Exploring X-GEAR for Zero-Shot Cross-Lingual Event Argument Extraction

Ramaswamy Iyappan
MS Computer Science
George Mason University
riyappan@gmu.edu

Abhijeet Banerjee
MS Computer Science
George Mason University
abanerj@gmu.edu

Bhargava Canakapally
MS Computer Science
George Mason University
bcanakap@gmu.edu

1 Introduction

1.1 Task / Research Question Description

The objective of this paper is to introduce a study that uses pre-trained generative language models to achieve zero-shot cross-lingual event argument extraction (EAE). The method proposed in this paper formulates EAE as a language generation task that effectively encodes event structures and captures argument dependencies. This paper proposes the use of language-agnostic templates that can represent event argument structures in any language, making cross-lingual transfer possible. The proposed model is trained by finetuning multilingual pre-trained generative language models to generate sentences that fill in the language-agnostic templates with arguments extracted from the input text. Our motivation is to use multilingual pre-trained generative language models that are able to effectively encodes event structures and captures the dependencies between arguments.

1.2 Motivation and Limitations of existing work

The majority of the models discussed in the paper are based on classification techniques, where classifiers are built on top of pre-trained masked language models that can operate in multiple languages. However, some of these models require additional information, like bilingual dictionaries (Liu et al., 2019; Ni and Florian, 2019), translation pairs (Zou et al., 2018), and dependency parse trees (Subburathinam et al., 2019; Ahmad et al., 2021; Nguyen and Nguyen, 2021), to address differences between languages. Nevertheless, according to previous papers (Li et al.; Hsu et al., 2022), classification-based models are not as effective as generation-based models in capturing the relationships between entities.

Numerous other studies have proven the effectiveness of generation-based models in achieving high performance on monolingual structured prediction tasks. However, since the targets generated by these models are language-specific, their direct application to zero-shot cross-lingual tasks may result in suboptimal performance. Recently, there has been increased interest in using prompts to guide the behavior of pre-trained language models and elicit knowledge (Peng et al., 2019; Schick and Schutze, 2021). Despite this, there has been limited focus on multilingual tasks (Winata et al., 2021).

1.3 Proposed Approach

The proposed approach is to devise cross-lingual event argument extraction using in zero-shot setting as a language generation task. Along with this we propose, X-GEAR which stands for Cross-lingual Generative Event Argument extractoR(Huang et al., 2022). The X-GEAR model fine-tunes generative models that are pre-trained for multiple languages, such as mBART-50 (Tang et al., 2020) or mT5 (Xue et al., 2021). It also incorporates a copy mechanism to improve its ability to adapt to changes in input language. This paper provides a detailed explanation of X-GEAR's design, including the language-agnostic templates used, the desired output, the format of the input, and the specifics of its training process.

1.4 Summary of Experiments

We observed that our model shows significant improvement in the performance of identifying arguments and classifying roles in zero-shot cross-lingual settings. Additionally, experiments were carried out to test against real-world noisy data showing that the model was significantly robust towards Irrelevant Additions, Named Entity Recognition(NER), Negation and Semantic Role Labeling(SRL). Further experiments were carried out

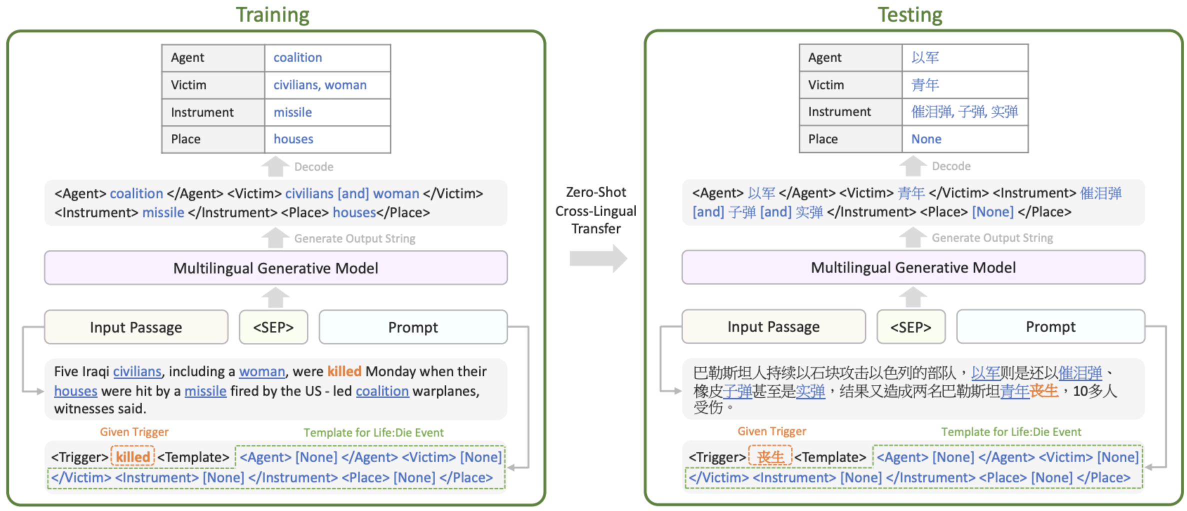
Training
Agent | coalition
Victim | civilians, woman
Instrument | missile
Place | houses
Decode
<Agent> coalition </Agent> <Victim> civilians [and] woman </Victim> <Instrument> missile </Instrument> <Place> houses</Place>
Generate Output String
Multilingual Generative Model
Input Passage
<SEP>
Prompt
Five Iraqi civilians, including a woman, were killed Monday when their houses were hit by a missile fired by the US - led coalition warplanes, witnesses said.
Given Trigger
Template for Life:Die Event
<Trigger> killed <Template> <Agent> [None] </Agent> <Victim> [None] </Victim> <Instrument> [None] </Instrument> <Place> [None] </Place>
Zero-Shot Cross-Lingual Transfer
Testing
Agent | 以军
Victim | 青年
Instrument | 催泪弹, 子弹, 实弹
Place | None
Decode
<Agent> 以军 </Agent> <Victim> 青年 </Victim> <Instrument> 催泪弹 [and] 子弹 [and] 实弹 </Instrument> <Place> [None] </Place>
Generate Output String
Multilingual Generative Model
Input Passage
<SEP>
Prompt
巴勒斯坦人持续以石块攻击以色列的部队，以军则是还以催泪弹、橡皮子弹甚至是实弹，结果又造成两名巴勒斯坦青年丧生，10多人受伤。
Given Trigger
Template for Life:Die Event
<Trigger> 丧生 <Template> <Agent> [None] </Agent> <Victim> [None] </Victim> <Instrument> [None] </Instrument> <Place> [None] </Place>

Figure 1

by augmenting existing sentences in the dataset into German, Spanish and French to analyze the model's multilingual capabilities.

2 Approach

We aim to present X-GEAR and compose zero-shot cross-lingual event argument extraction as a language generation task. X-GEAR stands for Cross-lingual Generative Event Argument extractor. Shown in Figure 1. In the figure we show a simplified representation of X-GEAR, which works the following way, passing an input passage along with a well-designed prompt that contains an event trigger and our proposed language-agnostic template, our model fills the template with event arguments. This formulation raises two main challenges first one being that the input language in the test set might be the one in the train set. The second challenge concerns the difficulty of parsing the output string into the final prediction. Therefore, the output string needs to be well-structured along with accommodating the intended changes.

To solve these challenges we introduce language-agnostic templates. Here's how it works, we pass an input passage and a prompt that contains the trigger and its event type along with our language-agnostic template, our model will extract information from the input passage and generate output strings to fill the template. With the use of this template, we can ensure that the output strings are in structured form and this also helps with cross-lingual transfer.

Our model fine-tunes multilingual pre-trained generative models like mBART-50 (Tang et al., 2020), or mT5 (Xue et al., 2021) and supplements them with a copy mechanism to accommodate input language changes.

2.1 Zero-Shot Cross-Lingual Event Argument Extraction

An Event Argument Extraction model detects arguments and their roles from a given input text and event trigger. Figure 1 represents an input passage x, which is of event type e *(Life:Die)* and has an event trigger t *(killed)*. An EAE model would predict a list of arguments a = $[a_1, a_2, .., a_l]$ *(coalition, civilians, women, missile, houses)* and their corresponding roles r = $[r_1, r_2, ..., r_l]$ *(Agent, Victim, Victim, Instrument, Place)* for this input. Since we approach a zero-shot cross-lingual setting, we consider $X_{train} = \{(x_i, t_i, e_i, a_i, r_i)\}_{i=1}^N$ as the training set that belongs to a source language and $X_{test} = \{(x_i, t_i, e_i, a_i, r_i)\}_{i=1}^M$ as the test set that belongs to a target language.

Just like monolingual EAE, zero-shot crosslingual EAE models are anticipated to comprehend the connections between arguments and generate organized predictions. Nevertheless, unlike monolingual EAE, zero-shot cross-lingual EAE models must address the dissimilarities (such as grammar and word order) among languages and acquire the ability to transfer knowledge from the source languages to the target languages.

2.2 Language-Agnostic Template

For each event type "e" we have one language-agnostic template "T_e" which in simple terms is a unique HTML-tag-style template. Taking the same example from the figure to demonstrate a template, the Life:Die event is associated with four roles: Agent, Victim, Instrument, and Place. Hence the template for Life:Die event will be:

```
<Agent>[None]</Agent><Victim>[None]</Victim>
<Instrument>[None]</Instrument>
<Place>[None]</Place >
```

For ease of understanding English words have been used to represent the above template. These tokens can be replaced with any other format, as the pre-trained model has never came across these representations which required the model to learn these representations from scratch anyways. As these are special tokens that are not pre-trained, they are considered as language-agnostic.

2.3 Target Output String

The X-GEAR learns to generate target output strings that follow the form of the language-agnostic template. Given the following, input passage, trigger, event type, argument prediction and role predictions, we then pick our language-agnostic template for the event type and replace all [None] with the corresponding arguments in argument prediction according to their role in role prediction. In some cases where there are multiple arguments for a role we use "[and]". To illustrate this, the training example in figure has two arguments (civilians and women) for the role of Victim, hence the output string would look like

```
<Victim>civilians [and] women </Victims>
```

And if there are no corresponding argument, we keep [None]. Accordingly the example in figure will look like

```
<Agent>coalition</Agent><Victim>civilians[and]
women</Victim><Instrument>missile</Instrument>
<Place>house </Place>
```

We can use a simple rule-base algorithm to decode the arguments and role predictions because the output string is in HTML-tag-style format.

2.4 Input Format

As previously stated, in order to create zero-shot cross-lingual EAE, it's important to guide the model to produce output in the desired format. To achieve this, both the input passage and a prompt are given to X-GEAR, as illustrated in Figure 1. The prompt includes valuable information for the model, such as a trigger and a language-agnostic template T_e, which implicitly contains the event type e information.

2.5 Training

In order to allow X-GEAR to produce sentences in various languages, we utilize a multilingual pre-trained generative model as our foundational model. This model calculates the likelihood of generating a new token based on the preceding tokens that were generated and the input context is given to the encoder c, i.e.

$$P(x|c) = \prod_{i} P_{gen}(x_i|x_{<i}, c),$$

where x_i is the output of the decoder at step i.

Copy Mechanism, While multilingual pre-trained generative models are capable of generating sequences in numerous languages, depending solely on them can lead to the creation of false arguments (Li et al.). Given that the majority of tokens in the target output sequence can be found in the input sequence, we add a Copy Mechanism to the multilingual pre-trained generative models to aid X-GEAR in adapting more effectively to the cross-lingual situation. To be more specific, we adopt the approach described by (See et al., 2017) to determine the probability of generating a token t. This probability is determined through a weighted combination of the vocabulary distribution calculated by the multilingual pre-trained generative model P_{gen} and the copy distribution P_{copy}.

$$P_{X-GEAR}(x_i = t|x_{<i}, c) =$$
$$w_{copy} \cdot P_{copy}(t) + (1 - w_{copy}) \cdot P_{gen}(x_i|x_{<i,c})$$

where $w_{copy} \in [0,\ 1]$ is the copy probability computed by passing the decoder hidden state at time step i to a linear layer. P_{copy} represents the probability of input tokens, weighted by the cross-attention computed by the last decoder layer at time step i. Our model is then trained end-to-end with the following loss:

$$\mathcal{L} = -log \sum_i P_{X-GEAR}(x_i|x_{<i}, c).$$

3 Experiments

3.1 Datasets

Initially, we planned on utilizing two popular and commonly used event argument extraction datasets: ACE-2005 and ERE. But since they are not freely available from LDC, we were unable to perform any pre-processing steps during checkpoint-1. However, we thank the authors (Huang et al., 2022) as in checkpoint-1, we ended up using their pre-processed train / dev / test splits of English (en), Arabic (ar) and Chinese (zh) annotations for ACE-2005.

Language	Sentence	Event	Argument
Train data			
en	17172	4202	4859
ar	2722	1743	2506
zh	6305	2926	5581
Dev data			
en	923	450	605
ar	289	117	174
zh	486	217	404
Test data			
en	832	403	576
ar	272	198	287
zh	482	190	336

Table 1: Dataset Statistics

Table 1 specifically shows more information about the splits. The English and Chinese splits are from (Wadden et al., 2019) and (Lin et al., 2020) respectively. The Arabic part is from the original authors (Huang et al., 2022).

After a few days of our baseline implementation, we managed to finally get the access to the original ACE-2005 from LDC. This will let us train and improve our model even better, and give us more understanding about it. At First, we consider English, Arabic, and Chinese annotations for ACE-2005 (Doddington et al., 2004) and adopted the pre-processing method outlined in (Wadden et al., 2019) to keep 33 event types and 22 argument roles.

It is important to note that prior research on the zero-shot cross-lingual transfer of event arguments has primarily focused on the role labeling of event arguments (Subburathinam et al., 2019; Ahmad et al., 2021), where they assume that ground truth entities are provided both during training and testing. Since they treat each event-entity pair as a separate instance, events in a sentence can be dispersed across all training, development, and test splits in their experimental data splits. In this work, we focus on event argument extraction (Wang et al., 2019; Wadden et al., 2019; Lin et al., 2020), which is a more realistic setting.

3.2 Evaluation Metric

We follow earlier research (Ahmad et al., 2021; Lin et al., 2020) and use the F1 score for argument classification to evaluate the effectiveness of models. If the role type and the argument offsets match the ground truth, the argument-role pair is considered accurate. The argument classification F1 score is defined as the F1 score between the sets $\{(a_i, r_i)\}$ and $\{(\tilde{a}_j, \tilde{r}_j)\}$ given the ground truth arguments a, ground truth roles r, predicted arguments ã, and predicted roles $\tilde{r}$. For every model, we experiment with three different random seeds and report the average results.

3.3 Compared Models

In this study, we compare the following models:

- **OneIE** (Lin et al., 2020), the leading technology for monolingual event extraction, is a classification-based model trained with multitasking that includes entity extraction, relation extraction, event extraction, and event argument extraction. To meet the zero-shot cross-lingual situation, we only swap out its pre-trained embedding with XLM-RoBERTa-large (Conneau et al., 2020). OneIE requires additional annotations, such as named entity annotations and relation annotations, as a result of the multi-task learning.

- **CL-GCN** (Subburathinam et al., 2019) is a classification-based model for cross-lingual event argument role labelling (EARL). In order to encode the parsing information, it uses GCN layers and dependency parsing annotations to bridge many languages (Kipf and Welling, 2017). We build two GCN layers on top of XLM-RoBERTa-large by following the implementation of previous work (Ahmad et al., 2021). We add one name entity recognition module that was jointly trained with CL-GCN because CL-GCN focuses on EARL tasks, which assume that the ground truth entities are accessible during testing.

- **TANL** (Paolini et al., 2021) is a generation-based model for monolingual EAE. Their projected target is a phrase that incorporates labels into the passage in the input, such as [Two soldiers|target] were attacked, which denotes that the ”Two soldiers” claim is a ”target” argument. We swap out the T5 (Raffel et al., 2020) pre-trained generative model for mT5-base (Xue et al., 2021) in order to adapt TANL to zero-shot cross-lingual EAE.

- **X-GEAR** is our proposed model. We consider three alternative pre-trained generative language models: mBART-50-large (Tang et al., 2020) and mT5-base (Xue et al., 2021).

3.4 Implementation

We use the code from `https://github.com/PlusLabNLP/X-Gear` for re-implementing X-Gear (Huang et al., 2022). We use mT5-base from (Xu et al., 2021) as our baseline model. To choose the optimal set of parameters, we carried out hyper-parameter tuning by analyzing test results with a number of combinations from learning-rate = [1e-5, 6e-5, 1e-4, 7e-4, 5e-3, 8e-3], train epochs = [20, 40, 60, 75] and batch-size = [8,16,50,64]. We ended up selecting the best model amongst these which was having learning-rate 10e-4, batch size 8 and number of training epochs as 25. check `https://github.com/ramiyappan/X-Gear` for our GitHub repository and detailed explanation about reproducing the experiments in this paper.

3.5 Data Perturbation

We consider **X-Gear** *(mT5-base)* model to analyze it across the dimensions of Robustness and Multilinguality. We use the capabilities mentioned in Checklist (Ribeiro et al., 2020) to analyze its sensitivity towards data perturbation, how our model handles real-world examples and then infer from its performance. For this, we introduce noise in our test dataset which could represent real-world data. Therefore, all the experiments that are performed below are based on the model's predictions on this test set and its results are provided by table 2. The capabilities explored in this study are as follows:

- **Robustness** refers to the ability of a model to handle various sources of noise, such as typos, irrelevant additions, and contractions. To explore the model's capabilities across this dimension, we added noise into our dataset such as spelling-mistakes, irrelevant words and urls, removing few words, and punctuations. For one of the examples, where the input passage was, *“Davies is leaving to become chairman of the London School of Economics, one of the best - known parts of the University of London .”*, the model was able to identify the arguments <*Person*> and <*Entity*> as *”Davies”* and *”London School of Economics”* respectively even after modifying the input with typos and irrelevant additions. However, it failed to capture relevant information and corresponding <*Entity*> argument in a different text that involved additions of irrelevant data and url handles. From table 2, we can infer that overall, the model has a fail-rate of about 18.8% towards real-world noisy data, which is quite good for a Language model to handle noisy data by itself.

- **Taxonomy** is the capability of a model that corresponds to its lexical knowledge of understanding synonyms, antonyms and word categories. Table 2 shows that the model fails to classify gold roles in most of the sentences (48.1% fail-rate). For example, In a test-set instance, the model predicted a person ”Diller” as the <*Agent*> instead of the company ”Twentieth Century Fox”. Similarly, It failed to predict ”U.S.” as the <*Attacker*> and ”city's bridges” as the <*Place*> when we introduced synonyms in the input text. This shows that the model is very sensitive towards changes based on synonyms, antonyms and word categories.

- **Fairness** evaluation helps to ensure that the model's outputs are not biased and do not discriminate based on race, gender or any other protected attributes. To analyze how fair our model is, we modified attributes such as gender, names, nationality, race, religion,

Capability	Fails	Fail-rate (%)	F-score
Robustness	76/403	18.8	0.81
Taxonomy	194/403	48.1	0.51
Fairness	253/403	62.7	0.37
Negation	20/403	4.9	0.85
NER	85/403	21.0	0.78
SRL	112/403	27.7	0.72

Table 2: Statistics of X-Gear (mT5-base) on ACE-05 English test-set having 403 instances, after introducing noise.

and pronouns in our test instances. Specifically we introduced these modifications in examples like *"Supermodel Rachel Hunter on Tuesday filed for divorce from her estranged rocker husband Rod Stewart"*, *"An African-American crashed into the city while flying yesterday."*, and *"3 Black-Asians were killed in a mall shooting that was caused by Russian attackers last Saturday"*. We observed that the model was unable to understand the differences and failed to correctly classify the argument roles when we modified protected attributes such as "African-American" and "Black-Asians". This can be clearly seen by looking at the values provided by table 2 which shows that the model is highly biased and sensitive towards Fairness.

- **Negation** experiments are performed to analyze the model's understanding of negative expressions. By analyzing the predictions of specific cases, it was observed that our model is significantly robust towards Negation. For instance, it was able to correctly figure out the arguments <*Entity*> and <*Place*> in multiple examples of {Personnel:Elect} event type. This is supported by the figures provided by table 2 which shows an F-score of 0.85.

- **Named Entity Recognition (NER)** refers to the model's ability to recognize and classify event-related named entities such as event triggers, types, and participants. Since, our model is already trained for this specific task of Event Argument Extraction, this capability directly corresponds to evaluating the model in our regular setting. The model is very less likely (21% fail-rate) to make mistakes in identifying arguments and entities as seen from table 2.

- **Semantic Role Labeling (SRL)** tests can help to access the model's ability to identify and assign roles to entities and arguments that participate in an event. This also pertains to our regular setting as X-Gear is directly trained to identify event triggers, arguments, and assign respective roles. It can be observed from table 2 that the model incorrectly classified only 112 out of 403 test examples.

3.6 Multilinguality

From the baseline implementation during checkpoint-1, X-Gear's architecture is set-up in a way that could support cross-lingual transfer, i.e., the source and target languages can be completely different while performing an event argument extraction task. This is mainly possible due to the usage of language-agnostic templates that consists of HTML tags as special tokens. Since the model just needs to fill-in the predicted arguments by fixing these special tokens, this facilitates effective cross-lingual transfer and allows the model to easily comprehend different languages.

For the purpose of testing our model against different languages beyond those covered in the original paper (Huang et al., 2022), we augment the test set from ACE-05 English annotations into German, Spanish and French using Google translate. This augmentation process was carefully carried out to ensure that all translations are still of the same respective event type and language-agnostic template.

Table 3 presents the outcomes obtained from various models experimented on our augmented test-set for German, Spanish, and French. The results highlight notable variations among the models. Specifically, CamemBERT exhibits remarkable superiority over other models in handling French examples. This is attributed to its specific pretraining on a French dataset, enabling it to ex-

Model	German	Spanish	French
X-Gear (mT5-base)	36.1	38.3	41.9
X-Gear (mBERT-base)	36.4	39.7	42.4
X-Gear (XLM-R-base)	30.6	31.1	37.6
X-Gear (CamemBERT)	26.1	37.4	**68.9**
X-Gear (BETO)	25.9	**65.6**	36.0
X-Gear (GermanBERT)	**67.2**	25.1	26.4

Table 3: Argument Classification (F1%) Scores of X-Gear with different pretrained models tested on augmented examples of ACE-2005. The best score for each language is in bold.

cel in French language tasks. Similarly, BETO demonstrates superior performance in Spanish due to its targeted pretraining on Spanish data. As the name suggests, GermanBERT shows a significant advantage in dealing with German language tasks compared to other languages and models.

mT5-base, mBERT-base and XLM-R-base are multilingual models pretrained across a wide range of languages that can be used for effectively performing Natural Language understanding and generation tasks. They show reasonable performance on unseen test examples in different languages as seen from the table. However, since the above mentioned language-specific models have an upper-hand in the languages they were trained, they obviously perform better than the multilingual models such as mBERT. Results from table 3 support this inference with noticeable difference in the F1 scores.

3.7 Results

Table 2 presents the F1-scores of Argument Classification by our baseline model X-Gear mT5-base in comparison to the published models X-GEAR (mT5-base and mBART-50-large) (Huang et al., 2022).

- **Comparison to prior generative models** Initially, we noticed that TANL does not perform well when it comes to transferring to various languages. The root cause of this issue is due to TANL's language-dependent template, which makes it susceptible to generating code-switching results, a scenario that pre-trained generative models have not encountered frequently, resulting in low performance. On the other hand, X-GEAR's approach uses language-agnostic templates, and as a result, it achieves better performance for zero-shot cross-lingual transfer.

- **Comparison to classification models** Our experiments demonstrate that X-GEAR, which employs mT5-base, performs better than OneIE, and CL-GCN in nearly all combinations of source and target languages. These results indicate that our proposed approach is highly promising for zero-shot cross-lingual EAE.
  It is important to mention that OneIE, and CL-GCN rely on an extra module for named entity recognition to make predictions. Additionally, CL-GCN require additional annotations for dependency parsing to align the representations of different languages. In contrast, X-GEAR can take advantage of the knowledge learned from pre-trained generative models and therefore does not require any additional modules or annotations.

- **Comparison to different pre-trained generative language models** To our surprise, we found that using mT5-base performs better than using mBART-50-large for X-GEAR, even though they have a similar number of parameters. We believe that this difference is due to the use of special tokens. mBART-50 has separate begin-of-sequence (BOS) tokens for each language. When generating sequences, we need to specify which BOS token to use as the start token. We suspect that these language-specific BOS tokens make it more difficult for mBART-50 to transfer knowledge from the source language to the target language. On the other hand, mT5 does not have language-specific BOS tokens and uses the padding token as the start token during generation. This design is more general and benefits zero-shot cross-lingual transfer.

- **Comparing Our model to X-GEAR** (Huang et al., 2022) Our model outperforms the

Model	en	en	en	ar	ar	ar	zh	zh	zh
	⇓	⇓	⇓	⇓	⇓	⇓	⇓	⇓	⇓
	en	ar	zh	en	ar	zh	en	ar	zh
OneIE (XLM-R-large)	63.6	37.5	42.5	27.5	57.8	**31.2**	51.5	31.1	69.6
CL-GCN (XLM-R-large)	59.8	25.0	29.4	25.4	47.5	19.4	40.8	23.3	62.2
TANL (mT5-base)	59.1	29.7	38.6	18.3	50.1	16.9	33.3	18.3	65.2
X-Gear (mBART-50-large) (Huang et al., 2022)	68.3	37.8	48.9	**30.5**	59.8	29.2	45.9	32.3	63.6
X-Gear (mT5-base) (Huang et al., 2022)	67.9	**42.0**	**53.1**	27.6	66.2	30.5	**52.8**	32.0	**69.4**
Ours	**68.7**	37.2	51.5	26.7	**66.4**	28.9	51.3	**33.4**	66.0

Table 4: Argument Classification (F1%) Scores of ACE-2005 in comparison to the published models. "en ⇒ ar" represents model transferring from en to ar. The best score for each transfer is in bold. Each value is an average of three different random seeds.

X-GEAR (mBART-50-large) and X-GEAR (mT5-base) in en⇒en transfer, ar⇒ar transfer and zh⇒ar transfer marginal difference in values. But, X-GEAR (mT5-base) still performs better for en⇒ar transfer, en⇒zh transfer, zh⇒en transfer and zh⇒zh transfer. And X-GEAR (mBART-50-large) performs better for ar⇒en transfer. Looking at this we can say that there is still room for improvement in our model.

4 Related Work

XTREME-R: Towards More Challenging and Nuanced Multilingual Evaluation (Ruder et al., 2021), proposes a method for zero-shot cross-lingual transfer learning in structured prediction tasks, where models are trained on high-resource languages and then applied to low-resource languages without additional labelled data. They evaluated their method on named entity recognition and dependency parsing tasks, showing significant improvements over baselines. While they build classifiers on top of pre-trained masked language models, our work focuses specifically on generating structured predictions in a zero-shot cross-lingual setting by incorporating prompts in a fine-tuning process.

A Unified Generative Framework for Various NER Subtasks (Yan et al., 2021), proposes a framework that is easy-to-implement and achieves state-of-the art (SoTA) or near SoTA performance on eight English NER datasets, including two flat NER datasets, three nested NER datasets, and three discontinuous NER datasets. It demonstrates the success of generation-based models for named entity recognition in a monolingual setting. However, their methods are language-dependent and not suitable for zero-shot cross-lingual structured prediction. In contrast, our work is designed for the zero-shot cross-lingual setting and can generate structured predictions for multiple languages with a single model.

Neural Cross-Lingual Event Detection with Minimal Parallel Resources (Liu et al., 2019), proposed a new method for cross-lingual ED, demonstrating a minimal dependency on parallel resources. It is effective in performing cross-lingual transfer concerning different directions and tackling the extremely annotation-poor scenario. It proposed using bilingual dictionaries and translation pairs, respectively, to handle the discrepancy between languages in zero-shot cross-lingual structured prediction. Our work, on the other hand, does not require such external resources and instead leverages fixed prompts and fine-tuning of the language model.

Exploiting Cloze Questions for Few Shot Text Classification and Natural Language Inference (Schick and Schutze, 2021), introduced Pattern Exploiting Training (PET), a semi-supervised training procedure that reformulates input examples as cloze-style phrases to help language models understand a given task. PET outperforms supervised training and strong semi-supervised approaches in low resource settings by a large margin. It also proposed a prompt-based method for named entity recognition that uses trainable prompts to adapt the language model to specific tasks. In contrast, our work can generate structured predictions for multiple languages without any labelled data for the target language. Additionally, we focus on zero-shot cross-lingual structured prediction tasks, while theirs is applied to monolingual named entity recognition.

Text generation with exemplar-based adaptive decoding (Peng et al., 2019), proposes an exemplar-based adaptive decoding method for text generation that selects exemplars (i.e., similar instances from a training set) to guide the decoding process. The approach was evaluated on several text generation tasks, including machine translation and text summarization, and showed improved performance over traditional decoding methods. Empirical results show that this model achieves strong performance and outperforms comparable baselines. In contrast, our work on Multilingual Generative Language Models for Zero-Shot Cross-Lingual Event Argument Extraction focuses specifically on the task of event argument extraction and utilizes a multilingual generative language model to generate cross-lingual event arguments without the need for labeled data in the target language. Additionally, our method employs a prompt-based approach to guide the model's behavior, rather than selecting exemplars from a training set.

5 Conclusion

We present a novel approach to zero-shot cross-lingual event argument extraction (EAE) by leveraging multilingual pre-trained generative language models. Our suggested model effectively encodes event structures and captures interdependence between arguments by treating EAE as a language creation process. We employ language-agnostic templates that represents event argument structures, making them compatible with any language and facilitating cross-lingual transfer. We also Investigated the robustness of our model to noisy or incorrect input data by exploring techniques such as changing synonyms, adding typos and irrelevant data that helped examine the model's robustness towards handling real-world noise.

Additionally, we also explored the Multilinguality dimension by augmenting our examples to other languages and switching our baseline models to other models such as XLM-R-base, CamemBERT etc. Our experimental findings showed that, in zero-shot cross-lingual EAE, our model, X-GEAR, outperforms the most recent state-of-the-art models. This highlights the potential of using a language generation framework for solving structured prediction tasks in a zero-shot cross-lingual setting.

6 Future work

Although our study achieved promising results, there are still a few steps for future research that we were unable to explore in this semester. These include:

- **Expansion to more languages** We conducted experiments on a small selection of source and target languages. Our method's generalizability and effectiveness could be better understood by applying it to a wider variety of languages.

- **Fine-grained evaluation** The performance of event argument extraction was the main focus of our evaluation. Conducting a more detailed analysis, including examining individual argument roles, would provide a deeper understanding of the model's strengths and weaknesses.

- **Incorporating domain-specific knowledge:** Adapting our approach to particular domains or incorporating domain-specific knowledge could enhance the model's accuracy and applicability in real-world applications.

- **Model optimization and efficiency** Exploring methods for model efficiency and optimization, such as model compression or knowledge distillation, might be helpful, especially for resource-constrained environments.

- **User feedback and iterative improvements** Gathering user feedback and implementing it into iterative model improvements can help address practical challenges and further enhance the usability and performance of our approach.

By addressing these future research directions, We may further develop the field of zero-shot cross-lingual event argument extraction and explore the full potential of generative language models in structured prediction tasks.

References

Wasi Uddin Ahmad, Nanyun Peng, and Kai-Wei Peng. 2021. Gate: Graph attention transformer encoder for cross-lingual relation and event extraction. In *In Thirty-Fifth AAAI Conference on Artificial Intelligence (AAAI).*

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online. Association for Computational Linguistics.

George Doddington, Alexis Mitchell, Mark Przybocki, Lance Ramshaw, Stephanie Strassel, and Ralph Weischedel. 2004. The automatic content extraction (ACE) program – tasks, data, and evaluation. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, Lisbon, Portugal. European Language Resources Association (ELRA).

I-Hung Hsu, Kuan-Hao Huang, Elizabeth Boschee, Scott Miller, Prem Natarajan, Kai-Wei Chang, and Nanyun Peng. 2022. Degree: A data-efficient generation-based event extraction model. In *arXiv preprint arXiv:2108.12724.*

Kuan-Hao Huang, I-Hung Hsu, Premkumar Natarajan, Kai-Wei Chang, and Nanyun Peng. 2022. Multilingual generative language models for zero-shot cross-lingual event argument extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL).*

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks.

Sha Li, Heng Ji, and Jiawei Han. Document-level event argument extraction by conditional generation. In *In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT).*

Ying Lin, Heng Ji, Fei Huang, and Lingfei Wu. 2020. A joint neural model for information extraction with global features. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

Jian Liu, Yubo Chen, Kang Liu, and Jun Zhao. 2019. Neural cross-lingual event detection with minimal parallel resources. 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).*

Minh Van Nguyen and Thien Huu Nguyen. 2021. Improving cross-lingual transfer for event argument extraction with language-universal sentence structures. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop.*

Jian Ni and Radu Florian. 2019. Neural cross-lingual relation extraction based on bilingual word embedding mapping. 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).*

Giovanni Paolini, Ben Athiwaratkun, Jason Krone, Jie Ma, Alessandro Achille, Rishita Anubhai, Cicero Nogueira dos Santos, Bing Xiang, and Stefano Soatto. 2021. Structured prediction as translation between augmented natural languages.

Hao Peng, Ankur Parikh, Manaal Faruqui, Bhuwan Dhingra, and Dipanjan Das. 2019. Text generation with exemplar-based adaptive decoding. 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).*

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online. Association for Computational Linguistics.

Sebastian Ruder, Noah Constant, Jan Botha, Aditya Siddhant, Orhan Firat, Jinlan Fu, Pengfei Liu, Junjie Hu, Dan Garrette, Graham Neubig, and Melvin Johnson. 2021. XTREME-R: Towards more challenging and nuanced multilingual evaluation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*.

Timo Schick and Hinrich Schutze. 2021. Exploiting cloze questions for few shot text classification and natural language inference. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers).*

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks.

Ananya Subburathinam, Di Lu, Heng Ji, Jonathan May, Shih-Fu Chang, Avirup Sil, and Clare Voss. 2019. Cross-lingual structure transfer for relation and event extraction. 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).

Yuqing Tang, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. 2020. Multilingual translation with extensible multilingual pretraining and finetuning. In *arXiv preprint arXiv:2008.00401*.

David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. 2019. Entity, relation, and event extraction with contextualized span representations. 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)*.

Xiaozhi Wang, Ziqi Wang, Xu Han, Zhiyuan Liu, Juanzi Li, Peng Li, Maosong Sun, Jie Zhou, and Xiang Ren. 2019. HMEAE: Hierarchical modular event argument extraction. 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)*, Hong Kong, China. 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*.

Haoran Xu, Seth Ebner, Mahsa Yarmohammadi, Aaron Steven White, Benjamin Van Durme, and Kenton Murray. 2021. Gradual fine-tuning for low-resource domain adaptation. In *Proceedings of the Second Workshop on Domain Adaptation for NLP*.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Hang Yan, Tao Gui, Junqi Dai, Qipeng Guo, Zheng Zhang, and Xipeng Qiu. 2021. A unified generative framework for various NER subtasks. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*.

Bowei Zou, Zengzhuang Xu, Yu Hong, and Guodong Zhou. 2018. Adversarial feature adaptation for cross-lingual relation classification. In *Proceedings of the 27th International Conference on Computational Linguistics*.