Sohaib Ayub, and Asim Karim. 2025. Prompt-based fine-tuning with multilingual transformers for language-independent sentiment analysis. *Scientific Reports*, 15(1):20834.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019. Superglue: A stickier benchmark for general-purpose language understanding systems. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. 2019. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641.

Genta Indra Winata, Andrea Madotto, Zhaojiang Lin, Rosanne Liu, Jason Yosinski, and Pascale Fung. 2021. Language models are few-shot multilingual learners. In *Proceedings of the 1st Workshop on Multilingual Representation Learning*, pages 1–15, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Liang Xu, Hai Hu, Xuanwei Zhang, Lu Li, Chenjie Cao, Yudong Li, Yechen Xu, Kai Sun, Dian Yu, Cong Yu, Yin Tian, Qianqian Dong, Weitang Liu, Bo Shi, Yiming Cui, Junyi Li, Jun Zeng, Rongzhao Wang, Weijian Xie, and 13 others. 2020. CLUE: A Chinese language understanding evaluation benchmark. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4762–4772, Barcelona, Spain (Online). International Committee on Computational Linguistics.

A Appendix

A.1 Overview of Urdu-GLUE Tasks

Table 4 provides an overview of the Urdu-GLUE benchmark. For each task, we specify the corresponding English dataset form the Wang et al. (2018) paper. Single sentence tasks includes three datasets, U-CoLA for grammatical acceptability, binary sentiment analysis, and multi-class (3 classes) sentiment analysis. The next category is similarity and paraphrase detection. This task comprises UPPC for paraphrase detection and U-STS-B for semantic similarity. The U-STS-B employs a regression-style approach with similarity scores ranging from 0 (differnet) to 5 (similar). Natural language inference is evaluated through three diverse tasks including U-XNLI, OpenBookQA Urdu (Bool Questions), and U-WNLI for Winograd-style NLI. Urdu-GLUE benchmark includes two structured prediction tasks, i.e., Urdu POS tagging named entity recognition.

A.2 ADAPT Templates

We provide all the templates used for the ADAPT methodology across all datasets. For each dataset, we manually create 10 diverse templates in Urdu to facilitate dynamic prompt-based fine-tuning (ADAPT). Each template includes a `[MASK]` token position where the label/verbalizer is predicted during the ADAPT training process. English translations are provided alongside the Urdu templates for reference and reproducibility.

No.	Task Category	English	Urdu Dataset	Classes
	Single-Sentence Tasks			
1	Grammatical Acceptability	CoLA	U-CoLA	Binary
2	Binary Sentiment	SST-2	Urdu Sentiment	Binary
3	Multi-class Sentiment	SST-M	Urdu Multi-Domain	3-class
	Similarity and Paraphrase Tasks			
4	Paraphrase Detection	MRPC, QQP	UPPC	Binary
5	Semantic Similarity	STS-B	U-STS-B	0–5
	Inference Tasks			
6	Natural Language Inference	MNLI, RTE	U-XNLI	3-class
7	Question Answering	QNLI	OpenBookQA Urdu	Binary
8	Winograd NLI	WNLI	U-WNLI	Binary
	Sequence Labeling Tasks			
9	POS Tagging	Penn Treebank	Urdu POS	34 tags
10	Named Entity Recognition	CoNLL-2003	UNER	8 entities

Table 4: Overview of Urdu-GLUE benchmark showing task categories, source datasets, and statistics.

Templates (Urdu)	English Translation
یہ جملہ [MASK] ہے۔	This sentence is [MASK].
دی گئی عبارت کا مواد اور پیغام [MASK] ہے۔	The given text, its content and message is [MASK].
اس عبارت کا تجزیہ کریں: اس کا احساس یا موڈ [MASK] ہے۔	Analyze this text: its feeling or mood is [MASK].
جملہ: ہمیں بتاتا ہے کہ یہ مواد [MASK] ہے۔	Sentence: tells us that this content is [MASK].
اس عبارت کا مطلب [MASK] ہے۔	This text's meaning is [MASK].
اس معاملے میں حتمی رائے [MASK]	In this matter the final opinion is [MASK]
اس مواد کی تشریح [MASK]	The interpretation of this content is [MASK]
اس حوالے سے فیصلہ [MASK]	The decision in this regard is [MASK]
اس متن کی درجہ بندی [MASK]	The classification of this text is [MASK]
اس اظہار کا نتیجہ [MASK]	The result of this expression is [MASK]

Table 5: COLA, SST-2, and SST-M Dataset Templates

Templates (Urdu)	English Translation
اور کیا ایک جیسے ہیں؟ [MASK]	And are they the same? [MASK]
اور معنی کی مطابقت [MASK] ہے۔	The semantic similarity is [MASK].
کیا یہ دونوں جملے اور ایک دوسرے کی تکرار ہیں؟ [MASK]	Are these two sentences repetitions of each other? [MASK]
کے مطابق [MASK] ہے۔	According to, is [MASK].
دونوں جملے کا تعلق [MASK] ہے۔	Both sentences have a relation that is [MASK].
اور ایک جیسے خیال رکھتے ہیں؟ [MASK]	And do they hold the same idea? [MASK]
اور کے الفاظ کی ہم آہنگی [MASK] ہے۔	The harmony of words is [MASK].
اگر ہم اور کو دیکھیں تو یہ [MASK] ہیں۔	If we look at them then they are [MASK].
کی روشنی میں [MASK] سمجھا جا سکتا ہے۔	In light of, can be understood as [MASK].
اور کے بیچ مفہوم کا تعلق [MASK] ہے۔	The relationship of meaning between is [MASK].

Table 6: Paraphrase Dataset Templates

Templates (Urdu)	English Translation
اور کا تعلق [MASK] ہے۔	The relationship is [MASK].
پہلا جملہ: دوسرا جملہ: ان دونوں کے درمیان تعلق [MASK] ہے۔	First sentence: Second sentence: the relationship between both is [MASK].
کے مقابلے میں کا مفہوم [MASK] ہے۔	Compared to, the meaning is [MASK].
اگر درست ہے تو [MASK] ہے۔	If it is correct then is [MASK].
بیان کے لحاظ سے [MASK] ہے۔	The statement in terms of is [MASK].
کے درمیان رشتہ [MASK] قرار پاتا ہے۔	The relationship is determined to be [MASK].
پہلے جملے کی بنیاد پر دوسرا جملہ [MASK] ہے۔	Based on the first sentence, the second sentence is [MASK].
کے تناظر میں کا تعلق [MASK] بنتا ہے۔	In the context of, the relationship becomes [MASK].
کے بعد کا نتیجہ [MASK] سمجھا جاتا ہے۔	After, the result is understood to be [MASK].
اور کے درمیان منطقی تعلق [MASK] ہے۔	The logical relationship is [MASK].

Table 7: XNLI Dataset Templates

Templates (Urdu)	English Translation
سوال: جواب: یہ جواب [MASK] ہے۔	Question: Answer: This answer is [MASK].
کے لیے جواب صحیح طور پر [MASK] ہے۔	For, the answer is correctly [MASK].
مندرجہ ذیل سوال: انتخاب: نتیجہ [MASK] ہونا چاہیے۔	The following question: Choice: the result should be [MASK].
سوال: اور آپشن: درست جواب [MASK] ہے۔	Question: and option: the correct answer is [MASK].
کے سوال کے لیے انتخاب جواب [MASK] ہے۔	For the question, the choice as answer is [MASK].
سوال: جواب: یہ جواب ہے [MASK]	Question: Answer: This answer is [MASK]
کے لیے آپشن: صحیح جواب [MASK] ہے۔	For, option: the correct answer is [MASK].
سوال: انتخاب: نتیجہ [MASK]	Question: Choice: result [MASK]
کے سوال کا جواب نتیجہ [MASK] ہے۔	The answer to the question, result is [MASK].
اور انتخاب: صحیح جواب [MASK]	And choice: correct answer [MASK]

Table 8: BOOLQ Dataset Templates

Templates (Urdu)	English Translation
کا تعلق [MASK] ہے۔	The relationship is [MASK].
پہلا بیان: دوسرا بیان: ان کا تعلق [MASK] ہے۔	First statement: Second statement: their relationship is [MASK].
کیا سے [MASK] ہوتا ہے؟	Does it become [MASK] from?
کی روشنی میں [MASK] ہے۔	In light of, is [MASK].
کا بیان کے مطابق [MASK] ہے۔	The statement according to is [MASK].
کے درمیان منطقی رشتہ [MASK] ہے۔	The logical relationship is [MASK].
سے، [MASK] طور پر جڑا ہے۔	Is connected as [MASK] from.
اگر دیکھیں تو [MASK] بنتا ہے۔	If we look at, then becomes [MASK].
کے حوالے سے [MASK] سمجھا جاتا ہے۔	In reference to, is understood as [MASK].
اور میں تعلق کی نوعیت [MASK] ہے۔	The nature of the relationship is [MASK].

Table 9: WNLI Dataset Templates

Templates (Urdu)	English Translation
لفظ: حصہ کلام [MASK]:	Word: Part of speech: [MASK]
جملے میں یہ لفظ کس قسم کا ہے: [MASK]	In the sentence, this word is of what type: [MASK].
[MASK] لفظ کا POS tag کیا ہے؟	What is the POS tag of the word? [MASK]
یہ لفظ جملے میں کس زمرے کا ہے؟ [MASK]	This word belongs to which category in the sentence? [MASK]
حصہ کلام: [MASK]	Part of speech: [MASK]
لفظ: اس کا حصہ کلام [MASK] ہے۔	Word: its part of speech is [MASK].
[MASK] لفظ کا ہے POS tag	POS tag for is [MASK]
کس قسم کا لفظ ہے [MASK]	Is what type of word: [MASK]
لفظ → POS: [MASK]	Word → POS: [MASK]
[MASK] کس لفظ جملے میں POS category سے تعلق رکھتا ہے؟	In the sentence, the word belongs to which POS category? [MASK]

Table 10: POS Dataset Templates

Templates (Urdu)	English Translation
جملہ: لفظ ایک [MASK] ہے۔	Sentence: the word is a [MASK].
میں موجود لفظ کی قسم [MASK] ہے۔	The type of word present is [MASK].
مندرجہ ذیل جملے میں: ایک [MASK]: کے طور پر آتا ہے	In the following sentence: comes as a [MASK]:
جملے: لفظ کا درجہ [MASK] ہے۔	Sentence: the category of word is [MASK].
لفظ کی شناخت [MASK]	→ identification of word: [MASK]
جملہ میں ایک [MASK] قسم کا لفظ ہے	In the sentence is a [MASK] type of word:
یہ لفظ جملے میں [MASK] ہے۔	This word in the sentence is [MASK].
جملے میں لفظ کی قسم کیا ہے؟ [MASK]	In the sentence, what is the type of word? [MASK]
جملہ: لفظ کی درجہ بندی کریں [MASK]	Sentence: classify the word: [MASK]
ایک [MASK] کی مثال ہے۔	Is an example of [MASK].

Table 11: NER Dataset Templates

Templates (Urdu)	English Translation
آپس میں [MASK] ہیں۔	They are [MASK] with each other.
کے معنی [MASK] ہیں۔	Their meanings are [MASK].
سے تعلق [MASK] ہے۔	The relationship is [MASK].
معنی کے لحاظ سے [MASK] ہیں۔	They are [MASK] in terms of meaning.
ایک دوسرے سے [MASK] ہیں۔	They are [MASK] with each other.
سے ملایا جائے تو نتیجہ [MASK] ہے۔	When compared, the result is [MASK].
کا مفہوم [MASK] بنتا ہے۔	The resulting meaning is [MASK].
کے مقابلے میں [MASK] ہے۔	“It is [MASK] in comparison.”
میں معنی کا رشتہ [MASK] ہے۔	There is a [MASK] semantic relationship.
ایک جیسے ہونے کے لحاظ سے [MASK] ہیں۔	They are [MASK] in terms of similarity.

Table 12: U-STS-B Dataset Templates