# SSP-CLT: Self-Supervised Prompting for Cross-Lingual Transfer to Low-Resource Languages using Large Language Models

**Anonymous ACL submission**

## Abstract

In-Context Learning (ICL) is a widely embraced paradigm for eliciting task-specific capabilities from large language models (LLMs). Present-day LLMs with ICL have shown exceptional performance on several English NLP tasks, but their utility on other languages is still underexplored. Our work investigates their effectiveness for NLP tasks in low-resource languages (LRLs), especially for cross-lingual transfer, where task-specific training data for one or more related languages is available.

We propose Self-Supervised Prompting for Cross-Lingual Transfer (SSP-CLT), a novel approach for zero-shot cross-lingual transfer to LRLs. SSP-CLT works in two stages and has 2 variants. In first variant, in Stage I, for a given target test instance, exemplars are retrieved from source training data and included in the LLM prompt for ICL – this obtains an initial labeling. Once all test data instances are labeled, Stage II repeats the whole process, but draws exemplars from Stage I labelings of other test datapoints in the target language. The second variant of SSP-CLT uses a fine-tuned model for stage 1 predictions, while stage 2 uses an Integer Linear Programming (ILP)-based exemplar selection that balances similarity, confidence and label coverage. Experiments on 3 tasks and 3 language families demonstrate that SSP-CLT strongly outperforms supervised baselines and also other prompting approaches.

## 1 Introduction

Recent *Large Language Models* (LLMs) such as GPT-3.5-Turbo & GPT-4 (Ouyang et al., 2022; Achiam et al., 2023) show exceptional performance on a variety of NLP and reasoning tasks via *In-Context Learning* (ICL) (Brown et al., 2020; Chowdhery et al., 2022). ICL feeds a task-specific instruction along with few exemplars, appended with the test input, to the LLM. As LLMs can be highly sensitive to selection and ordering of exemplars (Lu et al., 2022; Zhao et al., 2021), exemplar retrieval is a crucial component of ICL.

LLMs show excellent performance on English tasks, but their utility on other languages is relatively underexplored. In particular, we study *zero-shot cross-lingual transfer* to low-resource languages (LRLs) – a setting where labeled task data from one or more related languages is available, but no training data exists for the target LRL.

Cross-lingual transfer has been addressed through standard fine-tuning (Muller et al., 2021; Alabi et al., 2022), and language adapters (Pfeiffer et al., 2020; Üstün et al., 2020; Rathore et al., 2023), but there is limited work on cross-lingual ICL. There are two exceptions (Ahuja et al., 2023; Asai et al., 2023), where ICL is employed with exemplars from a source language, but they use random sampling for exemplar selection, resulting in performance inferior to cross-lingually fine-tuned models, such as mBERT and XLM-R (Devlin et al., 2019; Conneau et al., 2020).

In response, we present Cross-Lingual Self-Adaptive Prompting (SSP-CLT) – a two stage method for cross-lingual transfer to LRLs. In Stage I, SSP-CLT dynamically retrieves exemplars from source language(s) training data, based on the test sentence. The LLM labels the test input based on ICL over the retrieved exemplars in the prompt. In this fashion, all test data points get preliminary labels. In Stage II, SSP-CLT-SIM repeats the same process, but this time, the exemplars are retrieved from the test set itself using similarity as a metric, and are presented to LLM with their Stage I labels. The hypothesis is that an LLM can benefit further from similar sentences in the same language, even if the labels are not entirely accurate.

Noting that SSP-CLT-SIM labels each test instance via an LLM twice, we replace LLM-based Stage I with existing (non-ICL) approaches for cross-lingual transfer. In our work, we use ZGUL (Rathore et al., 2023), which uses language adapters in mBERT, to make preliminary predictions for a test sentence. These labelings are directly to be used in Stage II, i.e., LLM prompt uses labels from ZGUL. This cuts down expensive LLM calls by half. Finally, to select the best exemplars, we develop a novel Integer Linear Programming (ILP) based approach, called SSP-CLT-ILP, which balances the various objectives of (1) similarity with test sentence, (2) confidence in predictions, and (3) coverage of the various labels in the task.

We perform experiments on sequence labeling tasks (POS and NER), and natural language inference (NLI) – a text classification task. Our datasets encompass three typologically diverse language families: African, Germanic and Americas. Our experiments show consistent and substantial improvements over existing supervised as well as simpler ICL-based approaches. We will make both our codebase and prompts publicly accessible.

Our contributions are summarized as follows:

1. We investigate In-Context Learning (ICL) strategies for the task of zero-shot cross-lingual transfer to low-resource languages, utilizing the labeled data from related languages.
2. We propose SSP-CLT, a two-stage self-adaptive prompting paradigm for this task, where first stage may be done by an LLM or other cross-lingual transfer models.
3. We introduce a framework for exemplar selection utilizing an ILP. The ILP incorporates similarity to test input along with confidence of prediction (when available), and enforces label coverage constraints for better selection.
4. Our results show improved F1 scores across 3 tasks and 3 language families, as compared to both existing fine-tuning and LLM-based SoTA models.

## 2 Related Work

**Cross-lingual ICL**: In general, the cross-lingual ICL remains systematically unexplored in literature. Previous approaches for cross-lingual ICL rely on the utilization of random input-output pairs for prompt construction (Zhang et al., 2021; Winata et al., 2021; Ahuja et al., 2023; Asai et al., 2023). Recent methods (Agrawal et al., 2022; Tanwar et al., 2023) aim to fill this void by utilizing semantic similarity for cross-lingual retrieval from a high-resource language's labeled data as candidates, given the target LRL's instance as query. This is facilitated with embedding-based multilingual retrievers such as multilingual sentence-transformers (Reimers and Gurevych, 2020). More recently, OpenAI-based embeddings have been used effectively for cross-lingual retrieval (Nambi et al., 2023).

In above works, the prompting is done in a high-resource language, mostly English. This is called *cross-lingual (CL) prompting*. This is in contrast to *in-language (IL) prompting*, in which exemplars are also retrieved from the training data of the target language. In our setting, we assume no availability of labeled training data for target LRLs, making only CL prompting applicable in our scenario. However, we do conduct comparisons with the IL prompting skyline methods to validate our approaches.

**Self-Adaptive Prompting**: (Wan et al., 2023) proposed *Universal Self-Adaptive* (USP) framework, which utilizes an external unlabeled dataset of 100 instances and labels them using LLM in stage 1. It then performs Chain-of-thought (CoT) sampling to estimate the logits using the same LLM and then utilizes the entropy of logits for exemplar selection in stage 2. Their approach is significantly different from ours in that it is (a) costly- requires multiple runs of LLM to estimate logits, (b) has been shown effective for only English tasks, and (c) uses only *Task Description* in stage 1 and doesn't assume any labeled data (while in our setting, labeled data from source languages is assumed) for ICL.[1]

**Task-Specific prompting**: A prompt consists of (1) Task Description: To facilitate the understanding of task, (2) Labeled Input-Output pairs: Written sequentially in order of their relevance to input query, and (3) Input itself, all appended in the order of their mentions.

Recent studies have shown sensitivity of the output to the template/format of input-output pairs written in the prompt (Sclar et al., 2023; Voronov et al., 2024). We follow the best template given in Sclar et al. (2023) for NLI task, while for sequence labeling, we explore various templates on our own and report our results on the best one. We refer to Appendix section B for details and the exact templates used for each of our tasks.

[1]These 3 key differences make our technique not directly comparable with the USP.

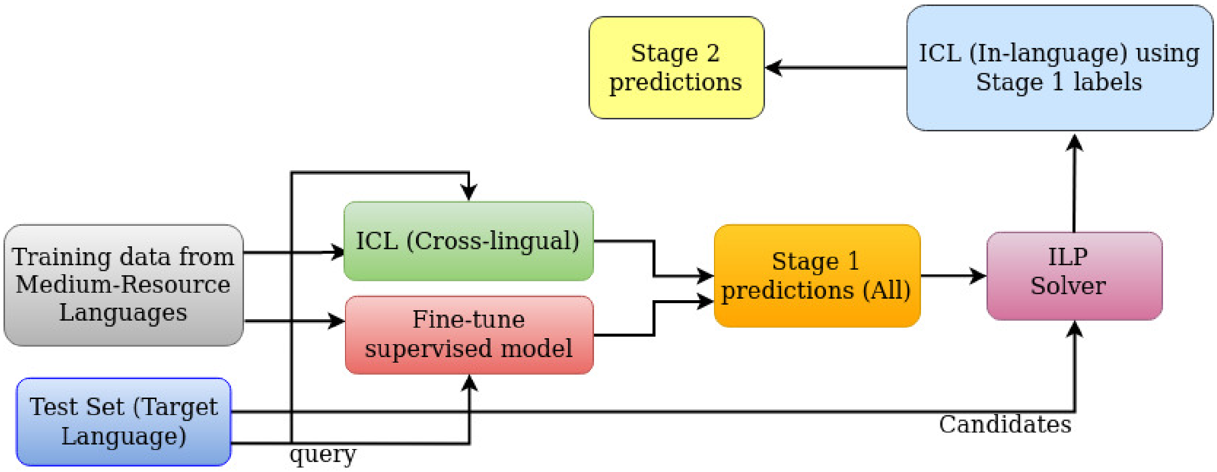


Figure 1: SSP-CLT Paradigm for Cross-Lingual Transfer to target low-resource language

## 3 Methodology

We propose 2 variants of SSP-CLT as explained in subsections 3.1 and 3.2.

### 3.1 SSP-CLT-SIM: *Similarity*-based Self-Supervised Prompting

In this method, LLM is used for obtaining stage 1 predictions, explained as follows:

**Stage 1: Cross-Lingual Transfer using similarity-based retrieval(CLT-SIM)**

The idea is to leverage labeled training data from MRLs and retrieve a set of similar ICL exemplars for each test instance in the target LRL. The retrieval process involved sampling top-K labeled exemplars from the source languages' combined training set based on cosine similarity of *Ada-002* embeddings. The selected exemplars, arranged in descending order of similarity scores, are appended into the prompt between the Task Description (TD) and the input test instance.

**Stage 2: Self-Supervised Prompting using CLT-SIM predictions (SSP-CLT-SIM)**

Since no labeled data for target languages is assumed available, we utilize stage 1 predictions as silver target exemplars to enhance performance. In stage 2, we sub-select a few *in-language* exemplars to be given into the prompt, without assuming any labeled data in the target language. This involves computing Ada-002-based cosine similarity scores between a test instance and the other test instances (excluding itself). Utilizing Stage 1 predictions as silver labels, these exemplars are fed into the prompt.

We name stages 1 and 2 as SSP-CLT and SSP-CLT-SIM, respectively.

### 3.2 SSP-CLT-ILP: Integer Linear Programming (ILP)-based selection framework for Self-Supervised Prompting

To diversify our exemplar selection process, we seek to incorporate other aspects such as *quality* and *diversity*. We hypothesize that the pure similarity-based retrieval is sub-optimal since this doesn't consider the *label* information into account while retrieval. Morover, the exemplars retrieved independently based on purely embedding-based similarity are often redundant as a whole set (Gupta et al., 2023). In response, we introduce 2 additional factors into our selection process as discussed below -

- **Confidence**: We seek to utilize the label confidence elicited from a smaller model, whose logits are accessible unlike OpenAI models. The hypothesis is that confident predictions are also accurate, assuming the model is well-calibrated and can serve as quality exemplars in the prompt.

- **Label Coverage**: We also hypothesize that ensuring coverage of all the labels (available in the label set) in the selected exemplars' set can be more effective in terms of performance.

We formulate the above factors into an ILP with primary and secondary objectives discussed as follows:

$$maximise \sum_{i \in T} y_i * s_i$$

$$s.t. \sum_{i \in T} y_i = M$$

$$\forall i \in T, y_i * (\tau - \hat{y}_i) \leq 0$$

$$\forall j \in LabelSet, \sum_{i \in T} y_i * count(label_j) \geq c_j$$

Where $T$ represents indices of target test samples, $y_i$ denotes binary variable which is 1 if $i^{th}$ sample of $T$ is selected and 0 o.w., $s_i$ denotes similarity score of $i^{th}$ sample with the query, M denotes the no. of exemplars in prompt, $\hat{y}_i$ represents the confidence (probability) of $i^{th}$ instance's prediction using fine-tuned model, $\tau$ is the confidence threshold (a hyperparameter), $label_j$ denotes $j^{th}$ label in the label set and $c_j$ denotes it's corresponding threshold count (another hyperparameter) in the entire set of selected exemplars.
We set $M = 8$, $c_j$=1 ($\forall$ j), $\tau$ = $90^{th}$ percentile prob. value (obtained from fine-tuned model) for a particular label and language, for all our experiments.

## 4 Experiments

### 4.1 Tasks and Datasets

We experiment on 3 tasks - POS tagging, NER and Natural Language Inference (NLI). The chosen language families and datasets are: Universal Dependency (Nivre et al., 2020) for Germanic POS tagging, MasakhaNER (Adelani et al., 2021) for African NER and AmericasNLI (Ebrahimi et al., 2022) for NLI task on Indigenous languages of Americas. We randomly sample 100 test samples for each target language for NER and POS tasks, while 99 test samples (33 for each class - 'entailment', 'contradiction' and 'neutral') for the NLI task. The source (train) and target (test) sets of languages for each task are presented in App. C.

### 4.2 Comparison Models

**Baselines:** We compare our approach with the SOTA supervised models as well LLM-based ICL methods using naive random exemplar selection strategy or the one with no exemplar selection at all. For supervised baselines, we use both publicly available SOTA models (in case applicable) as well as fine-tune our own models on the source languages' data and test zero-shot on the target LRLs.

**Skyline:** For comprehensively evaluating our approach, we utilize the available training data for target languages and perform *few-shot in-language similarity-based* (using Ada-002 embeddings) retrieval for *in-language* prompting to the LLM. This enables analyzing the performance gap due to non-assumption of labeled training data for the target LRLs.

**Ablations:** We perform 3 ablations of SST-CLT-ILP selection strategy - (a) without confidence thresholding, (b) without label coverage and (c) without both i.e. pure similarity-based retrieval. The ablation results have been shown with the best performing underlying LLM i.e. GPT-4x.

**LLMs and fine-tuned models:** We evaluate our method with a series of SOTA open-sourced and closed-sourced LLMs - GPT3.5-turbo (Ouyang et al., 2022), GPT-4x (GPT-4/GPT-4-Turbo) (Achiam et al., 2023), and LLAMA-2-70b (Touvron et al., 2023) for each task. For supervised baselines, we fine-tune ZGUL(Rathore et al., 2023) - mBERT Language Adapter-based SOTA model for NER and POS, mDeBERTa (He et al., 2021) for NLI. We further utilize the publicly available NLI model mDeBERTa-v3-base-xnli-multilingual-nli-2mil7 (Laurer et al., 2022) for NLI evaluation. We term our fine-tuned model as mDeBERTa$^{CL}$ and the puclic model as mDeBERTa$^{100}$, as it was trained on 100 languages. We note that GPT-4 has been used instead of GPT-4-Turbo for the POS task due to the inability of GPT-4-Turbo to follow the instructions and give output that was compatible with the verbalizer used across all the experiments.

## 5 Results and Analysis

We present the results for all tasks in tables 1, 2 and 3. We categorize the results as follows (in the specified order): Fine-tuned (cross-lingual) model, SSP-CLT-SIM approach with LLama-2, GPT3.5, and GPT4x LLMs, SSP-CLT-ILP approach with the same LLMs, ablations of SSP-CLT-ILP with GPT4x LLM, and finally, a skyline approach employing few-shot in-language retrieval (using OpenAI Ada) along with their gold labels in prompt to the GPT4x. We do not embolden skyline results because they are not comparable with our approaches, which do not use target gold labels in anyway. Our observations are as follows:

**SSP-CLT-SIM achieves gains over CLT-SIM**: We observe that The SSP-CLT-SIM method has improved gains over CLT-SIM strategy across all tasks and language families for GPT3.5 and GPT4. For LLaMa-2, the increase in average precision (65.6 to 70.3) was offset by a decrease in average

| Model | Hau | Ibo | Kin | Lug | Luo | Avg. |
|---|---|---|---|---|---|---|
| ZGUL | 52.2 | 56 | 53.7 | 54.5 | 44.4 | 52.2 |
| CLT-SIM (Llama-2-70b) | 64.3 | 61.2 | 59.2 | 60.1 | 47.3 | 58.4 |
| SSP-CLT-SIM (Llama-2-70b) | 57.6 | 62.6 | 56.0 | 57.6 | 43.1 | 55.4 |
| CLT-SIM (GPT-3.5-turbo) | 54.5 | 69.2 | 57.8 | 63.7 | 46.4 | 58.3 |
| SSP-CLT-SIM (GPT-3.5-turbo) | 62.8 | 68.4 | 64.0 | 63.8 | 47.6 | 61.3 |
| CLT-SIM (GPT-4-turbo) | 64.7 | 80.8 | 64.6 | 71.0 | 53.3 | 66.9 |
| SSP-CLT-SIM (GPT-4-turbo) | 67.2 | 79.6 | 63.3 | 74.1 | 54.4 | 67.7 |
| SSP-CLT-ILP (Llama-2-70b) | 68.4 | 58 | 56.1 | 54.7 | 42.3 | 55.9 |
| SSP-CLT-ILP (GPT-3.5) | 61.1 | 68.9 | 62.1 | 67.1 | 51.4 | 62.1 |
| SSP-CLT-ILP (GPT-4-turbo) | **72.5** | 79.8 | **71.4** | **77.4** | **55.1** | **71.2** |
| w/o Conf. thresholding | 71.3 | **81.9** | 69.2 | 74.6 | 52.7 | 69.9 |
| w/o Label Coverage | 71.1 | 79.8 | **71.4** | **77.4** | **55.1** | 71 |
| w/o both (sim-based) | 70.3 | 81.8 | 68 | 74.8 | 51.9 | 69.4 |
| Few-shot in-language (GPT-4-turbo) | 75.5 | 85.9 | 70.7 | 73.6 | 67.2 | 74.6 |

Table 1: African NER: Ablations of SSP-CLT-ILP strategy shown for GPT-4-Turbo

| Model | Fo | Got | Gsw | Avg |
|---|---|---|---|---|
| ZGUL | 77.2 | 21.1 | 65 | 54.4 |
| CLT-SIM (Llama-2-70b) | 79.1 | 36.0 | 71.8 | 62.3 |
| SSP-CLT-SIM (Llama-2-70b) | 78.5 | 37.9 | 73.5 | 63.3 |
| CLT-SIM (GPT-3.5 First Stage) | 81.2 | 37.9 | 72.2 | 63.8 |
| SSP-CLT-SIM (GPT-3.5) | 82.4 | 63.2 | 79.4 | 75.0 |
| CLT-SIM (GPT-4 First Stage) | 81.3 | 66.5 | 82.3 | 76.7 |
| SSP-CLT-SIM (GPT-4) | 81.8 | **73.7** | 85.4 | **80.3** |
| SSP-CLT-ILP (Llama-2-70b) | 81.1 | 27.1 | 73.5 | 60.6 |
| SSP-CLT-ILP (GPT-3.5) | **83.2** | 54.3 | 79.5 | 72.3 |
| SSP-CLT-ILP (GPT-4) | 82.2 | 63.8 | **85.6** | 77.2 |
| w/o Conf. thresholding | 82.8 | 57 | 81.4 | 73.7 |
| w/o Label Coverage | 82.2 | 63.9 | **85.6** | 77.2 |
| w/o both (sim-based) | 82.4 | 55.8 | 82.3 | 73.5 |
| Few-shot in-language (GPT-4) | 93.5 | 80.7 | 89.9 | 88 |

Table 2: Germanic POS: Ablations of SSP-CLT-ILP strategy shown for GPT-4

| Model | Aym | Quy | Nah | Gn | Avg |
|---|---|---|---|---|---|
| mDeBerta$^{100}$ | 40.4 | 45.5 | 43.4 | 43.4 | 43.2 |
| mDeBerta$^{CL}$ | 39.4 | 44.4 | 41.4 | 46.5 | 42.9 |
| CLT-SIM (GPT-4-turbo) | 34.7 | 42.9 | 44.9 | 55.1 | 44.4 |
| SSP-CLT-SIM (GPT-4-turbo) | 37.4 | 53.5 | 45.5 | 62.6 | 49.8 |
| SSP-CLT-ILP (Llama-2-70b) | 30.6 | 37.4 | 34.3 | 34.3 | 34.2 |
| SSP-CLT-ILP (GPT-3.5-turbo) | 42.4 | 48.5 | 41.4 | 47.5 | 45 |
| SSP-CLT-ILP (GPT-4-turbo) | 43.4 | 52.5 | **49.5** | 62.6 | **52** |
| w/o Conf. thresh-holding | **47.5** | 52.5 | 42.4 | 65.7 | 52 |
| w/o Label Coverage | 43.4 | **54.5** | 46.5 | 52.5 | 49.2 |
| w/o both (i.e. sim-based) | 40.4 | 45.5 | 44.4 | **66.7** | 49.3 |
| Few-shot in-language (GPT-4-turbo) | 43.4 | 56.6 | 51.5 | 61.6 | 53.3 |

Table 3: Americas NLI: Ablations of SSP-CLT-ILP strategy shown for GPT-4-Turbo

recall (53.2 to 46.2) in african NER, explaining the decrease in overall F1. We also obtain gains in POS tagging across all languages and models. This demonstrates that CLT-SIM can provide decent silver labels from stage 1, which along with their respective in-target retrieved sentences can serve as effective ICL exemplars for the next stage to the same LLM. More detailed analysis on this follows in sec. 5.1.

**SSP-CLT-ILP approach is effective across the board**: Our SSP-CLT-ILP method consistently outperforms supervised models (ZGUL for NER, POS, and DeBerta for NLI) across all three tasks, achieving up to a 19-point F1 gain in African NER. In NLI, we observe a statistically significant gain of approximately 9 F1 points compared to the Deberta cross-lingual baseline. These gains carry over to both GPT-3.5-Turbo and LLama-2-70b models, highlighting the robustness of our selection algorithm beyond GPT-4x.

Ablation analysis underscores the significance of confidence-thresholding and label coverage constraints, with their impact varying across tasks. Confidence-thresholding proves crucial for sequence-labeling tasks (NER and POS), while label coverage is critical for the NLI task. Detailed analyses of these findings are provided later. Removing both components results in a pure similarity-based retrieval approach, using the fine-tuned model's labels as silver labels for stage 2. This leads to a consistent performance drop across all three tasks (up to 4 points in POS), emphasizing the importance of diversity in exemplars induced by our ILP technique and its positive impact on downstream performance.

**SSP-CLT-ILP v/s SSP-CLT-SIM**: We additionally compare SSP-CLT-ILP and SSP-CLT-SIM, finding that the average F1 performance is superior for the ILP variant in African and Americas language families, but slightly inferior in the Germanic family. This discrepancy arises due to the notably poor performance of the fine-tuned model ZGUL on the specific language Got (Gothic). Consequently, utilizing ZGUL's labels has a disproportionately negative impact on this language. We defer a thorough investigation into whether GPT-4 has encountered this language during pre-training to future research.

**Label coverage is crucial for NLI:** We observe average gains of 2.8 F1 points over AmericasNLI task compared to the ablation that does not ensure label coverage as a constraint. To investigate further, we compute average no. of exemplars for each label that's covered in the prompt for both methods alongwith their label-wise F1 scores (details in Fig. 2). We observe that the 'neutral' label is not sampled in any of the *w/o label coverage* variant, while exactly one 'neutral' label is sampled in the SSP-CLT-ILP (w. label constraint). This so happens as the smaller fine-tuned model i.e. DeBerta-CL has poor recall (22 points) for 'neutral' class and hence the ILP solver has tendency to not sample this label, unless enforced via constraint. The class-wise scores for DeBerta$^{CL}$, SSP-CLT-ILP and SSP-CLT-ILP w/o label coverage are presented in table 4. We observe a difference of 22 recall points for 'neutral' class (57.6 vs 35.6) between the 2 ILP variants. An example depicting this behaviour in terms of the exemplars selected by both methods has been shown in Figure 6.

| Model | Neu. | Ent. | Con. |
|---|---|---|---|
| DeBerta$^{CL}$ | 21.8 | **74.5** | 32.3 |
| SSP-CLT-ILP | **57.6** | 47.7 | 50.72 |
| (w/o Label) | 35.6 | 43.9 | **68.2** |

Table 4: Labelwise Recall for fine-tuned model (DeBerta-based) and ILP variants w. and w/o Label coverage (GPT-4-Turbo)

**Confidence thresholding is helpful for NER and POS:** We observe that, contrary to the observation of label coverage being crucial in NLI, the confidence thresholding plays the key role in sequence labeling tasks NER and POS. This is validated from ablation results in tables 1 and 2, wherein removing confidence thresholding constraint from ILP leads to 3.5 points drop for POS tagging (Germanic) and 1.3 points for NER. The drop is particularly significant (around 7 F1 points) for Gothic (Got), showing that not utilizing the confidence scores of ZGUL leads to drastic drop. This is despite the fact that the performance of ZGUL was already poor on Got (21 F1 points), even then utilizing it's confidence scores leads to huge improvements. More insights into this follows in the next point.

**SSP-CLT-ILP effectively samples high-precision exemplars:** We investigate the precision of the exemplars being selected by SSP-CLT-ILP as well all it's ablation variants. We compute the label-wise precision of all M×N (M=8, N=no. of test instances) for each target language and compute their macro-average. It is observed for NLI task (Fig. 2) that the macro-precision of selected exemplars by SSP-CLT-ILP strategy is

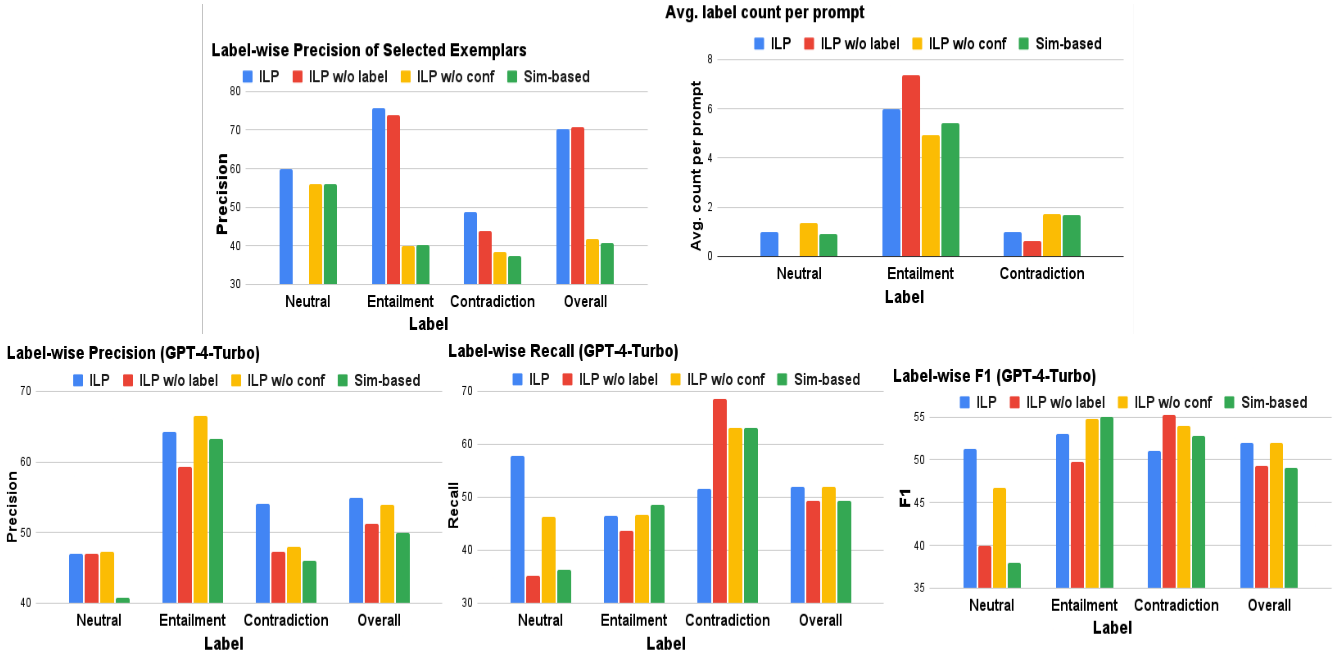


Figure 2: Label-wise statistics for AmericasNLI: Top row - Precision of ICL exemplars, Label-wise count per prompt, Bottom row - Label-wise Precision, Recall and F1 results using different selection strategies (GPT-4-Turbo)

consistently higher than it's other ablation variants, the least value being of w/o both (similarity-based) variant. This implies that the ILP is able to effectively sample high-precision exemplars which, in turn, are translated into superior F1 performance of SSP-CLT-ILP compared to other strategies.

For completeness, we also showcase the exemplar precision statistics for NER (label-wise) and POS (overall, for brevity) in Figure 3. The trends hold similar in the sense-that 'w/o confidence' and 'similarity-based' variants have significantly lower precision than SSP-CLT-ILP. This is expected because both the 'w/o confidence' and 'similarity-based' variants don't take into account the quality of predicted labels and are likely to sample sentences with incorrectly predicted labels, owing to high sentence similarity. This gap in precision of selected exemplars is translated into the downstream performance, as evident in tables 1 and 2. On the other hand, the 'w/o label' variant is competitive, unlike in NLI, in terms of both exemplars' precision as well as downstream performance for sequence labeling tasks.

### 5.1 Qualitative Analysis: SSP-CLT-SIM

We present the analysis for the gains obtained via SSP-CLT-SIM for Germanic POS in Figure 4. The confusion matrix difference between SSP-CLT-SIM and CLT-SIM suggests that the model misclassifies auxiliary verbs as verbs in CLT-SIM,

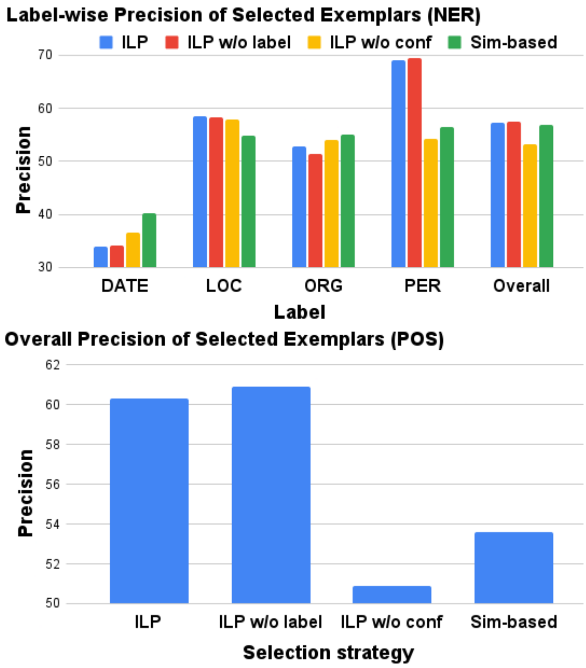


Figure 3: Top:Label-wise and overall precision of selected exemplars for Arican NER, Bottom: Overall precision of selected exemplars for Germanic POS

| Gold \ Predicted | ADJ | ADP | ADV | AUX | CCONJ | DET | NOUN | PRON | PROPN | PUNCT | VERB | X |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ADJ | -2 | 0 | 0 | 0 | 0 | 2 | -5 | 4 | 0 | 0 | 1 | 1 |
| ADP | -2 | 6 | -3 | 0 | 0 | 0 | 0 | -3 | 0 | 0 | -1 | 4 |
| ADV | -5 | -3 | 28 | 0 | 1 | -6 | -1 | -5 | 0 | 0 | -6 | -4 |
| AUX | 0 | -1 | -2 | 17 | 0 | 0 | 0 | -1 | -1 | 0 | -13 | 1 |
| CCONJ | 0 | -4 | -1 | 0 | 7 | 0 | 1 | -3 | 0 | 0 | -1 | 0 |
| DET | 1 | 1 | -4 | 0 | 0 | 9 | 0 | -3 | -4 | 0 | 0 | 0 |
| NOUN | 2 | 0 | 0 | -1 | 0 | -2 | 7 | -3 | 0 | 0 | -3 | 1 |
| PRON | -3 | -3 | -5 | -1 | 0 | 2 | -3 | 24 | -4 | 0 | -4 | -2 |
| PROPN | 0 | 0 | 0 | 0 | 0 | 0 | -2 | 0 | -1 | 0 | 0 | 3 |
| PUNCT | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | -2 | 0 | -1 |
| VERB | 0 | -1 | 0 | 4 | 0 | -1 | -15 | 0 | 0 | 0 | 15 | -2 |
| X | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | -1 | -1 | 0 | 1 |

Figure 4: Difference in confusion matrices between SSP-CLT-SIM and CLT-SIM for the POS task, summed across all languages (tags with less than 100 instances have been omitted). The increase in correct tags is visible along the diagonal, and misclassifications between VERB and AUX tags / NOUN and VERB tags have also improved.

and this is corrected in SSP-CLT-SIM. These errors are a consequence of the labels on the in-context exemplars the model receives, and not the tokens of the language itself.

We highlight this via the two Swiss-German POS examples in Figure 5. The misclassified verbs are corrected by SSP-CLT-SIM, and these labels are again misclassified when more than half of the labels in the in-context exemplars are corrupted.

## 6 Conclusions and Future Work

We present a novel SSP-CLT framework for Self-Supervised Prompting in Cross-Lingual Transfer settings. Our goal is to utilize target low-resource language's test instances (while not utilizing the gold labels) in a self-supervised fasion. We develop on top of Ada-002-embedding-based retrieval for cross-lingual prompting in stage 1, followed by in-language prompting in stage 2, while utilizing stage 1 labels as stage 2 exemplars. We observe consistent gains of stage 2 over stage 1 results across 3 LLMs - LLama-2-70b, GPT-3.5-Turbo and GPT-4x models. We term this method SSP-CLT-SIM.

We next seek to utilize the smaller fine-tuned models for stage 1. For this purpose, we additionally leverage their prediction probabilities (based on logits) from stage 1 along with the Ada-002 similarity scores. Moreover, we enforce the coverage of all labels for the given task in the selected exemplars via an Integer Linear Programming (ILP) framework that maximizes the aggregated similarity scores of selected exemplars, while ensuring their confidence scores being higher than a threshold (heuristically set to $90^{th}$ percentile probabilty score for each label), and each label being covered at least once. The results show consistent gains of SSP-CLT-ILP compared to SSP-CLT-SIM, despite incurring half the cost of LLM inference. The ablations show that each component of SSP-CLT-ILP is useful across tasks - Label coverage being crucial for NLI and Confidence thresholding being for NER and POS. Our detailed analysis show that ILP approach is able to effectively sample more high-precision exemplars compared to other retrieval strategies across tasks, and this, in turn, results in the overall superior performance for the downstream task at hand.

In future, we seek to extend our technique to more non-trivial applications such as cross-lingual generation, semantic parsing, etc. We also posit that smaller fine-tuned models, when calibrated properly, can result in more efficient selection of exemplars to an LLM, as compared to poorly calibrated counterparts, in terms of downstream performance. We leave a careful and systematic investigation into this hypothesis for future work. Moreover, we currently cover the languages having roman scripts, but we seek to extend our work for non-roman script languages as well in future.

## 7 Limitations

We show all our results and ablations on the recent state-of-the-art LLMs including GPT4. The inference for these LLMs is expensive, and makes the model deployment infeasible. Other potential limitations are extending our method to tasks such as fact checking, in which the LLMs suffer from *hallucinations* and overprediction issues. The reason why we don't use LLM logits in ILP framework is because they are not openly released by OpenAI and hence, there becomes a need to rely on smaller fine-tuned models - which can potentially lead to sub-optimal downstream performance, in case the fine-tuned models are poorly calibrated. Another serious implication of using LLMs for non-roman script languages is unreasonably high *fertility* (tokens per word split) of the LLM tokenizers, which increases the cost as well as strips the input prompt, which is not desirable.

## References


Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

David Ifeoluwa Adelani, Jade Abbott, Graham Neubig, Daniel D'souza, Julia Kreutzer, Constantine Lignos, Chester Palen-Michel, Happy Buzaaba, Shruti Rijhwani, Sebastian Ruder, et al. 2021. Masakhaner: Named entity recognition for african languages. *Transactions of the Association for Computational Linguistics*, 9:1116–1131.

Sweta Agrawal, Chunting Zhou, Mike Lewis, Luke Zettlemoyer, and Marjan Ghazvininejad. 2022. In-context examples selection for machine translation. In *Annual Meeting of the Association for Computational Linguistics*.

Kabir Ahuja, Rishav Hada, Millicent Ochieng, Prachi Jain, Harshita Diddee, Samuel Maina, Tanuja Ganu, Sameer Segal, Maxamed Axmed, Kalika Bali, et al. 2023. Mega: Multilingual evaluation of generative ai. *arXiv preprint arXiv:2303.12528*.

Jesujoba O Alabi, David Ifeoluwa Adelani, Marius Mosbach, and Dietrich Klakow. 2022. Adapting pre-trained language models to african languages via multilingual adaptive fine-tuning. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4336–4349.

Akari Asai, Sneha Kudugunta, Xinyan Velocity Yu, Terra Blevins, Hila Gonen, Machel Reid, Yulia Tsvetkov, Sebastian Ruder, and Hannaneh Hajishirzi. 2023. Buffet: Benchmarking large language models for few-shot cross-lingual transfer. *arXiv preprint arXiv:2305.14857*.

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.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2022. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Édouard 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.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *North American Chapter of the Association for Computational Linguistics*.

Abteen Ebrahimi, Manuel Mager, Arturo Oncevay, Vishrav Chaudhary, Luis Chiruzzo, Angela Fan, John Ortega, Ricardo Ramos, Annette Rios Gonzales, Ivan Meza-Ruiz, et al. 2022. Americasnli: Evaluating zero-shot natural language understanding of pretrained multilingual models in truly low-resource languages. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6279–6299.

Shivanshu Gupta, Matt Gardner, and Sameer Singh. 2023. Coverage-based example selection for in-context learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13924–13950, Singapore. Association for Computational Linguistics.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. *arXiv preprint arXiv:2111.09543*.

Moritz Laurer, Wouter van Atteveldt, Andreu Salleras Casas, and Kasper Welbers. 2022. Less Annotating, More Classifying – Addressing the Data Scarcity Issue of Supervised Machine Learning with Deep Transfer Learning and BERT - NLI. *Preprint*. Publisher: Open Science Framework.

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.

Benjamin Muller, Antonios Anastasopoulos, Benoît Sagot, and Djamé Seddah. 2021. When being unseen from mbert is just the beginning: Handling new languages with multilingual language models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 448–462.

Akshay Nambi, Vaibhav Balloli, Mercy Ranjit, Tanuja Ganu, Kabir Ahuja, Sunayana Sitaram, and Kalika Bali. 2023. Breaking language barriers with a leap: Learning strategies for polyglot llms. *arXiv preprint arXiv:2305.17740*.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajic, Christopher D Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and Daniel Zeman. 2020. Universal dependencies v2: An evergrowing multilingual treebank collection. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4034–4043.

Siqi Ouyang, Rong Ye, and Lei Li. 2022. On the impact of noises in crowd-sourced data for speech translation. In *Proceedings of the 19th International Conference on Spoken Language Translation (IWSLT 2022)*, pages 92–97, Dublin, Ireland (in-person and online). Association for Computational Linguistics.

Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2020. Mad-x: An adapter-based framework for multi-task cross-lingual transfer. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7654–7673.

Vipul Rathore, Rajdeep Dhingra, Parag Singla, et al. 2023. Zgul: Zero-shot generalization to unseen languages using multi-source ensembling of language adapters. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6969–6987.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4512–4525.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. 2023. Quantifying language models' sensitivity to spurious features in prompt design or: How i learned to start worrying about prompt formatting. *ArXiv*, abs/2310.11324.

Eshaan Tanwar, Subhabrata Dutta, Manish Borthakur, and Tanmoy Chakraborty. 2023. Multilingual LLMs are better cross-lingual in-context learners with alignment. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6292–6307, Toronto, Canada. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Ahmet Üstün, Arianna Bisazza, Gosse Bouma, and Gertjan van Noord. 2020. Udapter: Language adaptation for truly universal dependency parsing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2302–2315.

Anton Voronov, Lena Wolf, and Max Ryabinin. 2024. Mind your format: Towards consistent evaluation of in-context learning improvements. *arXiv preprint arXiv:2401.06766*.

Xingchen Wan, Ruoxi Sun, Hootan Nakhost, Hanjun Dai, Julian Eisenschlos, Sercan Arik, and Tomas Pfister. 2023. Universal self-adaptive prompting. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7437–7462, Singapore. Association for Computational Linguistics.

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.

Ningyu Zhang, Luoqiu Li, Xiang Chen, Shumin Deng, Zhen Bi, Chuanqi Tan, Fei Huang, and Huajun Chen. 2021. Differentiable prompt makes pre-trained language models better few-shot learners. *arXiv preprint arXiv:2108.13161*.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *International Conference on Machine Learning*, pages 12697–12706. PMLR.

# A Implementation and Hyperparameter Details

We use Azure OpenAI service [2] for all experiments involving GPT-3x and GPT-4x models. LLama-2-70b has been inferred on AMD A100 node having 8 GPUs. We set temperature as 0.0 consistently for all our experiments, making our results reproducible. The max_tokens (max. no. of generated tokens) parameter is set to 1024 for POS and NER tasks, while 15 for the NLI. For all experiments, the no. of exemplars (M) is set equal to 8 for fair comparison.

# B Prompt details

Prompts for the Named Entity Recognition (NER) and Part of Speech Tagging (POS) tasks are presented in the tab separated format shown in B.0.2 and B.0.3 respectively.

Prompts for Natural Language Inference (NLI) initially used the framework in Ahuja et al. (2023) . To improve our performance, we changed the prompt to use Sclar et al. (2023)'s framework, where the authors performed an exhaustive search over tokens used for a prompt in order to find the prompt with optimal performance. This increased Macro F1 score by atleast 10% across all the tested languages. We use the same prompt across all models used in our experiments.

### B.0.1 Natural Language Inference (NLI)

**Task Description:** You are an NLP assistant whose purpose is to solve Natural Language Inference (NLI) problems. NLI is the task of determining the inference relation between two (short, ordered) texts: entailment, contradiction, or neutral. Answer as concisely as possible in the same format as the examples below:
**Input format:**
{premise}
Question: Does this imply that {hypothesis}? Yes, No, or Maybe?
**Output format:**
Answer: {output}
**Verbalizer:**
Yes: Entailment
No: Contradiction
Maybe: Neutral

[2] https://azure.microsoft.com/en-in/products/ai-services/openai-service

### B.0.2 Named Entity Recognition (NER)

**Task Description:** Tag the following sentence according to the BIO scheme for the NER task, using the tags PER (person), LOC (location), ORG (organization) and DATE (date). Follow the format specified in the examples below:
**Input format:**
Sentence: $w_1$ $w_2$ ... $w_T$
**Output format:**
Tags:
$w_1$<TAB>$o_1$
$w_2$<TAB>$o_2$
...
$w_T$<TAB>$o_T$
**Verbalizer:**
Extract the sequence of labels $o_1, o_2, ...o_3$ from generated response.

### B.0.3 Part of Speech (PoS) tagging

**Task Description:** Tag the following sentence according to the Part of Speech (POS) of each word. The valid tags are ADJ, ADP, ADV, AUX, CCONJ, DET, INTJ, NOUN, NUM, PART, PRON, PROPN, PUNCT, SCONJ, SYM, VERB, X. Follow the format specified in the examples below:
**Input format:**
Sentence: $w_1$ $w_2$ ... $w_T$
**Output format:**
Tags:
$w_1$<TAB>$o_1$
$w_2$<TAB>$o_2$
...
$w_T$<TAB>$o_T$
**Verbalizer:**
Extract the sequence of labels $o_1, o_2, ...o_3$ from generated response.

## B.1 Prompts for GSW Examples

The base SSP-CLT-SIM prompts for the GSW examples highlighted in Figure 5 are given below. Labels which are misclassified in the in-context exemplars are coloured in red, and the AUX labels which are to be flipped in the ablations are coloured in blue. It is interesting to note that examples 1 and 2 are similar, as example 1 is retrieved as an in-context exemplar for example 2.

### B.1.1 Example 1

Tag the following sentence according to the Part of Speech (POS) of each word. The valid tags are ADJ, ADP, ADV, AUX, CCONJ, DET, INTJ, NOUN, NUM, PART, PRON, PROPN, PUNCT,

| | Ds | Gueten | isch | immerhin | gsi | , | dass | i | ungerdesse | söfu | müed | bi | gsi | , | dass | i | ändlech | ha | chönne | go | schlofe | . |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CLT-SIM** | DET | NOUN | AUX | ADV | VERB | PUNCT | SCONJ | PRON | ADV | VERB | ADJ | ADP | VERB | PUNCT | SCONJ | PRON | ADV | AUX | AUX | VERB | VERB | PUNCT |
| **SSP-CLT-SIM** | DET | NOUN | AUX | ADV | AUX | PUNCT | SCONJ | PRON | ADV | ADV | ADJ | ADP | AUX | PUNCT | SCONJ | PRON | ADV | AUX | AUX | PART | VERB | PUNCT |
| **SSP-CLT-SIM** (Half AUX->VERB) | DET | NOUN | AUX | ADV | AUX | PUNCT | SCONJ | PRON | ADV | ADV | ADJ | ADP | AUX | PUNCT | SCONJ | PRON | ADV | AUX | AUX | PART | VERB | PUNCT |
| **SSP-CLT-SIM** (All AUX->VERB) | DET | NOUN | VERB | ADV | VERB | PUNCT | SCONJ | PRON | ADV | ADV | ADJ | ADP | VERB | PUNCT | SCONJ | PRON | ADV | AUX | AUX | VERB | VERB | PUNCT |
| **Gold** | DET | NOUN | AUX | ADV | AUX | PUNCT | SCONJ | PRON | ADV | ADV | ADJ | AUX | AUX | PUNCT | SCONJ | PRON | ADV | AUX | AUX | PART | VERB | PUNCT |

| | I | cha | der | ihri | Telefonnummere | gä | , | de | nimmsch | mou | unverbindlech | Kontakt | uuf | . |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CLT-SIM** | PRON | VERB | DET | ADJ | NOUN | VERB | PUNCT | PRON | VERB | ADV | ADJ | NOUN | VERB | PUNCT |
| **SSP-CLT-SIM** | PRON | AUX | PRON | PRON | NOUN | VERB | PUNCT | PRON | VERB | ADV | ADJ | NOUN | ADP | PUNCT |
| **SSP-CLT-SIM** (Half AUX->VERB) | PRON | AUX | PRON | PRON | NOUN | VERB | PUNCT | PRON | VERB | ADV | ADJ | NOUN | ADP | PUNCT |
| **SSP-CLT-SIM** (All AUX->VERB) | PRON | VERB | PRON | PRON | NOUN | VERB | PUNCT | DET | VERB | ADV | ADJ | NOUN | ADP | PUNCT |
| **Gold** | PRON | AUX | PRON | DET | NOUN | VERB | PUNCT | ADV | VERB | ADV | ADJ | NOUN | PART | PUNCT |

Figure 5: Label flips for CLT-SIM and SSP-CLT-SIM, for POS tagging in Swiss-German (gsw). Incorrect labels are marked in red. SSP-CLT-SIM ablations include flipping half/all of the AUX labels in the prompt to VERB labels. Gold labels are given for reference.

SCONJ, SYM, VERB, X. Follow the format specified in the examples below:
Sentence: I main , das Ganze letscht Wuchä isch mier scho ächli iigfaarä .
Tags:
‘‘‘
I PRON
main VERB
, PUNCT
das DET
Ganze NOUN
letscht ADJ
Wuchä NOUN
isch AUX
mier PRON
scho ADV
ächli ADV
iigfaarä VERB
. PUNCT
‘‘‘
Sentence: Du gsehsch uus , wi wenn de nöime no hättisch z trinken übercho .
Tags:
‘‘‘
Du PRON
gsehsch VERB
uus PRON
, PUNCT
wi SCONJ
wenn SCONJ
de DET
nöime ADJ
no ADV
hättisch AUX
z PART
trinken VERB
übercho VERB
. PUNCT
‘‘‘
Sentence: Dir weit mer doch nid verzöue , di Wäutsche heige vo eim Tag uf en anger ufghört Chuttlen ässe .
Tags:
‘‘‘
Dir PRON
weit VERB
mer PRON
doch ADV
nid ADV
verzöue VERB
, PUNCT
di DET
Wäutsche NOUN
heige VERB
vo ADP
eim DET
Tag NOUN
uf ADP
en DET
anger ADJ
ufghört VERB
Chuttlen NOUN
ässe VERB
. PUNCT
‘‘‘
Sentence: es isch nämli echt usgstorbe gsi .
Tags:
‘‘‘
es PRON
isch AUX
nämli ADV
echt ADJ
usgstorbe VERB
gsi AUX

. PUNCT
“‘
Sentence: Aso bini rächt uufgschmissä gsi und dem entschprächend fascht verzwiiflät .
Tags:
“‘
Aso ADV
bini AUX
rächt ADV
uufgschmissä VERB
gsi AUX
und CCONJ
dem PRON
entschprächend ADJ
fascht ADV
verzwiiflät VERB
. PUNCT
“‘
Sentence: Der Ääschme wett nöd schaffe biin em .
Tags:
“‘
Der DET
Ääschme NOUN
wett AUX
nöd ADV
schaffe VERB
biin ADP
em PRON
. PUNCT
“‘
Sentence: Zerscht hends am Dani gsait , är söli dòch Hoochdütsch redä , das gängi denn grad gaar nöd , wenn är so redi , wiäner redi .
Tags:
“‘
Zerscht ADV
hends PRON
am ADP
Dani PROPN
gsait VERB
, PUNCT
är PRON
söli AUX
dòch ADV
Hoochdütsch ADJ
redä VERB
, PUNCT
das PRON
gängi VERB
denn ADV
grad ADV
gaar ADV
nöd ADV
, PUNCT
wenn SCONJ
är PRON
so ADV
redi VERB
, PUNCT
wiäner PRON
redi VERB
. PUNCT
“‘
Sentence: Isch das e Sach gsi , bis mer se gfunge hei gha .
Tags:
“‘
Isch AUX
das PRON
e DET
Sach NOUN
gsi AUX
, PUNCT
bis SCONJ
mer PRON
se PRON
gfunge VERB
hei AUX
gha VERB
. PUNCT
“‘
Sentence: Ds Gueten isch immerhin gsi , dass i ungerdesse söfu müed bi gsi , dass i ändlech ha chönne go schlofe .
Tags:
“‘

### B.1.2 Example 2

Tag the following sentence according to the Part of Speech (POS) of each word. The valid tags are ADJ, ADP, ADV, AUX, CCONJ, DET, INTJ, NOUN, NUM, PART, PRON, PROPN, PUNCT, SCONJ, SYM, VERB, X. Follow the format specified in the examples below:
Sentence: I ha ar Marie-Claire gseit , es sig mer chli schlächt und i mög jetz nümm liire .
Tags:
“‘
I PRON
ha AUX
ar PART
Marie-Claire PROPN
gseit VERB

, PUNCT
es PRON
sig AUX
mer PRON
chli ADV
schlächt ADJ
und CCONJ
i PRON
mög VERB
jetz ADV
nümm ADV
liire VERB
. PUNCT
‘‘‘

Sentence: De Spanier hed de Kontakt vermettlet , d Rumäne sölled d Holländer ombrocht ha .
Tags:
‘‘‘
De DET
Spanier NOUN
hed AUX
de DET
Kontakt NOUN
vermettlet VERB
, PUNCT
d DET
Rumäne NOUN
sölled AUX
d DET
Holländer PROPN
ombrocht VERB
ha AUX
. PUNCT
‘‘‘

Sentence: Ds Gueten isch immerhin gsi , dass i ungerdesse söfu müed bi gsi , dass i ändlech ha chönne go schlofe .
Tags:
‘‘‘
Ds DET
Gueten NOUN
isch AUX
immerhin ADV
gsi VERB
, PUNCT
dass SCONJ
i PRON
ungerdesse ADV
söfu VERB
müed ADJ
bi ADP
gsi VERB
, PUNCT
dass SCONJ
i PRON
ändlech ADV
ha AUX
chönne AUX
go VERB
schlofe VERB
. PUNCT
‘‘‘

Sentence: Isch das e Sach gsi , bis mer se gfunge hei gha .
Tags:
‘‘‘
Isch AUX
das PRON
e DET
Sach NOUN
gsi AUX
, PUNCT
bis SCONJ
mer PRON
se PRON
gfunge VERB
hei AUX
gha VERB
. PUNCT
‘‘‘

Sentence: De Dialäkt muess zu de Gschecht und zum Inhaut vonere Werbig passe .
Tags:
‘‘‘
De DET
Dialäkt NOUN
muess AUX
zu ADP
de DET
Gschecht NOUN
und CCONJ
zum ADP
Inhaut NOUN
vonere ADP
Werbig NOUN
passe VERB
. PUNCT
‘‘‘

Sentence: Mit der Zit hani mi mit mir säuber uf ei Schriibwiis pro Wort aafo einige .
Tags:
‘‘‘
Mit ADP
der DET

Zit NOUN
hani VERB
mi PRON
mit ADP
mir PRON
säuber ADJ
uf ADP
ei DET
Schriibwiis NOUN
pro ADP
Wort NOUN
aafo VERB
einige DET
. PUNCT
‘‘‘

Sentence: Mit all denä Wörter hani natürli nüt chönä aafangä .
Tags:
‘‘‘
Mit ADP
all DET
denä DET
Wörter NOUN
hani PRON
natürli ADV
nüt ADV
chönä VERB
aafangä VERB
. PUNCT
‘‘‘

Sentence: Aso bini rächt uufgschmissä gsi und dem entschprächend fascht verzwiiflät .
Tags:
‘‘‘
Aso ADV
bini AUX
rächt ADV
uufgschmissä VERB
gsi AUX
und CCONJ
dem PRON
entschprächend ADJ
fascht ADV
verzwiiflät VERB
. PUNCT
‘‘‘

Sentence: I cha der ihri Telefonnummere gä , de nimmsch mou unverbindlech Kontakt uuf .
Tags:
‘‘‘

# C Source and Target Languages for each task

| Language Family | Source languages | Source size |
|---|---|---|
| Germanic | {En,Is,De} | 30000 |
| African | {En,Am,Sw,Wo} | 19788 |
| Americas | {En,Es} | 19998 |

Table 5: Combined Source (Training) languages' data size (# Sentences)

| Language Family | Test languages | Test size |
|---|---|---|
| Germanic | {Fo, Got, Gsw} | 100 |
| African | {Hau,Ibo,Kin,Lug,Luo} | 100 |
| Americas | {Aym,Gn,Quy,Nah} | 99 |

Table 6: Combined Source (Training) languages' data size (# Sentences)

| Code | Language |
|---|---|
| En | English |
| Am | Amharic |
| Sw | Swahili |
| Wo | Wolof |
| Hau | Hausa |
| Ibo | Igbo |
| Kin | Kinyarwanda |
| Lug | Luganda |
| Luo | Luo |
| Is | Icelandic |
| De | German |
| Fo | Faroese |
| Got | Gothic |
| Gsw | Swiss German |
| Es | Spanish |
| Aym | Aymara |
| Gn | Guarani |
| Quy | Quechua |
| Nah | Nahuatl |

Table 7: Languages and their codes

Premise: Ah, huk chaypi allinqa apakurqa allin qawasqayqa paniypa ñawpaq yuyariyninmi, chaypas hina hipa pampapim karqa.
Hypothesis: Yuyaruniqa hipa pampapi huk ima apakusqantam.
Answer: entailment

Premise: Yaykuykuptiykuqa punkukunaqa wichqasqam kachkarqa.
Hypothesis: Punku wichqasqa kachkaptinpas yaykurqanikum.
Answer: entailment

Premise: Yanapawaqniy atiq sispasmi hatun llaqtapa waklawninpiraq tiyan.
Hypothesis: Yanapawaqniy warmi warman 5 millas nisqan karupiraq tiyan.
Answer: **neutral**

Premise: Manam mayman risqanta yacharqanikuchu.
Hypothesis: Mayman risqantam yacharqaniku.
Answer: entailment

Premise: Chayna kaptinqa hamutachkanim huktapiwan Ramonawan rimariyta.
Hypothesis: Ramonawanmi huktapiwan rimarqani.
Answer: entailment

Premise: Ripukusqañam hinaspam amaña llakikunaypaq niwarqa.
Hypothesis: Ama llakikunaytam niwarqa.
Answer: entailment

Premise: Ichapasyá huk kaq mana yachasqaymanta hamun ichaqa
Hypothesis: Apurawtam hamun, ichaqa maymanta hamusqanta yachanim.
Answer: entailment

Premise: Locust Hill oh awriki, ari, kusa
Hypothesis: Locust Hill nisqaqa allinmi.
Answer: contradiction

Premise: Oh, payllam isqun iskay iskayraq regulador nisqapi inyecciónta qinaq karqa.
Hypothesis: Martes punchawtam inyector nisqata hinarqani.
Answer: **neutral**

Premise: Ah, huk chaypi allinqa apakurqa allin qawasqayqa paniypa ñawpaq yuyariyninmi, chaypas hina hipa pampapim karqa.
Hypothesis: Yuyaruniqa hipa pampapi huk ima apakusqantam.
Answer: entailment

Premise: Yaykuykuptiykuqa punkukunaqa wichqasqam kachkarqa.
Hypothesis: Punku wichqasqa kachkaptinpas yaykurqanikum.
Answer: entailment

Premise: Manam mayman risqanta yacharqanikuchu.
Hypothesis: Mayman risqantam yacharqaniku.
Answer: entailment

Premise: Chayna kaptinqa hamutachkanim huktapiwan Ramonawan rimariyta.
Hypothesis: Ramonawanmi huktapiwan rimarqani.
Answer: entailment

Premise: Manam pachay karqachu ima kaqpas ruranaypaq.
Hypothesis: Mana pacha llapan qinanaypaq haypawarqachu
Answer: entailment

Premise: Ripukusqañam hinaspam amaña llakikunaypaq niwarqa.
Hypothesis: Ama llakikunaytam niwarqa.
Answer: entailment

Premise: Ichapasyá huk kaq mana yachasqaymanta hamun ichaqa
Hypothesis: Apurawtam hamun, ichaqa maymanta hamusqanta yachanim.
Answer: entailment

Premise: Locust Hill oh awriki, ari, kusa
Hypothesis: Locust Hill nisqaqa allinmi.
Answer: contradiction

Premise: Oh, payllam isqun iskay iskayraq regulador nisqapi inyeccióntа qinaq karqa.  Hypothesis: Martes punchawtam inyector nisqata hinarqani.
Answer: **contradiction**

Figure 6: Correct case of 'Neutral' detected by ILP (left), while 'w/o label' variant misses it (right). We note that exact one 'neutral' class has been sampled by ILP, while no 'neutral' is sampled in 'w/o label' version.