CrossAligner & Co: Zero-Shot Transfer Methods for Task-Oriented Cross-lingual Natural Language Understanding

Milan Gritta[1,†], **Ruoyu Hu**[2,†,*] and **Ignacio Iacobacci**[1]
[1]Huawei Noah's Ark Lab, London, UK
[2]Imperial College London, UK
{milan.gritta,ignacio.iacobacci}@huawei.com
ruoyu.hu18@imperial.ac.uk

Abstract

Task-oriented personal assistants enable people to interact with a host of devices and services using natural language. One of the challenges of making neural dialogue systems available to more users is the lack of training data for all but a few languages. Zero-shot methods try to solve this issue by acquiring task knowledge in a high-resource language such as English with the aim of transferring it to the low-resource language(s). To this end, we introduce **CrossAligner**, the principal method of a variety of effective approaches for zero-shot cross-lingual transfer based on learning alignment from unlabelled parallel data. We present a quantitative analysis of individual methods as well as their weighted combinations, several of which exceed state-of-the-art (SOTA) scores as evaluated across nine languages, fifteen test sets and three benchmark multilingual datasets. A detailed qualitative error analysis of the best methods shows that our fine-tuned language models can zero-shot transfer the task knowledge better than anticipated.

1 Introduction

Natural language understanding (NLU) refers to the ability of a system to 'comprehend' the meaning (semantics) and the structure (syntax) of human language (Wang et al., 2019) to enable the interaction with a system or device. Cross-lingual natural language understanding (XNLU) alludes to a system that is able to handle multiple languages simultaneously (Artetxe and Schwenk, 2019; Hu et al., 2020). We focus on task-oriented XNLU that comprises two correlated objectives: i) Intent Classification, which identifies the type of user command, e.g. 'edit_reminder', 'send_message' or 'play_music' and ii) Entity/Slot Recognition, which identifies relevant entities in the utterance including their types such as dates, messages, music tracks, locations, etc. In a modular dialogue system, this information is used by the dialogue manager to decide how to respond to the user (Casanueva et al., 2017; Gritta et al., 2021). For neural XNLU systems, the limited availability of annotated data is a significant barrier to scaling dialogue systems to more users (Razumovskaia et al., 2021). Therefore, we can use cross-lingual methods to zero-shot transfer the knowledge learnt in a high-resource language such as English to the target language of choice (Artetxe et al., 2020; Siddhant et al., 2020). To this end, we introduce a variety of alignment methods for zero-shot cross-lingual transfer, most notably **CrossAligner**. Our methods leverage unlabelled parallel data and can be easily integrated on top of a pretrained language model, referred to as **XLM**[1], such as XLM-RoBERTa (Conneau et al., 2020). Our methods help the XLM align its cross-lingual representations while optimising the primary XNLU tasks, which are learned only in the source language and transferred zero-shot to the target language. Finally, we also investigate the effectiveness of simple and weighted combinations of multiple alignment losses, which leads to further model improvements and insights. Our contributions are summarised as follows:

- We introduce CrossAligner, a cross-lingual transfer method that achieves SOTA performance on three benchmark XNLU datasets.
- We introduce Translate-Intent, a simple and effective baseline, which outperforms its commonly used counterpart 'Translate-Train'.
- We introduce Contrastive Alignment, an auxiliary loss that leverages contrastive learning at a much smaller scale than past work.
- We introduce weighted combinations of the above losses to further improve SOTA scores.
- Qualitative analysis aims to guide future research by examining the remaining errors.

*Work conducted as Research Intern at Huawei's Noah's Ark Lab, London. † - Equal contribution.

[1]Not to be confused with Lample and Conneau (2019).

Findings of the Association for Computational Linguistics: ACL 2022, pages 4048 - 4061

2 Related Work

Several approaches to zero-shot cross-lingual transfer exist and can broadly be divided into: a) *Data-based Transfer*, which focuses on training data transformation and b) *Model-based Transfer* that centres around modifying models' training routine.

Data-based Transfer Translating utterances for the intent classification task is relatively straightforward so previous works focused on projecting and/or aligning the entity labels between translated utterances. This is followed by standard supervised training with those pseudo-labels and is commonly known as the *translate-train* method. One of the earliest works still being used for this purpose is *fastalign* (Dyer et al., 2013). It's an unsupervised word aligner trained on a parallel corpus to map each word (thus its entity label) in the source utterance to the word(s) in the target user utterance. Projecting the entity labels can also be done with word-by-word translation and source label copying (Yi and Cheng, 2021). A teacher model then weakly labels the target data, which is used to train the final student model. Sometimes, this type of label projection is complemented with an additional entity alignment step (Li et al., 2021a). Better performance can be achieved by using machine translation with entity matching and distributional statistics (Jain et al., 2019) though this can be a costly process for each language. A category of 'word substitution' methods such as code-switching (Qin et al., 2020; Kuwanto et al., 2021) or dictionary-enhanced pretraining (Chaudhary et al., 2020) have also been shown to improve cross-lingual transfer.

Model-based Transfer Prior to the adoption of multilingual transformers (Lample and Conneau, 2019), task-oriented XNLU methods employed a BiLSTM encoder combined with different multilingual embeddings (Schuster et al., 2019). Newer approaches usually involve a pretrained XLM and the addition of some new training component(s) with the inference routine remaining mostly unchanged. Xu et al. (2020) learn to jointly align and predict entity labels by fusing the source and target language embeddings with attention and using the resulting cross-lingual representation for entity prediction. Qi and Du (2020) include an adversarial language detector in training whose loss encourages the model to generate language-agnostic sentence representations for improved zero-shot transfer. Pan et al. (2020) and Chi et al. (2020) added a contrastive loss to pretraining that treats translated sentences as positive examples and unrelated sentences as negative samples. This training step helps the XLM produce similar embeddings in different languages. However, these methods require large annotated datasets and expensive model pretraining (Chi et al., 2020). Our proposed methods only use the English task data (which is relatively limited) and its translations for each language.

The most related prior works are Arivazhagan et al. (2019) for machine translation and Gritta and Iacobacci (2021) for task-oriented XNLU. Both of these are cross-lingual alignment methods that use translated training data to zero-shot transfer the source language model to the target language. We focus on the latter work, called XeroAlign, which reported the most recent SOTA scores on our evaluation datasets. XeroAlign works by generating a sentence embedding of the user utterance for each language, e.g. English (source) and Thai (target) using the CLS token of the XLM. A Mean Squared Error loss function minimises the difference between the multilingual sentence embeddings and is backpropagated along with the main task loss. XeroAlign aims to bring sentence embeddings in different languages closer together with a bias towards intent classification due to the CLS embedding, which is the standard input to the intent classifier. We reproduce this method for analysis and comparisons but add a small post-processing step that distinctly improves the reported scores.

3 Methodology

3.1 CrossAligner

Intuition We introduce CrossAligner, the most notable of our proposed cross-lingual alignment methods, outlined in Algorithm 1. CrossAligner enables effective zero-shot transfer by leveraging unlabelled parallel data for our new language-agnostic objective created through a transformation of the English entity labels. CrossAligner was borne out of early error analysis where we observed that the model incorrectly predicted entities that didn't occur in the input and failed to predict entities that did occur in the input. Using this insight as our main motivation, the essence of CrossAligner is being able to exploit information about the *presence of entities/slots in the user utterance*.

Algorithm We have used a proprietary service (Huawei Translate) to translate the English user ut-

terances X_{Eng} into each target language X_{Tar}, however, a publicly available translator can also be used. Note that we use the same translations for each of our alignment methods to compare them fairly. Our language-agnostic objective is created by transforming the English slot labels y_{ec} into a fixed binary vector y_{ca} indicating which entities are present in the input (lines 1-7 in Algorithm 1), irrespective of the frequency of their occurrence.

The standard XNLU training (lines 15-20) features an Intent Classifier (IC) and an Entity Classifier (EC). Each computes a cross-entropy loss (ce_loss) with a softmax activation using English labelled data (multi-class classification). This yields the standard losses $\mathcal{L}_{ic}$ and $\mathcal{L}_{ec}$. The CrossAligner (CA) classifier then pools the EC logits matrix by reshaping it into a long vector (lines 24 and 29) and predicts which entities are present in the user utterance (multi-label classification). We compute a Binary Cross-Entropy loss (bce_loss) with a sigmoid activation between the predicted labels $pred_{eng}$ and $pred_{tar}$ (for English and Target languages respectively) and our language-agnostic labels y_{ca} (lines 26 and 31). This yields the CrossAligner losses $\mathcal{L}_{eng}$ and $\mathcal{L}_{tar}$. The fact that these gradients are propagated through the EC to the XLM token embeddings ensures a good alignment for entity/slot recognition, as shown in the results section. Note that EC, IC and CA are shared between languages to aid zero-shot cross-lingual transfer.

BIO versus IO Using the BIO sequence tagging format (Sang and De Meulder, 2003) can introduce easily avoidable model errors, e.g. predicting a B-tag after an I-tag, two B-tags in succession or skipping the B-tag altogether. We have therefore simplified the training process by making it agnostic w.r.t. the entity's BI order. The B-tags were removed in preprocessing, meaning the entity classifier predicts only IO-tags. At inference, the B-tags get restored with a simple post-processing rule. Note that all our models use this IO-only training.

Architecture We use a common task-oriented XNLU model that employs a pretrained XLM, e.g. JointBERT (Chen et al., 2019). The IC, EC and CA each feature a single multi-layer perceptron of sizes: $[\text{hidden\_size}, \text{len}(\text{intent\_classes})]$, $[\text{hidden\_size}, \text{len}(\text{entity\_classes})]$ and $[\text{seq\_len} \times \text{len}(\text{entity\_classes}), \text{len}(\text{entity\_classes})]$. Depending on the dataset, seq_len varies between 50-100 tokens. The model architecture is shown in Fig 1.

Algorithm 1 The CrossAligner alignment/loss.

1: **function** TRANSFORMLABELS(y_{ec})
2: $y_{ca} \leftarrow \text{zeros}(\text{len}(\text{entity\_classes}))$
3: **for** entity $\in y_{ec}$ **do**
4: $y_{ca}[\text{index\_of}(\text{entity})] \leftarrow 1$
5: **end for**
6: **return** y_{ca}
7: **end function**

8: XLM $\leftarrow$ Cross-lingual language model
9: IC $\leftarrow$ Intent Classifier
10: EC $\leftarrow$ Entity Classifier
11: CA $\leftarrow$ CrossAligner Classifier
12: $X_{Eng} \leftarrow$ Standard training data in English
13: $X_{Tar} \leftarrow X_{Eng}$ translated into Target language

14: **for** $(x_{eng}, y), (x_{tar}, y) \in X_{Eng}, X_{Tar}$ **do**
—Standard XNLU Training—
15: $y_{ic}, y_{ec} \leftarrow y$
16: $cls_{eng}, tokens_{eng} \leftarrow \text{XLM}(x_{eng})$
17: $pred_{ic} \leftarrow \text{IC}(cls_{eng})$
18: $\mathcal{L}_{ic} \leftarrow \text{ce\_loss}(pred_{ic}, y_{ic})$
19: $pred_{ec} \leftarrow \text{EC}(tokens_{eng})$
20: $\mathcal{L}_{ec} \leftarrow \text{ce\_loss}(pred_{ec}, y_{ec})$

—CrossAligner Training—
21: $y_{ca} \leftarrow$ TRANSFORMLABELS(y_{ec})
22: $shape \leftarrow (\text{seq\_len} \times \text{len}(\text{entity\_classes}))$
23: $logits_{eng} \leftarrow \text{EC}(tokens_{eng})$
24: $logits_{eng}.\text{reshape\_matrix\_into}(shape)$
25: $pred_{eng} \leftarrow \text{CA}(logits_{eng})$
26: $\mathcal{L}_{eng} \leftarrow \text{bce\_loss}(pred_{eng}, y_{ca})$
27: $cls_{tar}, tokens_{tar} \leftarrow \text{XLM}(x_{tar})$
28: $logits_{tar} \leftarrow \text{EC}(tokens_{tar})$
29: $logits_{tar}.\text{reshape\_matrix\_into}(shape))$
30: $pred_{tar} \leftarrow \text{CA}(logits_{tar})$
31: $\mathcal{L}_{tar} \leftarrow \text{bce\_loss}(pred_{tar}, y_{ca})$
32: $\mathcal{L}_{total} \leftarrow \mathcal{L}_{ic} + \mathcal{L}_{ec} + \mathcal{L}_{eng} + \mathcal{L}_{tar}$
33: **end for**

3.2 Contrastive Alignment for XNLU

Our contrastive alignment is based on InfoNCE (Oord et al., 2018). Previous work has employed a contrastive loss for cross-lingual alignment (Pan et al., 2020), however, the datasets were out-of-domain and orders of magnitude larger. We show that strong results can be obtained using only in-domain (fine-tuning) data. Similar to (Wu et al., 2021), if given a randomly sampled batch of N English sentences X_{Eng} and its parallel sentences

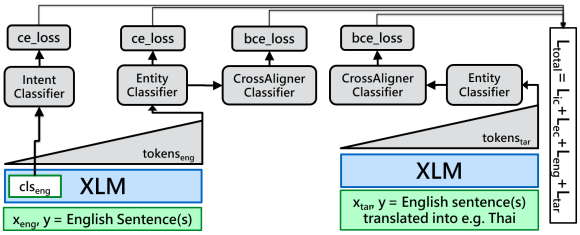

Figure 1: The architecture of CrossAligner. The parameters of the XLM model and all classifiers are shared between languages to enable cross-lingual transfer.

$\mathsf{X_{Tar}}$ in the target language, then the loss on the i^{th} sentence pair $\mathsf{x_{eng_i}} \in \mathsf{X_{Eng}}$ and $\mathsf{x_{tar_i}} \in \mathsf{X_{Tar}}$ equals:

$$\ell(\mathsf{x_{eng_i}}, \mathsf{x_{tar_i}}) = -\log \frac{e^{\mathsf{sim}(\mathsf{x_{eng_i}}, \mathsf{x_{tar_i}})}}{\sum_{\mathsf{k}=1}^{\mathsf{N}} e^{\mathsf{sim}(\mathsf{x_{eng_i}}, \mathsf{x_{tar_k}})}} \quad (1)$$

where $\mathsf{sim}(\mathsf{u}, \mathsf{v}) = \mathsf{u} \cdot \mathsf{v} \,/\, ||\mathsf{u}||_2 \cdot ||\mathsf{v}||_2$ is the cosine similarity between two sentence embeddings. A sentence $\mathsf{x_{eng_i}} \in \mathsf{X_{Eng}}$ symmetrically forms a positive pair with its translation $\mathsf{x_{tar_i}} \in \mathsf{X_{Tar}}$ while the other $\mathsf{N} - 1$ sentence embeddings are treated as negative samples. The batch loss is calculated as the average of all positive pair losses. Algorithm 2 below shows the steps that replace/complement the CrossAligner block (lines 21-32 in Algorithm 1).

Algorithm 2 The Contrastive Alignment loss.

1: $\mathsf{cls_{eng}}, \mathsf{tokens_{eng}} \leftarrow \mathsf{XLM}(\mathsf{x_{eng}})$
2: $\mathsf{cls_{tar}}, \mathsf{tokens_{tar}} \leftarrow \mathsf{XLM}(\mathsf{x_{tar}})$
3: $\mathsf{sim} \leftarrow \mathsf{batch\_cosine\_sim}(\mathsf{cls_{eng}}, \mathsf{cls_{tar}})$
4: $\mathsf{labels} \leftarrow \mathsf{arange}(\mathsf{batch\_size})$
5: $\mathcal{L}_{\mathsf{cl}} \leftarrow \mathsf{ce\_loss}(\mathsf{sim}, \mathsf{labels})$
6: $\mathcal{L}_{\mathsf{total}} \leftarrow \mathcal{L}_{\mathsf{ic}} + \mathcal{L}_{\mathsf{ec}} + \mathcal{L}_{\mathsf{cl}}$

3.3 Translate-Intent

The translate-train method is used in multilingual NLP as a competitive baseline (Liang et al., 2020; Hu et al., 2020). After machine translation, the sequence tagging tasks require an additional transformation, i.e. entity label projection and/or word alignment (Schuster et al., 2019; Li et al., 2021b; Xu et al., 2020). This is followed by supervised fine-tuning with the new pseudo-labels. However, both label projection and word alignment are *sources of common errors*. We therefore introduce a simpler baseline called Translate-Intent, which to the best of our knowledge, has not been featured in task-oriented XNLU. We omit the entity/slot recognition for the target language (given the unreliable pseudo-labels) and only use the IC, which is trained with the parallel data $\mathsf{X_{Tar}}$ (labels copied from English). Algorithm 3 above shows the steps that either replace or complement (in case of a combination of multiple losses) the CrossAligner steps, shown in lines 21-32 in Algorithm 1.

Algorithm 3 The Translate-Intent loss.

1: $\mathsf{cls_{tar}}, \mathsf{tokens_{tar}} \leftarrow \mathsf{XLM}(\mathsf{x_{tar}})$
2: $\mathsf{pred_{ic}} \leftarrow \mathsf{IC}(\mathsf{cls_{tar}})$
3: $\mathcal{L}_{\mathsf{ti}} \leftarrow \mathsf{ce\_loss}(\mathsf{pred_{ic}}, \mathsf{y_{ic}})$
4: $\mathcal{L}_{\mathsf{total}} \leftarrow \mathcal{L}_{\mathsf{ic}} + \mathcal{L}_{\mathsf{ec}} + \mathcal{L}_{\mathsf{ti}}$

3.4 Adaptive Weighting of Auxiliary Losses

In order to evaluate the benefits of combinations of two or more alignments, we employ the Multi-Loss Weighting with Coefficient of Variations (Groenendijk et al., 2021) technique (CoV) to calculate a weighted sum of auxiliary losses (Aux) that we add to the main XNLU losses $\mathcal{L}_{\mathsf{ic}}$ and $\mathcal{L}_{\mathsf{ec}}$ as follows:

$$\mathcal{L}_{\mathsf{total}} = \mathcal{L}_{\mathsf{ic}} + \mathcal{L}_{\mathsf{ec}} + \sum_{\mathsf{a} \in \mathsf{Aux}} \mathsf{w_a} \mathcal{L}_{\mathsf{a}} \quad (2)$$

The sole difference to CoV is that we opt to omit the loss weight normalisation step before application. The weights for an auxiliary loss $\mathcal{L}_{\mathsf{a,t}}$ for $\mathsf{a} \in \mathsf{Aux}$ at training step t are calculated as follows:

$$\mathsf{w_{a,t}} = \frac{\sigma_{\ell_{\mathsf{a,t}}}}{\mu_{\ell_{\mathsf{a,t}}}} \quad \ell_{\mathsf{a,t}} = \frac{\mathcal{L}_{\mathsf{a,t}}}{\mu_{\mathcal{L}_{\mathsf{a,t-1}}}} \quad (3)$$

where $\ell_{\mathsf{a,t}}$ is the loss ratio of loss $\mathsf{a} \in \mathsf{Aux}$ at training step t, σ is the standard deviation over the history of loss ratios and $\mu_{\ell_{\mathsf{a,t-1}}}$ is the mean of the loss ratio ℓ_{a} up to and including step $\mathsf{t} - 1$. We also compare CoV to a simple sum of all losses i.e. equal weight for each loss, as shown in Algorithms 1, 2 and 3 (line beginning with $\mathcal{L}_{\mathsf{total}}$).

4 Experimental Setup

Datasets Three multilingual datasets are used to compare our methods with their most relevant counterparts. The datasets, which are used as standard benchmarks for the XNLU tasks, comprise nine unique languages (de, pt, zh, ja, hi, tr, fr, es, th) from 15 test sets (20,000+ instances in total) featuring diverse examples of users interacting with task-oriented personal assistants designed to test the XNLU capabilities of multilingual models. Two related tasks are being evaluated, Intent Classification and Entity/Slot Recognition.

Models	MTOP (5)	MTOD (2)	M-ATIS (8)	MEAN (15)	Overall
Zero-Shot	91.7/77.1	94.1/75.1	91.1/79.9	91.7/76.5	84.1
Target Language	95.7/88.7	98.4/91.8	92.5/88.9	94.3/89.2	91.8
Translate-Train SOTA	94.5/77.9	97.5/67.9	94.9/78.0	95.1/76.6	85.9
Translate-Intent (Ours)	95.2/77.1	98.1/76.5	95.9/80.0	95.9/78.5	87.2
Previous SOTA	**95.6**/80.3	**98.8**/72.9	96.0/81.2	96.1/79.8	88.0
XeroAlign$_{\text{IO}}$ (Ours)	95.3/81.3	98.5/75.1	96.4/82.3	96.3/81.1	88.7
CrossAligner (Ours)	94.4/81.6	95.3/78.8	94.8/**84.1**	94.7/**82.5**	88.6
Contrastive (Ours)	95.3/80.9	98.3/**79.6**	96.5/79.3	96.3/79.8	88.1
XeroAlign$_{\text{IO}}$ + CrossAligner (1+1)	95.3/81.5	98.6/78.2	96.2/81.6	96.2/81.1	88.7
XeroAlign$_{\text{IO}}$ + CrossAligner (CoV)	95.4/**82.2**	**98.8**/78.3	**96.6**/83.1	**96.5**/82.1	**89.3**

Table 1: Accuracy/F-Score for MTOP, MTOD, M-ATIS (number of non-English languages in brackets), MEAN over all datasets. Translate-Train SOTA is (Li et al., 2021b) for MTOP/MTOD and (Xu et al., 2020) for M-ATIS.

Multilingual Task-Oriented Parsing (MTOP) comprises 15K-22K utterances in each of 6 languages (en, de, fr, es, hi, th) spanning 11 domains (Li et al., 2021b). The **Multilingual Task-Oriented Dialogue** (MTOD) consists of 43K English, 8K Spanish and 5K Thai utterances covering 3 domains (Schuster et al., 2019). The **Multilingual ATIS++** (M-ATIS) contains up to 4.5K commands in each of 8 languages (en, es, pt, de, fr, zh, ja, hi, tr) featuring user interactions with a travel information system (Xu et al., 2020).

XLM Our pretrained language model of choice is XLM-RoBERTa (Conneau et al., 2020). We use the large (550M parameters) model from HuggingFace (Wolf et al., 2019) with a hidden_size = 1,024.

Training Setup We use a minimalist setup that features default settings and components to focus the results on the methods rather than hyperparameter tuning or custom architecture design. We implemented all models with PyTorch using fixed hyperparameters between experiments except for MTOD, where due to its size, we trained with fewer epochs and a lower learning rate (both 50% lower[2]).

5 Results

Terminology Henceforth, we refer to models trained with labelled data in each language as **Target Language**, the models trained only on English data as **Zero-Shot**, our translate-intent method as **Translate-Intent** (**TI**), the scores reported by Gritta and Iacobacci (2021) as **Previous SOTA**, our IO-only implementation of that model as **XeroAlign$_{\text{IO}}$** (**XA$_{\text{IO}}$**), our contrastive alignment method as **Contrastive** (**CTR**) and our main method as **CrossAligner** (**CA**). Lastly, the simple sum of alignment losses is referred to as **1+1** and the weighted sum from 3.4 as **CoV**.

Metrics We use Accuracy for intent classification and F-Score for entity/slot recognition. In addition, we use an Overall score (the average of F-Score and Accuracy) for model ranking, similar to Hu et al. (2020); Wang et al. (2019, 2018). Results are shown as averages (MEAN) over all test sets and datasets, presented in Tables 1 and 2. Intent classification is thus evaluated on ∼20,000 diverse user commands and entity recognition on ∼60,000 individual slots from 100+ slot types. For a full breakdown, see Tables 4, 5 and 6 in Appendix A.2.

Statistical Significance For a robust comparison with the previous SOTA, we conduct a two-tailed z-test for the difference of proportions (Schumacker, 2017). Our most effective method is statistically significant for all datasets at $p < 0.01$. The margin of improvement for slot tagging is +2.3 (F-Score) over previous SOTA and significant at $p < 0.01$.

5.1 Individual Zero-Shot Transfer Methods

CrossAligner The focus of our primary method was to improve slot filling as the model must classify dozens of entity types in each dataset and to that end, it is an effective approach. CrossAligner exceeds the F-Score of the Previous SOTA by 2.7 points (82.5 versus 79.8). This is 1.4 points higher than XeroAlign$_{\text{IO}}$ and 6 points higher than Zero-Shot. Despite the intent accuracy being 1.4 points lower than Previous SOTA and 1.6 lower

[2]Download code and data at https://github.com/huawei-noah/noah-research

Setup	Auxiliary Losses				CoV Weighting			1+1 Weighting		
	CA	XA_{IO}	CTR	TI	MEAN (15)		Overall	MEAN (15)		Overall
2-Loss	x	x			**96.5**	**82.1**	**89.3**	96.2	81.1	88.7
		x	x		95.9	80.1	88.0	96.1	80.1	88.1
	x		x		96.2	81.3	88.8	96.1	78.2	87.2
	x			x	96.2	81.3	88.8	96.2	79.2	87.7
		x		x	96.2	80.3	88.3	96.3	80.2	88.3
			x	x	96.1	79.6	87.9	96.2	79.7	88.0
3-Loss	x	x	x		96.4	81.4	88.9	96.3	80.1	88.2
	x	x		x	**96.5**	80.6	88.6	96.2	81.0	88.6
	x		x	x	96.3	81.2	88.8	96.3	79.0	87.7
		x	x	x	96.1	80.3	88.2	96.4	80.0	88.2
4-Loss	x	x	x	x	96.3	79.7	88.0	96.4	79.7	88.1

Table 2: The Accuracy and F-Score for combinations of auxiliary losses with different weighting schemes. The number of non-English test languages is shown in brackets, MEAN is computed for all languages in the 3 XNLU datasets. More detailed breakdowns of each dataset and language are shown in Tables 4, 5 and 6 in Appendix A.2.

than $XeroAlign_{IO}$, 94.7 is still 0.4 higher than Target Language. CrossAligner's overall score is 0.6 higher than previous SOTA, which outperformed the common 'translate-train' models, including entity projection and word alignment. In order to demonstrate the necessity and specificity of the proposed architecture, we tested mean-pooled token embeddings as well as a CLS embedding as the input to CrossAligner instead of the entity classifier logits. The scores declined from 94.7/82.5 (88.6 Overall) to 92.3/80 (86.2 Overall) with a CLS sentence representation and 82.1/78.7 (80.4 Overall) for mean-pooled embeddings. Future applications of our method to other NLP tasks must note that CrossAligner is most effective for tasks with a *complex entity tag set* where the presence of entities in a sentence is informative, i.e. a higher complexity and slot density should lead to a higher performance. In addition, CrossAligner combines well with other losses as we show in Section 5.2.

Translate-Intent Our alternative to the common 'translate-train' baseline is not only conceptually simpler (no explicit slot recognition training), it also outperforms the previous Translate-Train SOTA scores (78.5 vs 76.6 F-Score, 95.9 vs 95.1 accuracy and 87.2 vs 85.9 Overall). Translate-Intent does not require error-prone preprocessing such as word/label alignment and can therefore be readily used as a default 'translate-train' baseline in future work. Note that using mean-pooled token embeddings as sentence representations is not recommended for Translate-Intent as this causes the F-Score to decline sharply (-25 points).

Contrastive Alignment Despite orders of magnitude less data than used in related work (Section 2), our Contrastive Alignment showed a marginal improvement over the previous SOTA on intent classification (96.3 vs 96.1) thus by 0.1 Overall. That said, even though the contrastive loss pushes negative sentence embeddings away from the positives, this does not seem to confer a strong advantage over the previous SOTA, which only used positive examples. We have also evaluated an implementation of Contrastive Alignment using mean-pooled token embeddings as sentence representations, however, the Overall score declined to 86.8 (versus 88.1 with a standard CLS embedding).

$XeroAlign_{IO}$ Our implementation of the previous SOTA with an additional post-processing step (described in 3.1) increased the F-Score by 1.3 points and accuracy by 0.2 (+0.7 Overall). For a comparison, training $XeroAlign_{IO}$ with the conven-

tional BIO tags results in a drop of 1.8 points (81.1 to 79.3 F-Score) on entity recognition and 0.4 on intent classification (96.1 to 95.7). Mean-pooled tokens are not recommended for XeroAlign$_{IO}$ as this yields a 2-point decline (88.7 to 86.7 Overall). Other models also benefit from IO-only training, for example, the Zero-Shot model gains 2.6 points (73.9 up to 76.5 F-Score). One theoretical limitation of IO-only training is that given a sequence of 'B-LOC I-LOC B-LOC', the IO-only models would incorrectly classify this as a single entity. However, in practice, this is rare and not something we have seen during preprocessing or error analysis.

5.2 Combinations of Losses

As our alignment methods have different strengths and weaknesses, we have also evaluated their combinations (see Table 2) as either a simple sum of losses (**1+1**) or a weighted sum of losses (**CoV**) using the Coefficient of Variation. The highest overall score was achieved by a CoV-weighted combination of XeroAlign$_{IO}$ and CrossAligner, which considerably improved on the previous SOTA (96.5 vs 96.1 Accuracy, 82.1 vs 79.8 F-Score, 89.3 vs 88.0 Overall). In total, three individual and almost a dozen combinations of losses improve over the best previously reported scores. In the following paragraphs, we analyse and explain why the combinations that include CrossAligner consistently produce higher scores and why adding more losses can result in diminishing returns.

Compatibility of Losses We propose a hypothesis that can further help us interpret the numbers in Table 2. It states that combining losses which use dissimilar sentence representations may be more beneficial than combining losses using similar sentence embeddings. In order to test that assumption, we clustered our alignment methods into two groups based on how their sentence representations are obtained: 1) XeroAlign$_{IO}$, Translate-Intent and Contrastive Alignment, which all use the CLS *embedding* and 2) CrossAligner, which aligns through the *token embeddings* (used as the entity classifier input). In Figure 2, we note that for combinations of any two alignment losses using the CLS embedding (shown as blue squares), there is no difference in the overall scores when using CoV or 1+1. However, when combining losses with different sentence representations (orange with any blue square) using CoV weighting, we observe consistent increases over the 1+1 setup (on average 1+ point overall) as well as an increase over their highest individual score. Additionally, in a 3-loss combination, we note that adding CrossAligner to any two losses from the CLS-embedding group using CoV weighting yields an average improvement of 0.7 points compared to no improvement using 1+1.

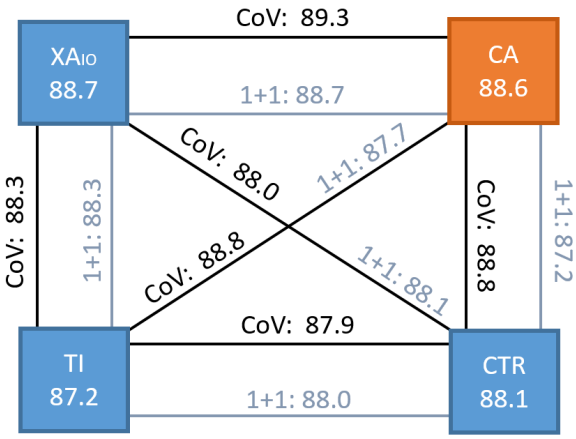

Figure 2: Overall scores for combinations of auxiliary losses weighted using either CoV or 1+1 (simple sum).

Oversaturation of Losses Another important observation harks back to our hypothesis stating that alignment methods with similar input embeddings do not lend themselves to being readily combined. We offer further evidence of this by testing combinations of CrossAligner with each of the CLS-embedding losses [XA$_{IO}$, TI and CTR], however, we use mean-pooled embeddings. The Overall scores decline in line with our hypothesis (XA$_{IO}$ by -1.2, TI by -4.9, CTR by -0.6) with CoV-weighted losses and even more with the 1+1 weighted combinations (XA$_{IO}$ by -2.1, TI by -7.6, CTR by -1.4). Similarly, combining multiple CLS-embedding losses leads to a gradually diminishing benefit relative to the individual scores. Once again, the CoV-weighted losses show a significantly lower decline than the 1+1 combinations (Table 2). Note that in our multi-loss scenario, intent classification remains unaffected by the choice of input embeddings as the accuracy remains stable at SOTA levels across experiments. We think this is due to the unequal task difficulty. In other words, sentence-level inference (intent recognition) is easier than token-level inference (entity recognition).

6 Error Analysis

In order to contextualise the numbers reported in Tables 1 and 2 in relevant linguistic insights, we have conducted a qualitative error analysis and

Categories	TH	HI	FR	DE	ZH	ES	PT	MEAN
Acceptable Transfer	28	51	53	40	37	38	23	38.6
Partial Transfer	34	15	13	30	34	47	58	33.0
Poor Transfer	38	34	34	30	29	15	19	28.4
Boundary Error	72	43	44	52	33	47	64	50.7
Semantics Error	38	40	37	38	59	30	32	39.1
Annotation Error	8	26	11	20	30	17	17	18.4

Table 3: The summary of our qualitative error analysis with native speakers (700 samples from 7 languages).

present the highlights in this section. Readers interested in language-specific analysis (including many more examples) are encouraged to read **Appendix A.1**. We focused on errors committed by CrossAligner and XeroAlign$_{\text{IO}}$, which achieved the best individual and combined scores. We sampled 100 random errors from each of the following settings: Hindi, French and German from MTOP, Portuguese, Chinese and Spanish from M-ATIS and Thai from MTOD for a diverse pool of errors. The authors adjudicated with native speakers to categorise mistakes into the following types.

Error Types We discovered two main sources of mistakes: A *boundary error* occurs when the model predicts more or fewer entity words/tokens than given in the gold annotation. A *semantics error* occurs when the wrong entity class/type is predicted. Models can therefore commit: 1) both errors resulting in *Poor Transfer*, 2) a boundary error without a semantic error and vice versa giving us a *Partial Transfer* or 3) neither error (a false negative), which we deemed an *Acceptable Transfer*. We report individual and average error occurrences as well as transfer type percentages in Table 3.

Poor Transfer indicates that the prediction error is too serious and unusable (even misleading) in a real-world personal assistant. This is typically due to both a boundary and a semantics error, however, some mistakes can be serious enough alone to result in poor transfer. For example, a boundary error can cause the retrieved name of a dish, person or a location to be incomplete and therefore invalid. A semantics error that classifies '10 secondes' (French) as 'date_time' instead of 'music_rewind_time' would elicit the wrong agent response thus is unusable. On average, ∼28% of mistakes fall into the 'Poor Transfer' category.

Partial Transfer is defined as either a boundary or a semantics error where neither is considered a serious problem. Such entities could be made usable in a personal assistant application with simple post-processing rules. Around 33 percent of errors were deemed to be partially correct. Often, this was due to including some adjacent punctuation or an article/preposition as part of the entity or a slightly shorter/longer news headline even though a search engine query with that string would have returned the relevant article. Entities such as '24 minat ka' (Hindi) versus '24 minat' (24 minutes) exemplify the fact that a disputed entity boundary is the most frequent source of error in this category. On the semantics side, we considered a location partially correct if 'state_name' instead of 'city_name' (for Washington D.C.) was predicted, a location was expected and the boundary was accurate.

Acceptable Transfer These examples are 'errors' we considered correct and usable 'as is' because neither the entity boundary nor its semantics were thought to be wrong. On average, we deemed almost 39% of entities acceptable for a real-world personal assistant application with around half of those being down to *annotation problems* (labels missing or incorrect). In other cases, we accepted predictions that offered a valid alternative e.g. when both 'me' (French) and 'je' (I/me) are present in the user utterance and both refer to the same 'person_reminded'. Valid alternatives were predicted but annotated somewhat differently. For example, when the entity boundary was slightly wider 'de ida e volta' (Portuguese) instead of 'ida e volta' (round trip) where both entities are correct. Similarly, classifying 'salmon' as an ingredient rather than a dish (when 'salmon' is an object of 'prepare') was considered an acceptable transfer.

6.1 Error Analysis Summary

While the intent classification task is transferred well in a cross-lingual setting, performing better than training on labelled data, our SOTA slot recog-

nition F-Score is almost 7 points behind Target Language. We think there are several factors involved. Articles, some prepositions, conjunctions, determiners and/or possessives do not transfer easily and may largely be ignored by the XLM as they don't carry important sentence level (e.g. intent) semantics. English is not ideal as a cross-lingual pivot for many of the dozens of languages covered by the XLM as elements of culture and vernacular that may not have a direct English equivalent don't transfer easily in a zero-shot setting. Aligning on the most well-resourced language in the same family should help (Xia et al., 2021). The limits of machine translation, especially for low-resource languages (Mager et al., 2021), can further inhibit alignment methods that leverage parallel data. Inconsistency of annotation (intra-language and inter-language) is a source of errors when the key concepts are learnt in one language and evaluated (sometimes unreliably) in the target language. Finally, there were no substantial qualitative differences between XeroAlign$_{\text{IO}}$ and CrossAligner in our error analysis suggesting that the aforementioned error patterns may be a feature of the XLM model itself, the nature of the datasets or some as yet unknown confounding variable rather than the choice of the alignment method.

7 Conclusions

We have introduced a variety of cross-lingual methods for task-oriented XNLU to enable effective zero-shot transfer by learning alignment with unlabelled parallel data. The principal method, *CrossAligner*, transforms English train data into a new language-agnostic task used to align model predictions across languages, achieving SOTA on entity recognition. We then presented a *Contrastive Alignment* that optimises for a small cosine distance between translated sentences while increasing it between unrelated sentences, using orders of magnitude less data than previous works. We proposed *Translate-Intent*, a fast and simple baseline that beats previous Translate-Train SOTA approaches without error-prone data transformations such as slot label projection. The best overall performance across nine languages, fifteen tests sets and three task-oriented multilingual datasets was achieved by a *Coefficient of Variation* weighted combination of CrossAligner and XeroAlign$_{\text{IO}}$. Our quantitative analysis investigated which types of auxiliary losses yield the most effective combinations. This resulted in several proposed configurations also exceeding previous SOTA scores. Our detailed qualitative error analysis revealed that the best methods have the potential to approach target language performance as most errors were deemed to be of low to medium severity. We hope our contributions and resources will inspire exciting future work in this fascinating NLP research area.

Acknowledgements

We want to thank Philip John Gorinski, Guchun Zhang, Sushmit Bhattacharjee and Nicholas Aussel for providing the native speaker expertise for our qualitative error analysis. We are grateful to the ARR reviewers for their insightful comments and feedback. We also want to thank the MindSpore[3][4] team members for the technical support.

References

Naveen Arivazhagan, Ankur Bapna, Orhan Firat, Roee Aharoni, Melvin Johnson, and Wolfgang Macherey. 2019. The missing ingredient in zero-shot neural machine translation. *arXiv preprint arXiv:1903.07091*.

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2020. On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4623–4637.

Mikel Artetxe and Holger Schwenk. 2019. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610.

Inigo Casanueva, Paweł Budzianowski, Pei-Hao Su, Nikola Mrkšić, Tsung-Hsien Wen, Stefan Ultes, Lina Rojas-Barahona, Steve Young, and Milica Gašić. 2017. A benchmarking environment for reinforcement learning based task oriented dialogue management. *arXiv preprint arXiv:1711.11023*.

Aditi Chaudhary, Karthik Raman, Krishna Srinivasan, and Jiecao Chen. 2020. Dict-mlm: Improved multilingual pre-training using bilingual dictionaries. *arXiv preprint arXiv:2010.12566*.

Qian Chen, Zhu Zhuo, and Wen Wang. 2019. Bert for joint intent classification and slot filling. *arXiv preprint arXiv:1902.10909*.

Zewen Chi, Li Dong, Furu Wei, Nan Yang, Saksham Singhal, Wenhui Wang, Xia Song, Xian-Ling Mao, Heyan Huang, and Ming Zhou. 2020. Infoxlm: An information-theoretic framework for

[3] https://github.com/mindspore-ai

[4] https://mindspore.cn/

cross-lingual language model pre-training. *arXiv preprint arXiv:2007.07834*.

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.

Chris Dyer, Victor Chahuneau, and Noah A Smith. 2013. A simple, fast, and effective reparameterization of ibm model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648.

Milan Gritta and Ignacio Iacobacci. 2021. XeroAlign: Zero-shot cross-lingual transformer alignment. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 371–381, Online. Association for Computational Linguistics.

Milan Gritta, Gerasimos Lampouras, and Ignacio Iacobacci. 2021. Conversation graph: Data augmentation, training, and evaluation for non-deterministic dialogue management. *Transactions of the Association for Computational Linguistics*, 9:36–52.

Rick Groenendijk, Sezer Karaoglu, Theo Gevers, and Thomas Mensink. 2021. Multi-loss weighting with coefficient of variations. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1469–1478.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation. In *International Conference on Machine Learning*, pages 4411–4421. PMLR.

Alankar Jain, Bhargavi Paranjape, and Zachary C. Lipton. 2019. Entity projection via machine translation for cross-lingual NER. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1083–1092, Hong Kong, China. Association for Computational Linguistics.

Garry Kuwanto, Afra Feyza Akyürek, Isidora Chara Tourni, Siyang Li, and Derry Wijaya. 2021. Low-resource machine translation for low-resource languages: Leveraging comparable data, code-switching and compute resources. *arXiv preprint arXiv:2103.13272*.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. *arXiv preprint arXiv:1901.07291*.

Bing Li, Yujie He, and Wenjin Xu. 2021a. Cross-lingual named entity recognition using parallel corpus: A new approach using xlm-roberta alignment. *arXiv preprint arXiv:2101.11112*.

Haoran Li, Abhinav Arora, Shuohui Chen, Anchit Gupta, Sonal Gupta, and Yashar Mehdad. 2021b. MTOP: A comprehensive multilingual task-oriented semantic parsing benchmark. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2950–2962, Online. Association for Computational Linguistics.

Yaobo Liang, Nan Duan, Yeyun Gong, Ning Wu, Fenfei Guo, Weizhen Qi, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, et al. 2020. Xglue: A new benchmark datasetfor cross-lingual pre-training, understanding and generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6008–6018.

Manuel Mager, Arturo Oncevay, Abteen Ebrahimi, John Ortega, Annette Rios, Angela Fan, Ximena Gutierrez-Vasques, Luis Chiruzzo, Gustavo Giménez-Lugo, Ricardo Ramos, et al. 2021. Findings of the americasnlp 2021 shared task on open machine translation for indigenous languages of the americas. Association for Computational Linguistics.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Lin Pan, Chung-Wei Hang, Haode Qi, Abhishek Shah, Mo Yu, and Saloni Potdar. 2020. Multilingual bert post-pretraining alignment. *arXiv preprint arXiv:2010.12547*.

Kunxun Qi and Jianfeng Du. 2020. Translation-based matching adversarial network for cross-lingual natural language inference. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8632–8639.

Libo Qin, Minheng Ni, Yue Zhang, and Wanxiang Che. 2020. Cosda-ml: Multi-lingual code-switching data augmentation for zero-shot cross-lingual nlp. *arXiv preprint arXiv:2006.06402*.

Evgeniia Razumovskaia, Goran Glavaš, Olga Majewska, Anna Korhonen, and Ivan Vulić. 2021. Crossing the conversational chasm: A primer on multilingual task-oriented dialogue systems. *arXiv preprint arXiv:2104.08570*.

Erik Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

R Schumacker. 2017. Z test for differences in proportions. *Learning statistics using R. SAGE Publications*.

Sebastian Schuster, Sonal Gupta, Rushin Shah, and Mike Lewis. 2019. Cross-lingual transfer learning for multilingual task oriented dialog. 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 3795–3805.

Aditya Siddhant, Melvin Johnson, Henry Tsai, Naveen Ari, Jason Riesa, Ankur Bapna, Orhan Firat, and Karthik Raman. 2020. Evaluating the cross-lingual effectiveness of massively multilingual neural machine translation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8854–8861.

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

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, pages arXiv–1910.

Chuhan Wu, Fangzhao Wu, and Yongfeng Huang. 2021. Rethinking infonce: How many negative samples do you need? *arXiv preprint arXiv:2105.13003*.

Mengzhou Xia, Guoqing Zheng, Subhabrata Mukherjee, Milad Shokouhi, Graham Neubig, and Ahmed Hassan Awadallah. 2021. Metaxl: Meta representation transformation for low-resource cross-lingual learning. *arXiv preprint arXiv:2104.07908*.

Weijia Xu, Batool Haider, and Saab Mansour. 2020. End-to-end slot alignment and recognition for cross-lingual nlu. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5052–5063.

Huixiong Yi and Jin Cheng. 2021. Zero-shot entity recognition via multi-source projection and unlabeled data. In *IOP Conference Series: Earth and Environmental Science*, volume 693, page 012084. IOP Publishing.

A Appendix

A.1 Error Analysis per Language

German examples of partial transfer are boundary errors such as tagging punctuation '-' as part of the entity (e.g. in a timer or alarm name) as well as *not* tagging some punctuation e.g. in '14. Mai' where '.' is equivalent to the English 'th' in '14th' and is expected in German dates. Such entities can be used with a basic post-processing rule as their classes and contents were sufficiently well predicted. Similarly, including 'die', 'den', 'das', 'mir', 'des', 'der' in the retrieved entity, particularly in *free-format entities* such as news headlines, text message contents and memos need not invalidate the prediction, e.g. '*die* hausschliessung' (house closure), '*des* Reiseverbots' (*of* travel ban) and '*den* Termin' (appointment). Just a few of such linguistic 'bad habits' can quickly accumulate to cause more than half of all errors.

Chinese cross-lingual transfer problems often include boundary issues featuring the 'of' preposition or the possessive ''s' (*de* in Chinese) e.g. 'Zuì piányí *de*' (cheapest), 'Jiāzhōu *de*' (California) or 'āndàlüè *de* hángbān' (flights to Ontario). Depending on context, we considered these at least partially correct rather than a failed transfer. More serious though less explicable errors were 'Gěi wǒ zhōu'èr' (*give me* Tuesday's), 'Liè chū zhōu liù' (*list* Saturday's) or 'Xiǎnshì zhōusān' (*show me* Wednesday's) where 'give me', 'list' and 'show me' were tagged as part of 'date_time' a total of 20 times. The most frequent semantic (partial) error was 'Washington D.C.', which was tagged as a state rather than a city no fewer than 16 times.

French instances of acceptable transfer include tagging 'Ankara' and 'Turquie' separately rather than as a single chunk 'Ankara, en Turquie' (possible annotation problem). Reminiscent of the patterns seen in other languages, articles tend to feature prominently in boundary errors, e.g. '*la* famille' (family), '*l'*arrosage' (watering), '*les* elections' (elections), '*le* chat Zoom' (Zoom chat) and '*la* mere de Kylie' (Kylie's mother), which we considered usable 'as is'. For an example of annotation inconsistency across languages, consider the entity 'gros titres' or 'top headlines' in English. The model correctly transferred the English tags for 'top' (news_reference) and 'headline' (news_type) although the French annotation was given as 'gros

titres' (news_type), which is plausible but less coherent than the model's prediction.

Spanish Once again, articles, some prepositions and occasionally conjunctions e.g. 'de', 'las', 'y', 'la', 'por' (of, the, and, a, in/for) have caused the majority of boundary errors, most of which are partially acceptable. Examples include time '*las* 10 a.m.' (10 a.m.) and 'no mas tarde que *las*' (no later than), journey specifications '*de* ida' and 'ida *y* vuelta' (round trip), periods of day '*la* mañana' (morning) and '*la* noche' (night), dates 'seis *de* junio' (June 6th) as well as skipping '*en punto*' (o'clock) in 'antes de las 4 p.m. *en punto*' (before 4 p.m.). These minor errors show that the current SOTA in cross-lingual zero-shot transfer is close to solving these cases. Other errors such as 'conexión' instead of '*con* conexión' (*with* connection) and '*la* mañana' rather than '*por la* mañana' (*in* the morning) are more examples of disputed annotations.

Portuguese predictions closely follow Spanish error patterns and reflect the wider issues with articles and prepositions, e.g. 'terça-feira *de* manhã' (Tuesday morning), '*de* ida e volta' (round trip), 'cinco *de* abril' (April 5th) or '5 horas *da* tarde' (5 p.m.) with '*de*' and '*da*' (both *of*) being the unannotated parts that did not transfer optimally. Other boundary mistakes were caused by '*das*' (of) and '*as*' (the), for example, 'antes *das* 6 horas da tarde' and 'após *as* 6 horas da tarde' (before and after 6 p.m). Annotations that needlessly punished cross-lingual transfer included '*somente* de ida' (one way) where '*somente*' (only) was not annotated in the English dataset and 'econômica' (economy), which was annotated as 'class_type' in English, correctly transferred but flagged as wrong.

Hindi errors have a relatively high number (26) of problematic annotations although most mistakes are caused by the now familiar improper handling of prepositions, articles and/or possessives e.g. '*ke*', '*tak*', '*ka*' ('of', 'by', ''s') in phrases such as '30 minat *ka*' (30 minutes), 'kitanee der *tak*' (how long), 'kal *ke*' (yesterday's), 'aaj *ke*' (today's) or '1 baje *ka*' (1 p.m.). Transliterated entities i.e. English pronunciation written in Devanagari, is the second largest category of transfer problems in Hindi, e.g. 'pakrino romaano' (Pecorino Romano), 'goda cheez' (Goda cheese), 'braun aaid garl' (Brown Eyed Girl), 'paindora' (Pandora), 'pool leeg' (Pool League), 'daayanaasor jooniyar' (Dinosaur Junior) or 'da most byooteephul moment' (The Most Beatiful Moment). These are problematic because such entities are neither native to Hindi nor are they written in Latin alphabet hence may not have been observed in this form during XLM pretraining.

Thai errors were analysed with a translation service as we were unable to secure a native speaker. Even so, we observed boundary errors previously seen in other languages. Words such as '*nai*' ('of' or 'in', the most frequent cause) and '*bpai*' ('in', 'off' or 'to', no direct English translation) were the typical sources of boundary issues, e.g. '*nai* sùt sàp-daa née' (this weekend), '*nai* wan pút' (Wednesday), '*bpai* séu XYZ' (go buy XYZ) and '*nai* wan née' (today or on this day). Such patterns accounted for more than half of all mistakes. Machine translation can also be a source of errors. For example, the word 'reminder' is an entity in English (tag: reminder/noun). It was translated as 'kam dteuuan', however, 'reminder' appears in Thai data as 'dteuuan kwaam jam', which the model repeatedly missed, leading to 18 errors for what should be an easy case of zero-shot transfer.

A.2 Full Tables

The full language breakdown for MultiATIS++ (Table 4) and MTOP+MTOD (Table 5). Table 6 shows the full details of the combinations of losses from Table 2 in Results (5).

Model	DE	ES	FR	TR	HI	ZH	PT	JA	MEAN
Zero-Shot	95.1/84.8	97.3/84.9	97.9/79.5	75.4/41.8	91.3/78.4	88.6/82.1	96.9/80.9	86.6/79.9	91.1/79.9
Target Language	96.9/95.4	96.6/85.8	97.9/93.8	77.2/71.6	88.8/84.4	94.5/94.9	96.8/92.1	91.4/93.0	92.5/88.9
Trans-Train SOTA	96.7/89.0	97.2/76.4	97.5/79.6	93.7/61.7	92.8/78.6	96.0/83.3	96.8/76.3	88.3/79.1	94.9/78.0
Translate-Intent	97.3/84.7	97.6/84.1	97.5/84.7	91.6/65.6	94.4/80.9	96.4/83.0	97.0/82.7	95.2/74.1	95.9/80.0
Previous SOTA	**97.6**/84.9	**97.8**/85.9	95.4/81.4	93.4/70.6	94.0/79.7	96.4/83.3	97.6/79.9	96.1/83.5	96.0/81.2
XeroAlign$_{IO}$	97.4/84.1	97.4/**86.2**	97.9/83.3	93.6/76.0	**95.1**/80.1	96.0/83.7	**98.0**/81.6	95.6/83.7	96.4/82.3
CrossAligner	96.9/**90.4**	97.4/73.1	**98.0**/88.7	88.7/75.4	94.5/**86.6**	94.2/88.2	97.4/**81.7**	91.6/88.7	94.8/**84.1**
Contrastive	97.5/79.6	97.3/77.1	97.6/76.1	93.3/**76.3**	94.6/79.8	**96.9**/87.5	97.4/77.0	**97.3**/80.9	96.5/79.3
XA$_{IO}$, CA (1+1)	97.3/89.7	97.5/72.3	97.8/82.1	**94.0**/69.1	95.3/79.1	95.9/89.6	97.4/80.1	94.0/**91.1**	96.2/81.6
XA$_{IO}$, CA (CoV)	**97.6**/88.7	97.6/72.5	**98.0/88.8**	93.3/73.9	**95.1**/79.7	96.5/**89.9**	97.5/80.9	97.0/90.5	**96.6**/83.1

Table 4: Accuracy/F-Score for M-ATIS. Xu et al. (2020) is the previous translate-train SOTA.

Model	DE	ES	FR	TH	HI	MEAN	ES	TH	MEAN
Zero-Shot	90.5/80.1	93.8/81.7	92.5/83.2	89.7/64.9	91.8/75.5	91.7/77.1	97.5/87.3	90.6/62.8	94.1/75.1
Target Language	96.5/88.7	96.1/90.6	95.6/89.3	95.0/87.0	95.1/87.8	95.7/88.7	98.9/89.3	97.8/94.3	98.4/91.8
Trans-Train SOTA	94.8/80.0	96.3/84.8	95.1/82.5	92.1/65.6	94.2/76.5	94.5/77.9	98.0/83.0	96.9/52.8	97.5/67.9
Transl.Intent	96.4/83.4	96.2/73.5	95.5/84.5	92.9/68.4	94.9/75.6	95.2/77.1	99.2/87.5	96.9/65.4	98.1/76.5
Previous SOTA	**96.6**/84.4	96.5/83.3	95.7/84.5	**94.1**/69.1	95.2/80.1	**95.6**/80.3	99.2/88.4	**98.4**/57.3	**98.8**/72.9
XeroAlign$_{IO}$	96.4/86.1	96.4/**84.4**	95.4/**86.2**	93.1/69.5	95.1/80.5	95.3/81.3	**99.3**/88.8	97.6/62.0	98.5/75.4
CrossAligner	95.4/86.0	95.1/81.9	94.5/84.9	92.6/73.9	94.3/**81.1**	94.4/81.6	98.2/87.1	92.4/**70.4**	95.3/78.8
Contrastive	96.3/84.3	96.2/83.3	95.5/85.2	93.0/70.6	**95.5**/80.9	95.3/80.9	99.2/89.0	97.3/70.1	98.3/**79.6**
XA$_{IO}$, CA (1+1)	96.3/85.3	**96.6**/83.1	**95.9**/85.9	93.3/72.8	94.5/80.4	95.3/81.5	99.2/88.7	98.0/67.7	98.6/78.2
XA$_{IO}$, CA (CoV)	96.4/**86.6**	96.1/83.8	95.8/85.7	93.3/**74.3**	95.2/80.4	95.4/**82.2**	**99.3/89.3**	98.2/67.2	**98.8**/78.3

Table 5: Accuracy/F-Score for MTOP (left) and MTOD (right). Li et al. (2021b) is previous translate-train SOTA.

Setup	Auxiliary Losses				Weight.	MTOP(5)		MTOD(2)		M-ATIS(8)		MEAN(15)		Overall
	CA	XA_{IO}	CTR	TI										
2-Loss	x	x			CoV	95.4	82.2	**98.8**	78.3	**96.6**	**83.1**	**96.5**	**82.1**	**89.3**
	x	x			1+1	95.3	81.5	98.6	78.2	96.2	81.6	96.2	81.1	88.7
		x	x		CoV	95.1	80.9	97.3	73.5	96.0	81.3	95.9	80.1	88.0
		x	x		1+1	95.2	79.1	97.1	77.1	96.3	81.5	96.1	80.1	88.1
	x		x		CoV	95.2	**82.3**	98.6	77.6	96.3	81.6	96.2	81.3	88.8
	x		x		1+1	95.2	82.2	98.6	78.3	96.1	75.7	96.1	78.2	87.2
	x			x	CoV	95.3	81.5	**98.8**	78.0	96.1	82.0	96.2	81.3	88.8
	x			x	1+1	**95.5**	81.3	98.6	77.4	96.1	78.3	96.2	79.2	87.7
		x		x	CoV	95.2	80.6	98.5	78.2	96.2	80.6	96.2	80.3	88.3
		x		x	1+1	95.3	79.3	**98.8**	78.1	96.4	81.3	96.3	80.2	88.3
			x	x	CoV	95.2	79.8	97.8	76.8	96.2	80.3	96.1	79.6	87.9
			x	x	1+1	95.3	79.9	98.6	79.2	96.1	79.7	96.2	79.7	88.0
3-Loss	x	x	x		CoV	95.3	82.1	98.6	78.7	**96.6**	81.7	96.4	81.4	88.9
	x	x	x		1+1	95.3	81.8	**98.8**	79.4	96.4	79.3	96.3	80.1	88.2
	x	x		x	CoV	95.4	81.4	98.7	78.4	**96.6**	80.6	**96.5**	80.6	88.6
	x	x		x	1+1	95.4	81.0	98.0	78.9	96.4	81.5	96.2	81.0	88.6
	x		x	x	CoV	95.1	81.5	**98.8**	79.2	96.5	81.4	96.3	81.2	88.8
	x		x	x	1+1	**95.5**	80.1	98.7	78.5	96.1	78.5	96.3	79.0	87.7
		x	x	x	CoV	95.2	80.2	97.7	78.4	96.2	80.8	96.1	80.3	88.2
		x	x	x	1+1	95.2	80.0	98.5	78.6	**96.6**	80.4	96.4	80.0	88.2
4-Loss	x	x	x	x	CoV	95.3	81.6	98.7	**79.7**	96.3	78.6	96.3	79.7	88.0
	x	x	x	x	1+1	95.4	81.1	98.4	79.1	**96.6**	78.9	96.4	79.7	88.1

Table 6: Accuracy and F-Score for combinations of auxiliary losses with different weighting schemes.