A Regularization-based Transfer Learning Method for Information Extraction via Instructed Graph Decoder

Kedi Chen, Jie Zhou, Qin Chen*, Shunyu Liu, Liang He
School of Computer Science and Technology, East China Normal University, Shanghai, China
{kdchen,71265901045}@stu.ecnu.edu.cn {jzhou, qchen, lhe}@cs.ecnu.edu.cn

Abstract

Information extraction (IE) aims to extract complex structured information from the text. Numerous datasets have been constructed for various IE tasks, leading to time-consuming and labor-intensive data annotations. Nevertheless, most prevailing methods focus on training task-specific models, while the common knowledge among different IE tasks is not explicitly modeled. Moreover, the same phrase may have inconsistent labels in different tasks, which poses a big challenge for knowledge transfer using a unified model. In this study, we propose a regularization-based transfer learning method for IE (TIE) via an instructed graph decoder. Specifically, we first construct an instruction pool for datasets from all well-known IE tasks, and then present an instructed graph decoder, which decodes various complex structures into a graph uniformly based on corresponding instructions. In this way, the common knowledge shared with existing datasets can be learned and transferred to a new dataset with new labels. Furthermore, to alleviate the label inconsistency problem among various IE tasks, we introduce a task-specific regularization strategy, which does not update the gradients of two tasks with 'opposite direction'. We conduct extensive experiments on 12 datasets spanning four IE tasks, and the results demonstrate the great advantages of our proposed method.

Keywords: Information Extraction, Transfer Learning, Instruction Learning

1. Introduction

Information extraction (IE) is a task to extract structured information (e.g., entities, relationships, and events) from textual data. IE encompasses many subtasks, including named entity recognition (NER), relation extraction (RE), event extraction (EE), and aspect-based sentiment analysis (ABSA). It is a challenging task due to the large label space and complex structure of various tasks.

Existing researches in IE can be categorized into two main classes: task-specific and unified models. Task-specific models (Chen et al., 2023; Wadhwa et al., 2023; You et al., 2023; Ma et al., 2023) entail designing a unique structure for each individual task. These independent architectures require higher development costs and resources. Unified models, on the other hand, deploy a cohesive framework to address multiple tasks simultaneously. Presently, unified models predominantly employ a generative framework, translating extraction tasks into a sequence generation architecture (Lu et al., 2022; Huang and Chang, 2023). Although the frameworks are structurally unified, most of the previous methods (Lu et al., 2022; Yan et al., 2023) merely finetune the models on the target dataset, disregarding the common knowledge within numerous existing IE datasets, including ACE 2005 (Walker et al., 2006), CoNLL03 (Sang and Meulder, 2003), 16-res (Pontiki et al., 2016), and others. This paper emphasizes the acquisition of shared knowledge across these tasks and datasets.

CoNLL03 example	**label**	**mention**
VICORP restaurants[ORG] names Sabourin CFO.	PER	-
	ORG	restaurants
	LOC	-
	...	...
ACE05-Rel example	**label**	**mention**
The explosion comes after a bomb exploded at a restaurant[FAC] in Istanbul[GPE], leading to damage but no injuries.	PER	-
	ORG	-
	FAC	restaurant
	...	...
	PartWhole†	restaurant,Istanbul
	...	...
ACE05-Evt example	**label**	**mention**
He was a segregationist who once closed a restaurant he owned[Trig] rather than served African-Americans.	Trig	owned
	ORG†	owned,restaurant
	Place†	-
	PER†	-
	...	...
16-res example	**label**	**mention**
The Petrus and Vonglas's tiny restaurant[Asp] is as cozy[Exp] as it gets, with that certain Parisian flair.	Expression	cozy
	Aspect	restaurant
	Positive†	cozy,restaurant
	Negative†	-
	Neutral†	-

Table 1: An example of inconsistent annotations among different subtasks. † means the relation labels. A relation label exists between two words with underscores.

Nevertheless, several challenges persist in knowledge transfer across distinct IE tasks. **First**, the datasets originate from various IE tasks, resulting in substantial diversity in data structures. More specifically, 1) Two datasets from the same subtask may exhibit distinct entity or relation types. Thus, the target datasets may contain new label classes that do not occur in the source datasets; 2) Although two entity or relation types are semantically similar in different datasets, they are labeled with

* Corresponding author.

LREC-COLING 2024, pages 1472–1485

different names. For instance, the relation types 'OrgBased_In' in CoNLL04 and 'PART-WHOLE' in ACE05-Rel share the same semantic meaning, signifying 'locate in'; 3) Furthermore, certain labels encompass the meanings of multiple other labels. 'MISC' of CoNLL03 is applied to label diverse miscellaneous entities. ACE05-Rel utilizes 'GEN-AFF' to denote generic affiliations without specific references. These labels with vague semantics significantly influence the model's learning process. **Second**, in different datasets pertaining to distinct IE subtasks, the same phrase may have inconsistent labels owing to various annotation guidelines. As depicted in Table 1, the phrase 'restaurant' in several datasets related to different IE subtasks exhibits different annotation information, including 'ORG', 'FAC', 'Aspect', etc. This discrepancy introduces conflicts in the comprehension of the phrase for various IE subtasks.

To address the aforementioned challenges, we introduce a regularization-based transfer learning method for IE (TIE) via an instructed graph decoder. First, we design an instructed graph decoder to learn task-shared knowledge by modeling the various formats of different IE tasks as a graph. Then, we propose a task-specific regularization transfer strategy to resolve conflicting knowledge among tasks. The instructed graph decoder consists of two parts: 1) Instruction pool, which contains manually crafted task-specific instructions for each dataset of different IE tasks. These instructions serve as guiding text to facilitate model's adaptation to different datasets and mitigate disparities, thereby enhancing the generalization capability; 2) Graph decoder, which decodes various formats of different tasks into a unified graph structure with instructions. The task-specific regularization strategy does not update the gradients of two tasks 'in the opposite direction', for resolving conflicting knowledge across tasks, ultimately preparing the model for testing on the target dataset. The experimental results demonstrate that our approach achieves state-of-the-art in most of the IE datasets, even with improvements in data-scarce scenarios.

The main contributions of this paper can be summarized as follows:

- We propose a TIE method that explicitly models the common knowledge from various IE datasets with an instructed graph decoder.
- A task-specific regularization strategy is designed to help reduce the inconsistent labels or conflicts across diverse IE tasks, by not updating the gradients 'in the opposite direction' during transfer learning.
- Experiments on 12 datasets of four IE subtasks show the advantages of our proposed method. Moreover, our method is superior to the baselines on low-resource and few-shot scenarios[1].

2. Related Work

Information Extraction Information extraction (IE), deriving structured information from unstructured source data, is an essential task in natural language processing (NLP). Information extraction contains several subtasks, such as named entity recognition (Marrero et al., 2013), relation extraction (Cui et al., 2017), event extraction (Wadden et al., 2019), aspect-based sentiment analysis (Do et al., 2019), etc. For a period of time, researchers tend to work on these subtasks separately.

In recent years, Lu et al. (2022) proposes a generative unified information extraction (UIE) model with structured extraction language and structural schema instructor. The generative paradigm generates too much redundant information and has poor completeness. The same authors then introduce a new framework USM (Lou et al., 2023) with token linking operations. However, USM brings unnecessary loss of time in both training and inference periods. The Plusformer architecture harnessed by Yan et al. (2023) requires high algorithmic complexity, hence simplification is indispensable. Ping et al. (2023) converts IE tasks into span classification via the triaffine mechanism, but the reliability on syncretic complex-label datasets has not been validated.

With the advent of large language models (LLMs) (Huang and Chang, 2023), there have been significant changes in IE. ChatIE (Wei et al., 2023) makes an initial attempt to use ChatGPT3.5 for performing information extraction tasks, through multi-turn conversations. The accuracy is not as precise as expected. Li et al. (2023); Han et al. (2023) assess the information extraction capabilities of ChatGPT3.5 systematically, and find a gap between ChatGPT3.5 and SOTA results. InstructUIE presented by Wang et al. (2023b) tests on 32 diverse information extraction datasets, employing language model FlanT5-11B (Chung et al., 2022) in a generative pattern. This method consumes a significant amount of computational resources, making the reproducibility of results challenging.

Given the aforementioned issues, our method leverages a simple architecture and is capable of addressing complex annotations, finally achieving commendable performances in both small and large language model settings.

Gradient Regularization is a regularization technique for deep learning, in order to improve generalization performance and prevent overfitting (Li

[1]Our codes can be found in https://github.com/141forever/TransferUIE

and Spratling, 2023). This technique is widely deployed for coordinating the training of multiple tasks and preventing interference between them (Saha et al., 2021; Lin et al., 2022). A previous study illustrates that *If the angle between the gradients of the current task and the past task is acute, it is less likely to increase the loss of the previous task* (Lopez-Paz and Ranzato, 2017). This finding serves as a critical theoretical foundation for our method, presenting a possibility that models can resolve inconsistent knowledge of different tasks.

Transfer Learning for IE Transfer learning is an important approach to enhance the generalization of deep learning. The purpose of transfer learning is to enhance the performance of models within target domains by leveraging the knowledge from correlated source domains (Zhuang et al., 2021). In the field of IE, many works indicate the superiority of transfer learning. When it comes to NER, Bhatia et al. (2020) proposes a dynamic transfer network to learn sharing parameters between tasks. Di et al. (2019) addresses the label sparsity problem of relation extraction in a real-world scenario. By combining variational information bottleneck into a model called SharedVIB which can search for structured common knowledge, Zhou et al. (2022) boosts the correlation between three event argument extraction tasks. However, these works merely focus on the knowledge among one IE subtask (intra). In contrast, our approach focuses on the inter-task transfer. Moreover, we explore to resolve the inconsistent labels or conflicts during transfer learning.

3. Our Method

In this section, we introduce the framework of our method (Figure 1), which consists of two parts: instructed graph decoder and task-specific regularization. First, we manually craft instructions for each dataset and utilize ChatGPT3.5 to paraphrase, forming an instruction pool. Then, we present an instructed graph decoder to obtain the instruction-activating representations of the input text. It also learns common knowledge by modeling all the structured information with a graph represented by a token matrix. Moreover, in order to alleviate the conflicts between various IE tasks, we present a task-specific regularization strategy that does not update the gradients 'in opposite direction' between source tasks during training on source datasets, and finally finetune on the target dataset.

3.1. Task Definition

We regard any single IE task as an instruction-activating span annotation mission on dataset $\mathcal{D}$.

Dataset	Instruction Example
ACE04	Identify entities (organization, person, vehicle, geographic, location, weapon, facility) mentioned in the sentence.
CoNLL04	Explore the relationships work for, locate in, base in, live in, and kill someone between the entities location, organization, people and other.
ACE05-Evt	Locate the mentioned event types: acquit,..., trial hearing. Identify the argument types: adjudicator,..., victim.
16-res	Find the sentiment (positive, negative or neutral) of the sentence and identify the expression, aspect element.

Table 2: One instruction example for four datasets of different IE subtasks respectively.

Given an instance $(\boldsymbol{x}, \mathcal{E}, \mathcal{R}, \mathcal{I}) \in \mathcal{D}$, where $\boldsymbol{x} = (x_1, \ldots, x_{|\boldsymbol{x}|})$ is the input sentence with $|\boldsymbol{x}|$ tokens. $\mathcal{E}$, $\mathcal{R}$, $\mathcal{I}$ denotes the set of entity types, the set of relation types and the set of instructions separately. Regarding the named entity recognition task, $\mathcal{R} = \varnothing$. As for the event extraction task, $\mathcal{E} = \mathcal{T}$ and $\mathcal{R} = \mathcal{A}$, where $\mathcal{T}$ regards the set of trigger types or event types, and $\mathcal{A}$ represents the set of argument roles. In reference to aspect-based sentiment analysis task, $\mathcal{E} = \{\text{Aspect}, \text{Expression}\}$ and $\mathcal{R} = \{\text{Positive}, \text{Negative}, \text{Neutral}\}$. We aim to achieve a scoring matrix $\mathbf{M}^{|x|*|x|*(|\mathcal{E}|+|\mathcal{R}|)}$, which can indicate the label of each span of the input sentence. $\mathbf{M}[i, j, k] = 1$ means the span (i, j) of the input sentence has label k.

3.2. Instructed Graph Decoder

In this module, we first use an instruction pool to translate the IE labels into instructions so that the model can learn the representations of class effectiveness and capture new label classes. Then, we apply a graph decoder based on instructions obtained from the instruction pool to decode the complex and various structures into a graph uniformly. We combine the instruction pool and the graph decoder together, referring to them as an instructed graph decoder.

Instruction Pool For various datasets or tasks, the label spaces are different. We create a set of instructions for each dataset $\mathcal{D}$, all the instructions are referred to as an instruction pool. Each instruction contains all entity types $e \in \mathcal{E}$ and relation types $r \in \mathcal{R}$ of the corresponding dataset. In this way, the model can learn representations of similar labels and new classes.

For each dataset, we first write an instruction manually. Take dataset 16-res of ABSA task as an example, given entity types $\mathcal{E}$ and relation types $\mathcal{R}$, the instruction we designed artificially is: "*Annotate the polarity (positive, negative or neutral),*

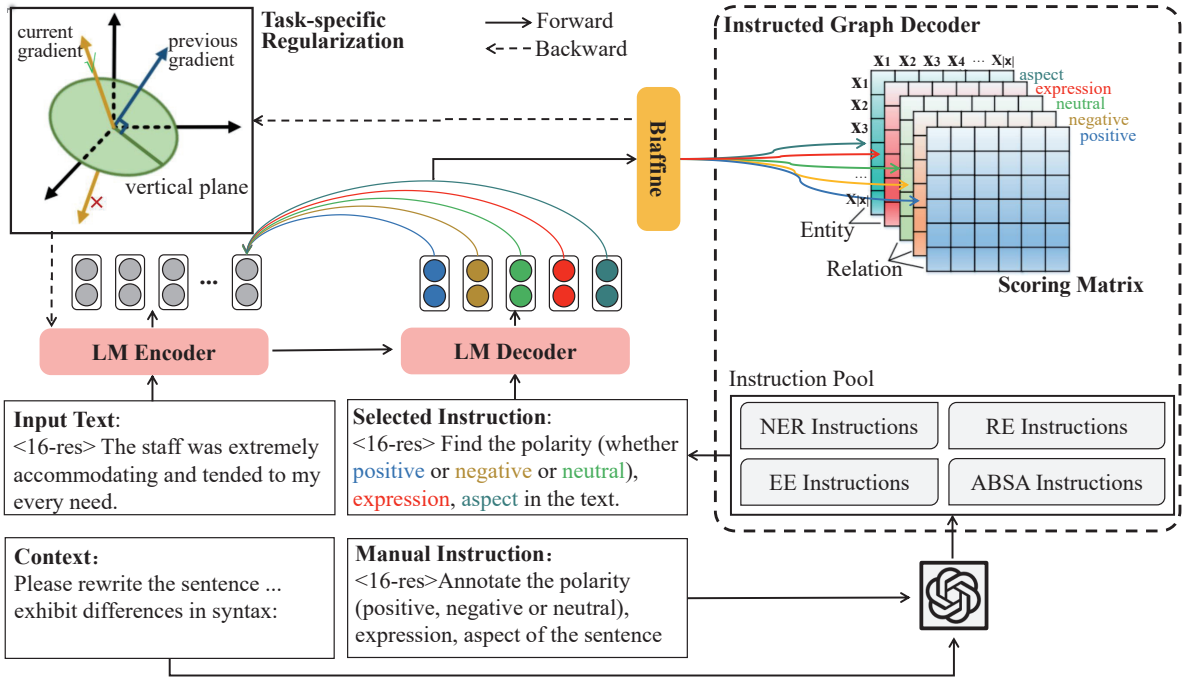


Figure 1: The framework of our TIE method.

expression, aspect of the sentence." Then, to improve the diversity of the instruction, we adopt ChatGPT3.5 to augment the instruction. Specifically, the complete prompt input into ChatGPT3.5 is: "*Please rewrite the following sentence several times and make sure the rewritten sentences exhibit significant differences in syntax, compared to the original sentence: Annotate the polarity (positive, negative or neutral), expression, aspect of the sentence.*" More details of the construction of the instruction pool are shown in Appendix 8.1.

Instructions of the other datasets can be obtained in the same way. We provide one instruction example for four datasets of different IE subtasks respectively in Table 2. For more examples, please refer to Appendix 8.2.

Graph Decoder To capture the complex structures of various IE tasks, we design a graph decoder to decode all the structured information as a graph. Given an input text with $|x|$ tokens $\boldsymbol{x} = [x_1, \ldots, x_{|x|}]$, we harness T5 (Raffel et al., 2020; Chung et al., 2022) series to model the sentences and instructions. It is an adaptable encoder-decoder pre-trained language model (PLM) $\mathcal{M} = [\mathcal{M}_{enc}, \mathcal{M}_{dec}]$ designed to tackle many NLP tasks.

We first use the encoder of PLM to obtain the hidden representation of input sentence $\boldsymbol{x}$ as follows.

$$\mathbf{H}_x^{enc} = \left[\mathbf{h}_x^1, \ldots, \mathbf{h}_x^{|x|}\right] = \mathcal{M}_{enc}([x_1, \ldots, x_{|x|}]) \quad (1)$$

where $\mathbf{H}_x^{enc} \in \mathbb{R}^{|x|*d}$, d is the dimension of hidden layers.

Next, to model the interaction between the sentence and the instruction, the decoder part of the PLM is leveraged to get sentence-aware instruction representation.

As mentioned earlier, we construct several diverse instructions for each dataset of different IE subtasks, which make up an instruction pool. For each sample, we randomly select an instruction corresponding to the dataset. The selected corresponding instruction is denoted as $\boldsymbol{u}$ with length $|u|$ from the instruction pool and is inputted into the decoder.

$$\mathbf{H}_u^{dec} = \left[\mathbf{h}_u^1, \ldots, \mathbf{h}_u^{|u|}\right] = \mathcal{M}_{dec}(\mathbf{H}_x^{enc}; \boldsymbol{u}) \quad (2)$$

where $\mathbf{H}_u^{dec} \in \mathbb{R}^{|u|*d}$, d is the size of the hidden dimension.

We can then achieve the representations of $K = |\mathcal{E}| + |\mathcal{R}|$ label slots, $\mathbf{H}_{slot} = \left\{\mathbf{h}_u^{\text{slot\_index}(i)}\right\}_{i=1}^{K}$, where slot_index (i) is the index of the i-th label slot in the instruction. Each $\mathbf{h}_u^{\text{slot\_index}(i)} \in \mathbf{H}_u^{dec}$ and $\mathbf{H}_{slot} \in \mathbb{R}^{K*d}$.

Finally, to obtain the label-sensitive text representation $\mathbf{H}_x = \left[\mathbf{h}^1, \ldots, \mathbf{h}^{|x|}\right]$, we deploy attention operations (Vaswani et al., 2017) to $\mathbf{H}_x^{enc}$ and $\mathbf{H}_{slot}$.

$$\mathbf{H}_x = \text{Softmax}(\mathbf{H}_x^{enc}\mathbf{W}_1(\mathbf{H}_{slot}\mathbf{W}_2)^T)\mathbf{H}_{slot}\mathbf{W}_2 \quad (3)$$

$\mathbf{W}_1, \mathbf{W}_2 \in \mathbb{R}^{d*d}$ are learnable parameters.

At last, we represent the graph structure of the tokens using a matrix and calculate the scoring matrix in a biaffine way (Barnes et al., 2021; Yan

et al., 2023) with multilayer perceptron (MLP).

$$\mathbf{H}_x^{head} = MLP_{head}(\mathbf{H}_x) \quad (4)$$

$$\mathbf{H}_x^{tail} = MLP_{tail}(\mathbf{H}_x) \quad (5)$$

$$\begin{aligned}\mathbf{M}_x[i,j] =& (\mathbf{H}_x^{head}[i])^T \mathbf{W}_3 \mathbf{H}_x^{tail}[j] \\ &+ \mathbf{W}_4[\mathbf{H}_x^{head}[i]; \mathbf{H}_x^{tail}[j]] \quad (6)\end{aligned}$$

$$\mathbf{M} = MLP_{score}(\mathbf{M}_x) \quad (7)$$

where $\mathbf{H}_x^{head}, \mathbf{H}_x^{tail} \in \mathbb{R}^{|x|*d}$, $\mathbf{M}_x, \mathbf{M} \in \mathbb{R}^{|x|*|x|*K}$, $\mathbf{W}_3 \in \mathbb{R}^{d*K*d}$,$\mathbf{W}_4 \in \mathbb{R}^{K*2d}$. $[;]$ means the concatenation between two vectors.

3.3. Task-Specific Regularization

Training all IE datasets within a unified model can facilitate the acquisition of shared knowledge across diverse datasets. Nevertheless, differences in task definitions and annotation guidelines can result in inconsistent labels. These variations in task-specific knowledge significantly impact the effectiveness of transfer learning. Consequently, we introduce a task-specific regularization technique, aimed at mitigating the influence of task-specific knowledge.

Task-Specific Knowledge Unlearning During this step, we design a task-specific regularization method to unlearn conflicting knowledge. Particularly, to resolve the conflicting knowledge among tasks, we do not update the model when the gradients of two consecutive tasks are 'in an opposite direction'. That is, parameters are updated only when the angle between the current gradient and the previous time step's gradient is less than 90 degrees, otherwise, no update is performed (Lopez-Paz and Ranzato, 2017). We ensure that all data within one batch came from the same dataset, while the data in the two adjacent batches come from two different datasets of different IE tasks. Under the circumstances, the neighboring gradients signify the updating directions for different tasks. The angle is determined by the sign of the dot product result between two consecutive gradients.

$$\textbf{Update} = \begin{cases} \text{True}, & \text{if } \langle \mathbf{g}_t, \mathbf{g}_{t-1} \rangle > 0 \\ \text{False}, & \text{otherwise} \end{cases} \quad (8)$$

where $\mathbf{g}_t$ and $\mathbf{g}_{t-1}$ are the gradients of the adjacent two batches respectively. If **Update** $=$ Fasle, we freeze the parameters of the corresponding layers.

Then, we finetune the transferred model directly on the target dataset. We take advantage of binary cross-entropy (BCE) loss to optimize the model.

$$\mathcal{L}(\mathbf{M}, \mathbf{G}) = \sum_{i=1}^{|x|} \sum_{j=1}^{|x|} \sum_{r=1}^{K} \text{BCE}(\mathbf{G}[i,j,r], \mathbf{M}[i,j,r]) \quad (9)$$

where $\mathbf{G}$ is the ground truth matrix, K denotes the number of label slots and r is the index of each label.

The task-specific regularization strategy is applied for updating parameters of the whole model, including the instructed graph decoder, which aims to preserve common knowledge and resolve conflicting knowledge among tasks during pre-training. Then, we finetune the whole pre-trained model including the instructed graph decoder on the target dataset.

4. Experimental Setups

4.1. Datasets

For the main experiment, we follow the previous works (Lu et al., 2022) and select 12 IE benchmark datasets of 4 IE subtasks: NER, RE, EE and ABSA. The specific datasets include: ACE04 (Mitchell et al., 2005), ACE05-Ent (Walker et al., 2006), CoNLL03(Sang and Meulder, 2003); CoNLL04 (Roth and Yih, 2004), ACE05-Rel (Walker et al., 2006), SciERC (Luan et al., 2018); ACE05-Evt (Walker et al., 2006), CASIE (Satyapanich et al., 2020); 14-res (Pontiki et al., 2014), 14-lap (Pontiki et al., 2014), 15-res (Pontiki et al., 2015), 16-res (Pontiki et al., 2016). According to our transfer learning configuration, we pre-train the model to learn the common knowledge and alleviate the inconsistency on 11 source datasets, and then fintune on the target dataset. The specific information about these datasets can be found in Table 3.

For data-scarce scenarios, in order to make a fair comparison with Lu et al. (2022), we adopt CoNLL03, CoNLL04, ACE05-Evt and 16-res datasets in few-shot and low-resource settings.

4.2. Metrics

We employ Micro-F1 to assess the model's performance across various IE tasks.

- **Entity (Ent.F1).** An entity is correct if its entity type and span offsets both match a ground truth.
- **Relation (Rel.F1).** A relation is correct if its type, along with the types and span offsets of both head and tail entities all match a ground truth.
- **Event trigger (Trig.F1).** An event trigger is correct if its offsets and the event type both match a ground truth.
- **Event argument (Arg.F1).** An event argument is correct if its offsets, role type and event type all match a ground truth.
- **Sentiment Triplet (Senti Trip.F1).** We actually conduct an aspect sentiment triplet extraction

Dataset	#Train	#Dev	#Test	|Ent|	|Rel|	|Evt|
ACE04	6,297	742	824	7	-	-
CoNLL03	14,041	3,250	3,453	4	-	-
ACE05-Ent	7,178	960	1,051	7	-	-
ACE05-Rel	10,051	2,424	2,050	7	6	-
CoNLL04	922	231	288	4	5	-
SciERC	1,861	275	551	6	7	-
ACE05-Evt	19,204	901	676	-	-	33
CASIE	5,235	1,115	2,121	-	-	5
14-res	1,266	310	492	2	3	-
14-lap	906	219	328	2	3	-
15-res	605	148	322	2	3	-
16-res	857	210	326	2	3	-

Table 3: Dataset statistics. # means the number of instances, and |*| is the number of categories of the corresponding dataset.

(ASTE) task, so a sentiment triplet is correct if its offsets of expression (opinion), offsets of aspect and the sentimental polarity all match a ground truth.

4.3. Baselines

To validate the effectiveness of our method, we select several task-specific models and four unified models as baselines, compared with our approach.

These task-specific methods are shown as follows.

- **BERT-base**[2] (Devlin et al., 2019) is the most famous PLM for many nlp tasks. The results of ACE04 and ACE05-Ent are copied from Peng et al. (2023), which replaces the backbone of UIE (Lu et al., 2022) with BERT-base. The results of four ABSA tasks are from Xu et al. (2021). It is a span-level method, considering the interaction between the spans of targets and opinions.
- **UnifiedNER** (Yan et al., 2021) utilizes a seq2seq framework for three NER datasets. Given the similarity, we select the Span setting.
- **NERGraph** (Wan et al., 2022) treats a sentence as a graph, applying graph convolutional network (GCN) for encoding.
- **PURE** (Zhong and Chen, 2021) works on two independent encoders and solely uses the entity model to construct the relation model.
- **DEGREE** (Hsu et al., 2022) manually designs prompts to help event extraction task.
- **BDTF** (Zhang et al., 2022) is a boundary-driven table filling (BDTF) approach for ABSA tasks.

Here, we introduce the four unified models.

- **TANL** (Paolini et al., 2021) is an early-stage unified information extraction model.
- **UIE** (Lu et al., 2022) is a popular unified information extraction framework in the generative way. To ensure consistency in the backbone, we chose results from the official UIE-base model.
- **ChatGPT3.5**[3] (Li et al., 2023; Han et al., 2023) is a groundbreaking conversational LLM developed by OpenAI. Researchers assess the information extraction capabilities of ChatGPT3.5 from many perspectives systematically. Since the code is not open-source, all reports are based on zero-shot setting.
- **InstructUIE** (Wang et al., 2023b) is an end-to-end LLM framework for universal information extraction, which harnesses FlanT5-11B[4] as the backbone.

4.4. Implementation Details

In order to make a fair comparison with the four unified methods, we leverage T5-base[5] (Raffel et al., 2020) and FlanT5-3B[6] (Chung et al., 2022) (due to resource constraint). According to transfer learning configuration, we pre-train the model on 11 source datasets, and then fintune on the target dataset. In light of the randomness in instruction selection, we fix the seed, repeat the experiment five times and average the outcomes as the reported results. For each dataset, we train on the training set, the reported results on the test set are derived from the checkpoint that yields the best performance on the development set. Except for the results of our method, all other data is recorded from the original papers of the baselines.

5. Experimental Analysis

5.1. Main Results

To evaluate the effectiveness of TIE, we compare our method with several strong baselines (Table 4). From the results, we can get the following conclusions. **First**, *there is an advantage of TIE by deploying the conventional language model.* TIE with backbone T5-base exceeds on 10 out of

[2]https://huggingface.co/google-bert/bert-base-uncased

[3]https://chat.openai.com/

[4]https://huggingface.co/google/flan-t5-xxl

[5]https://huggingface.co/google-t5/t5-base

[6]https://huggingface.co/google/flan-t5-xl

	ACE04	ACE05-Ent	CoNLL03	ACE05-Rel	CoNLL04	SciERC	ACE05-Evt		CASIE		14-res	14-lap	15-res	16-res
	Ent.F1			Rel.F1			Trig.F1	Arg.F1	Trig.F1	Arg.F1	Senti	Trip.F1		
BERT-base	84.09	84.63	-	-	-	-	-	-	-	-	71.85	59.38	63.27	70.26
UnifiedNER	84.22	82.31	92.88	-	-	-	-	-	-	-	-	-	-	-
NERGraph	86.31	85.11	-	-	-	-	-	-	-	-	-	-	-	-
PURE	-	-	-	63.90	-	35.60	-	-	-	-	-	-	-	-
DEGREE	-	-	-	-	-	-	70.90	56.30	-	-	-	-	-	-
BDTF	-	-	-	-	-	-	-	-	-	-	74.35	61.74	66.12	72.27
TANL	-	84.90	91.70	63.70	71.40	-	68.40	47.60	-	-	-	-	-	-
UIE	85.69	83.88	91.94	62.73	73.48	35.35	71.33	50.62	69.14	58.56	72.55	**62.94**	64.41	72.86
ChatGPT3.5	-	-	67.20	40.50	-	25.90	15.50	30.90	-	-	41.50	33.17	38.89	47.67
InstructUIE (FlanT5-11B)	-	86.66	92.94	-	-	-	**77.13**	**72.94**	67.80	63.53	-	-	-	-
TIE (T5-base)	87.59	86.42	92.92	64.44	73.58	34.41	73.09	56.71	74.43	63.14	75.69	60.36	66.78	**75.17**
TIE (FlanT5-3B)	**88.86**	**87.74**	**93.17**	**64.45**	**74.32**	**40.90**	74.89	63.30	**75.38**	**66.99**	**76.97**	62.04	**66.84**	74.05

Table 4: Main results of TIE and the baselines. The upper part and the middle part are task-specific and unified methods respectively. The best result of each dataset is bolded, and the second-best is underlined.

	ACE04	ACE05-Ent	CoNLL03	ACE05-Rel	CoNLL04	SciERC	ACE05-Evt		CASIE		14-res	14-lap	15-res	16-res
	Ent.F1			Rel.F1			Trig.F1	Arg.F1	Trig.F1	Arg.F1	Senti	Trip.F1		
TIE (T5-base)	**87.59**	**86.42**	**92.92**	**64.44**	**73.58**	**34.41**	**73.09**	**56.71**	**74.43**	**63.14**	**75.69**	**60.36**	**66.78**	**75.17**
- Instruction	86.18	84.22	91.67	62.87	71.71	32.79	71.97	52.19	68.78	57.77	73.08	58.55	64.32	72.93
- Transfer	87.44	85.11	92.15	63.69	73.49	34.11	72.85	55.11	73.57	62.38	73.95	59.69	66.61	73.79
- Regularization	86.14	85.95	91.86	64.09	73.46	33.87	72.64	55.32	74.12	62.53	72.78	58.68	65.73	73.06

Table 5: Ablation studies for 12 IE datasets with T5-base backbone.

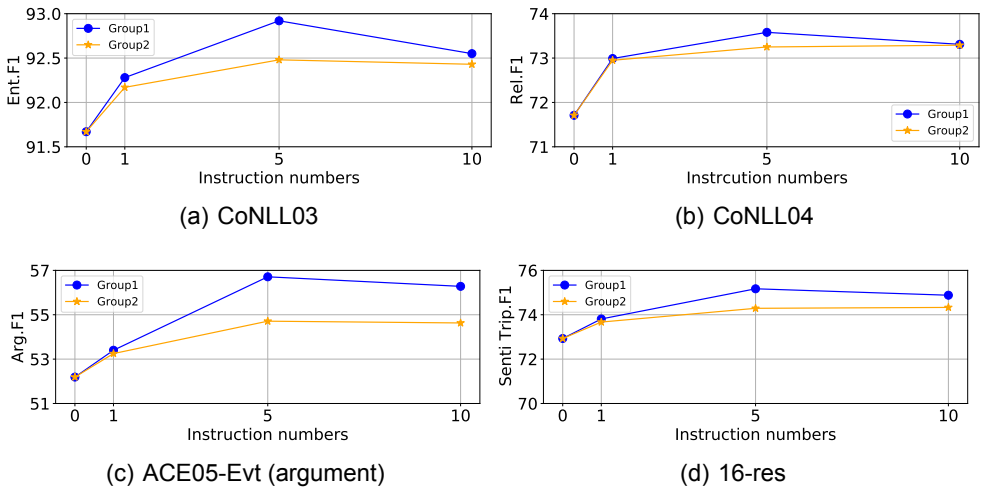
Dataset	Method	Few-Shot			Low-Resource			Full	AVG
		1-shot	5-shot	10-shot	1%	5%	10%	100%	
CoNLL03	UIE	46.43	67.09	73.90	82.84	88.34	89.63	91.94	77.16
	TIE (T5-base)	**49.46**	**70.62**	**74.85**	**87.22**	**89.84**	**90.16**	**92.92**	**79.29**
	w/o Transfer	46.24	68.09	74.41	85.56	88.76	90.10	92.15	77.90
CoNLL04	UIE	22.05	**45.41**	52.39	30.77	**51.72**	59.18	73.48	**47.85**
	TIE (T5-base)	**22.09**	38.08	**52.43**	**31.32**	49.21	**59.28**	**73.58**	46.57
	w/o Transfer	19.02	35.14	52.08	31.12	47.32	59.20	73.49	45.34
ACE05-Evt (trigger)	UIE	38.14	51.21	53.23	41.53	55.70	60.29	71.33	53.06
	TIE (T5-base)	**39.12**	**52.88**	**55.56**	**42.86**	**57.84**	61.20	**73.09**	**54.65**
	w/o Transfer	38.88	52.14	54.94	41.80	56.11	**61.28**	72.85	54.00
ACE05-Evt (argument)	UIE	11.88	27.44	33.64	12.80	30.43	36.28	50.62	29.01
	TIE (T5-base)	**12.31**	**30.63**	**36.36**	**15.75**	**34.73**	**39.42**	**56.71**	**32.27**
	w/o Transfer	11.92	28.56	35.17	14.58	34.38	37.68	55.11	31.05
16-res	UIE	**10.50**	26.24	39.11	24.24	49.31	57.61	72.86	39.98
	TIE (T5-base)	6.860	**27.19**	**39.67**	**24.79**	**50.26**	**58.29**	**75.17**	**40.32**
	w/o Transfer	5.500	25.36	39.27	24.06	48.73	58.20	73.79	39.27

Table 6: Results of few-shot and low-resource scenarios on four datasets.

12 datasets, comparing to six task-specific methods and three unified IE methods (except InstructUIE, which bases on LLM). In contrast to UIE (T5-base), there is an average improvement of 2.09 points. This result demonstrates the effectiveness of our approach. **Second**, *in LLM setting, TIE still holds an edge.* When changing our backbone with LLM FlanT5-3B, we also outperform InstructUIE (FlanT5-11B) on three out of four common datasets. Besides, TIE (FlanT5-3B) achieves new state-of-the-art on almost all datasets. **Third**, *the classification method proves to be more potent than the generative one in IE tasks.* The poor performance of ChatGPT3.5 in zero-shot setting proves that there are limitations in using a conversational generative model for information extraction tasks. Although the parameter scale of InstructUIE is approximately four times larger than ours, we still maintain a lead on most of the tasks, which indicates the great potential of using the classification models for information extraction (Appendix 8.3).

5.2. Ablation Studies

For ablation studies, we experiment on each dataset of different IE tasks and study the effect of different components of our method (Table 5). '- Instruction' removes the decoder part with instructions from TIE. Whereas '- Transfer' means we train our model on target datasets without trans-



(a) CoNLL03 (b) CoNLL04

(c) ACE05-Evt (argument) (d) 16-res

Figure 2: The influence of instruction numbers and the syntactic diversity of instructions.

fer, which means we conduct single-task learning. '- Regularization' stands for the absence of task-specific regularization while retaining transfer.

We have several observations. **First**, *the instructions are most important to* TIE. It decreases by an average of 2.55 F1 without instructions. Despite transfer learning can excavate commonality, it is the instructions that help learn a wealth of common knowledge from various tasks. Instructions inform the model of the label information to provide proper guidance, thereby alleviating the disruption. **Second**, *the gradient-regularization strategy plays a role in resolving inconsistency.* Removing task-specific regularization leads to an average decrease of 1.036 points. However, there is only a decrease of 0.771 F1 averagely when we conduct single-task learning. For the inconsistency and complexity of knowledge, direct transfer learning does not significantly help model perform IE tasks. In order to achieve a balance between tasks, the regularization is indispensable.

5.3. Results on Data-scarce Scenarios

As shown in Table 6, we conduct experiments on 4 datasets of different IE subtasks in data-scarce scenarios. TIE averagely improves the F1 for 2.13, 1.59, 3.26, 0.34, compared to UIE (Lu et al., 2022) on CoNLL03, ACE05-Evt (Trig.F1), ACE05-Evt (Arg.F1) and 16-res, respectively. Similarly, we remove the transfer learning step. We observe that: *It is the transfer learning by gradient-regularization that fosters the effectiveness of the model in data-scarce scenarios.* The effectiveness in data-scarce scenarios demonstrates that the inconsistent knowledge resolved by regularization is negative in IE tasks. Without a large volume of training corpus, TIE can still acquire rich semantic information from labels within instructions. These results reveal that TIE has good generalization performance and is highly sensitive to new data.

5.4. Analyses on Instruction Diversity

To investigate the impact of instructions on model learning, we conduct experiments on four datasets: CoNLL03, CoNLL04, ACE05-Evt (Arg.F1) and 16-res with the following setups:

- First, we investigate the influence of the number of instructions;
- Second, we explore the influence of syntax diversity by customizing instructions with diverse syntactic similarity for each dataset. We generate instructions with varying syntactic similarity using ChatGPT3.5 and score them with GPT4.

The Influence of Instruction Numbers We select instruction with quantities of 0, 1, 5 and 10 (Figure 2). Although TIE excavates commonality and resolves inconsistency via transfer learning, it still struggles when the instruction number is 0. Instructions can provide the model with elaborate task guides and label semantics, so that the decision-making would be more accurate. TIE's performances on four datasets all reach the optimum when the quantity is set to 5. However, the accuracy of our method experiences a decline when employing instructions with a quantity of 10 on each dataset. More instructions could bring noise to the model, thereby decreasing the property.

The Influence of Syntax Diversity of Instructions While building the instruction pool, the prompt '... the rewritten sentences exhibit significant differences in syntax...' is input into ChatGPT3.5, for the purpose that we want to get instructions with high syntactic richness. These rephrased

Dataset	Group1	Group2
CoNLL03	0.88	0.74
CoNLL04	0.81	0.62
ACE05-Evt	0.81	0.72
16-res	0.86	0.74

Table 7: Syntactic richness scored by GPT4 of two instruction groups on 4 IE datasets.

instructions are assigned to **Group1**. As a comparison, we perform partial word replacements in the manual instructions for each dataset, ensuring syntactical consistency. These instructions are allocated to **Group2**.

We utilize the superior LLM GPT4 to score the syntactic richness of these instructions. The score of two groups of instructions on four datasets is illustrated in Table 7. Instructions in Group1 have higher scores than the other group, which means they have higher syntactic richness. So that we can conclude from Figure 2: when the instruction number is 5, more diverse instructions lead to better model performance, while the quantity is 10, instructions possessing a greater syntactic richness could interfere with the model's learning. Nevertheless, similar instructions, whether the quantity is 5 or 10, have a limited impact on model performance. For information on the use of GPT4, please refer to the Appendix 8.4. In this way, more diverse instructions with rich syntax will be used for a specific dataset, which yields better results.

In a nutshell, we choose 5 instructions with higher syntactic richness for each dataset.

6. Conclusion

In this paper, we propose a regularization-based transfer learning method for IE named TIE, applying an instructed graph decoder. It captures the shared common knowledge among tasks while preventing inconsistencies using the instructed graph decoder and the task-specific regularization strategy. Experimental results demonstrate that TIE achieves new state-of-the-art performance on most IE datasets, compared to both task-specific and unified baselines. The ablation studies show the great advantages of the main components contained in TIE. Also, we observe that TIE performs well on data-scarce scenarios. In the future, it would be interesting to explore the effectiveness of our method with large-scale language models such as LLaMa and Vicuna.

Limitations

In this paper, we propose a novel regularization-based transfer learning method for IE (named TIE), whose main component is a specially designed instructed graph decoder. Our method pre-trains on the source datasets and finetunes on the target one. We conduct extensive experiments on 12 datasets spanning four IE tasks, and the results demonstrate the great advantages of our proposed method in both fully supervised and data-scarce scenarios. However, there are still some limitations of our method.

(1) Although the model's structure is quite simple, the entire process of pre-training on the source datasets and finetuning on the target datasets is relatively complex and time-consuming.

(2) Due to resource limitations, we are unable to train the FlanT5-11B model, just as InstructUIE does.

(3) We do not investigate how to construct instructions that cover an open set of options. This is a very valuable area for our future work. And, for each instruction, we need to extract the label slots in the instruction sentences, which also increases the workload.

(4) For fair comparisons, our baseline data is sourced directly from their original papers. In the future, we can test our method on a wider range of models such as LLaMa and Vicuna.

Ethics Statement

The model architecture in this paper, such as the encoder-decoder and biaffine parts, are commonly used deep learning components. All datasets mentioned in this paper are widely used public datasets in information extraction tasks. We annotate the sources of these datasets in Section 4.1, so there are no copyright issues.

All authors are knowledgeable about the research presented in this paper.

The entire process and outcomes are free from intellectual property and ethical legal disputes.

All intellectual property rights for the content of this paper belong to the authors.

Acknowledgement

This research is funded by the National Key Research and Development Program of China (No.2021ZD0114002), the National Natural Science Foundation of China (No.62307028), the Science and Technology Commission of Shanghai Municipality Grant (No.22511105901 and No.21511100402), and Shanghai Science and Technology Innovation Action Plan (No.23ZR1441800 and No.23YF1426100).

We still want to thank Hang Yan for providing us with some processed data. The compulsory code inspection by Jiaju Lin and Zhikai Lei also plays a significant role.

7. Bibliographical References

Jeremy Barnes, Robin Kurtz, Stephan Oepen, Lilja Øvrelid, and Erik Velldal. 2021. Structured sentiment analysis as dependency graph parsing. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 3387–3402. Association for Computational Linguistics.

Parminder Bhatia, Kristjan Arumae, and E. Busra Celikkaya. 2020. Dynamic transfer learning for named entity recognition. In Arash Shaban-Nejad and Martin Michalowski, editors, *Precision Health and Medicine - A Digital Revolution in Healthcare*, volume 843 of *Studies in Computational Intelligence*, pages 69–81. Springer.

Jiawei Chen, Yaojie Lu, Hongyu Lin, Jie Lou, Wei Jia, Dai Dai, Hua Wu, Boxi Cao, Xianpei Han, and Le Sun. 2023. Learning in-context learning for named entity recognition. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13661–13675, Toronto, Canada. Association for Computational Linguistics.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models.

Meiji Cui, Li Li, Zhihong Wang, and Mingyu You. 2017. A survey on relation extraction. In *Knowledge Graph and Semantic Computing. Language, Knowledge, and Intelligence - Second China Conference, CCKS 2017, Chengdu, China, August 26-29, 2017, Revised Selected Papers*, volume 784 of *Communications in Computer and Information Science*, pages 50–58. Springer.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Shimin Di, Yanyan Shen, and Lei Chen. 2019. Relation extraction via domain-aware transfer learning. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*, pages 1348–1357. ACM.

Hai Ha Do, P. W. C. Prasad, Angelika Maag, and Abeer Alsadoon. 2019. Deep learning for aspect-based sentiment analysis: A comparative review. *Expert Syst. Appl.*, 118:272–299.

Ridong Han, Tao Peng, Chaohao Yang, Benyou Wang, Lu Liu, and Xiang Wan. 2023. Is information extraction solved by chatgpt? an analysis of performance, evaluation criteria, robustness and errors. *CoRR*, abs/2305.14450.

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 *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 1890–1908. Association for Computational Linguistics.

Jie Huang and Kevin Chen-Chuan Chang. 2023. Towards reasoning in large language models: A survey. In *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 1049–1065. Association for Computational Linguistics.

Bo Li, Gexiang Fang, Yang Yang, Quansen Wang, Wei Ye, Wen Zhao, and Shikun Zhang. 2023. Evaluating chatgpt's information extraction capabilities: An assessment of performance, explainability, calibration, and faithfulness. *CoRR*, abs/2304.11633.

Lin Li and Michael W. Spratling. 2023. Understanding and combating robust overfitting via input loss landscape analysis and regularization. *Pattern Recognit.*, 136:109229.

Zhen Li, Xiaohan Xu, Tao Shen, Can Xu, Jia-Chen Gu, and Chongyang Tao. 2024. Leveraging large language models for nlg evaluation: A survey.

Sen Lin, Li Yang, Deliang Fan, and Junshan Zhang. 2022. TRGP: trust region gradient projection for continual learning. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

David Lopez-Paz and Marc'Aurelio Ranzato. 2017. Gradient episodic memory for continual learning. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 6467–6476.

Jie Lou, Yaojie Lu, Dai Dai, Wei Jia, Hongyu Lin, Xianpei Han, Le Sun, and Hua Wu. 2023. Universal information extraction as unified semantic matching. In *Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI 2023, Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence, IAAI 2023, Thirteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2023, Washington, DC, USA, February 7-14, 2023*, pages 13318–13326. AAAI Press.

Yaojie Lu, Qing Liu, Dai Dai, Xinyan Xiao, Hongyu Lin, Xianpei Han, Le Sun, and Hua Wu. 2022. Unified structure generation for universal information extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 5755–5772. Association for Computational Linguistics.

Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. 2018. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 3219–3232. Association for Computational Linguistics.

Fukun Ma, Xuming Hu, Aiwei Liu, Yawen Yang, Shuang Li, Philip S. Yu, and Lijie Wen. 2023. AMR-based network for aspect-based sentiment analysis. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 322–337, Toronto, Canada. Association for Computational Linguistics.

Mónica Marrero, Julián Urbano, Sonia Sánchez-Cuadrado, Jorge Morato, and Juan Miguel Gómez-Berbís. 2013. Named entity recognition: Fallacies, challenges and opportunities. *Comput. Stand. Interfaces*, 35(5):482–489.

Alexis Mitchell, Stephanie Strassel, Shudong Huang, and Ramez Zakhary. 2005. Ace 2004 multilingual training corpus. *Linguistic Data Consortium, Philadelphia*, 1:1–1.

Giovanni Paolini, Ben Athiwaratkun, Jason Krone, Jie Ma, Alessandro Achille, Rishita Anubhai, Cícero Nogueira dos Santos, Bing Xiang, and Stefano Soatto. 2021. Structured prediction as translation between augmented natural languages. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Tianshuo Peng, Zuchao Li, Lefei Zhang, Bo Du, and Hai Zhao. 2023. FSUIE: A novel fuzzy span mechanism for universal information extraction. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 16318–16333. Association for Computational Linguistics.

Yang Ping, Junyu Lu, Ruyi Gan, Junjie Wang, Yuxiang Zhang, Pingjian Zhang, and Jiaxing Zhang. 2023. Uniex: An effective and efficient framework for unified information extraction via a span-extractive perspective. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 16424–16440. Association for Computational Linguistics.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad Al-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, Véronique Hoste, Marianna Apidianaki, Xavier Tannier, Natalia V. Loukachevitch, Evgeniy V. Kotelnikov, Núria Bel, Salud María Jiménez Zafra, and Gülsen Eryigit. 2016. Semeval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2016, San Diego, CA, USA, June 16-17, 2016*, pages 19–30. The Association for Computer Linguistics.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. Semeval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2015, Denver, Colorado, USA, June 4-5, 2015*, pages 486–495. The Association for Computer Linguistics.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation, SemEval@COLING 2014, Dublin, Ireland, August 23-24, 2014*, pages 27–35. The Association for Computer Linguistics.

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. *Journal of Machine Learning Research*, 21(140):1–67.

Dan Roth and Wen-tau Yih. 2004. A linear programming formulation for global inference in natural language tasks. In *Proceedings of the Eighth Conference on Computational Natural Language Learning, CoNLL 2004, Held in cooperation with HLT-NAACL 2004, Boston, Massachusetts, USA, May 6-7, 2004*, pages 1–8. ACL.

Gobinda Saha, Isha Garg, and Kaushik Roy. 2021. Gradient projection memory for continual learning. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Erik F. 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, CoNLL 2003, Held in cooperation with HLT-NAACL 2003, Edmonton, Canada, May 31 - June 1, 2003*, pages 142–147. ACL.

Taneeya Satyapanich, Francis Ferraro, and Tim Finin. 2020. CASIE: extracting cybersecurity event information from text. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8749–8757. AAAI Press.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

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 2019, Hong Kong, China, November 3-7, 2019*, pages 5783–5788. Association for Computational Linguistics.

Somin Wadhwa, Silvio Amir, and Byron Wallace. 2023. Revisiting relation extraction in the era of large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15566–15589, Toronto, Canada. Association for Computational Linguistics.

Christopher Walker, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. 2006. Ace 2005 multilingual training corpus. *Linguistic Data Consortium, Philadelphia*, 57:45.

Juncheng Wan, Dongyu Ru, Weinan Zhang, and Yong Yu. 2022. Nested named entity recognition with span-level graphs. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 892–903. Association for Computational Linguistics.

Jiaan Wang, Yunlong Liang, Fandong Meng, Zengkui Sun, Haoxiang Shi, Zhixu Li, Jinan Xu, Jianfeng Qu, and Jie Zhou. 2023a. Is ChatGPT a good NLG evaluator? a preliminary study. In *Proceedings of the 4th New Frontiers in Summarization Workshop*, pages 1–11, Singapore. Association for Computational Linguistics.

Xiao Wang, Weikang Zhou, Can Zu, Han Xia, Tianze Chen, Yuansen Zhang, Rui Zheng, Junjie Ye, Qi Zhang, Tao Gui, Jihua Kang, Jingsheng Yang, Siyuan Li, and Chunsai Du. 2023b. Instructuie: Multi-task instruction tuning for unified information extraction. *CoRR*, abs/2304.08085.

Xiang Wei, Xingyu Cui, Ning Cheng, Xiaobin Wang, Xin Zhang, Shen Huang, Pengjun Xie, Jinan Xu, Yufeng Chen, Meishan Zhang, Yong Jiang, and Wenjuan Han. 2023. Zero-shot information extraction via chatting with chatgpt. *CoRR*, abs/2302.10205.

Lu Xu, Yew Ken Chia, and Lidong Bing. 2021. Learning span-level interactions for aspect sentiment triplet extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 4755–4766. Association for Computational Linguistics.

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, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021, pages 5808–5822. Association for Computational Linguistics.

Hang Yan, Yu Sun, Xiaonan Li, Yunhua Zhou, Xuanjing Huang, and Xipeng Qiu. 2023. UTC-IE: A unified token-pair classification architecture for information extraction. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 4096–4122. Association for Computational Linguistics.

Huiling You, Lilja Vrelid, and Samia Touileb. 2023. JSEEGraph: Joint structured event extraction as graph parsing. In *Proceedings of the 12th Joint Conference on Lexical and Computational Semantics (*SEM 2023)*, pages 115–127, Toronto, Canada. Association for Computational Linguistics.

Yice Zhang, Yifan Yang, Yihui Li, Bin Liang, Shiwei Chen, Yixue Dang, Min Yang, and Ruifeng Xu. 2022. Boundary-driven table-filling for aspect sentiment triplet extraction. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 6485–6498. Association for Computational Linguistics.

Zexuan Zhong and Danqi Chen. 2021. A frustratingly easy approach for entity and relation extraction. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 50–61. Association for Computational Linguistics.

Jie Zhou, Qi Zhang, Qin Chen, Liang He, and Xuanjing Huang. 2022. A multi-format transfer learning model for event argument extraction via variational information bottleneck. In *Proceedings of the 29th International Conference on Computational Linguistics, COLING 2022, Gyeongju, Republic of Korea, October 12-17, 2022*, pages 1990–2000. International Committee on Computational Linguistics.

Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. 2021. A comprehensive survey on transfer learning. *Proc. IEEE*, 109(1):43–76.

8. Appendices

8.1. The Construction of Instruction Pool

As described in Section 3.2 (Instruction Pool), we manually write an instruction for each dataset as the seed, such as ”Annotate the polarity (positive, negative or neutral), expression, aspect of the sentence.” for 16-res dataset.

Then we use the prompt ”Please rewrite the following sentence several times and make sure the rewritten sentences exhibit significant differences in syntax, compared to the original sentence: Annotate the polarity (positive, negative or neutral), expression, aspect of the sentence.” to augment the manual instruction.

Regarding to the ChatGPT3.5 settings, we utilize ChatGPT (gpt-3.5-turbo)[7] with four different temperatures ranging from 0.1 to 0.4. For each temperature, we generate several instructions and manually select one instruction with significant syntactic differences from the results. This process is repeated for each temperature, resulting in a total number of five instructions (including the seed) for each dataset.

8.2. More Examples of the Instructions

In this section, we will display all instructions for the 14-res, 14-lap, 15-res and 16-res datasets (the ABSA task) below. All other instructions can be seen in our github repository.

1. Annotate the polarity (positive, negative or neutral), expression, aspect of the sentence.
2. Get the polarity (whether positive or negative or neutral) of the sentence, and the distinct expression and aspect of this statement.
3. Retrieve the emotional tone (positive, negative, or neutral) of the sentence and identify the corresponding content as either expression text, aspect text.
4. Find the sentiment (positive, negative, or neutral) of the sentence and identify the expression, aspect element.
5. Determine whether the sentiment of this sentence is positive, negative, or neutral and pinpoint the specific expression and aspect.

8.3. Explanations about the Comparison between Generation and Classification

ChatGPT3.5 and InstructUIE are generative models, which generate a sequence as the extraction result directly. In contrast, our method is a classification one, which predicts the probability of each

[7] https://openai.com/blog/gpt-3-5-turbo-fine-tuning-and-api-updates

label for each position as formulated in equation (4)-(7), and utilizes the binary cross-entropy (BCE) loss to optimize the model in equation (9).

This claim suggests that despite having significantly fewer parameters compared to these baselines as ChatGPT3.5 and InstructUIE, our method still achieves good performance on most datasets, which indicates the great potential of using the classification models for information extraction.

8.4. The Details on the Use of GPT4

Inspired by the effectiveness of using GPT4[8] in NLG evaluation in previous studies (Wang et al., 2023a; Li et al., 2024), we also utilize it to better quantify the syntax diversity of instructions. Specifically, we first use the Spacy[9] library to obtain the syntactic parse trees for the instructions, and then integrate the parsing results into the prompt for scoring. - We will provide more details of using GPT4 for scoring the syntax diversity, including the prompt settings and the previously founded studies. We use the following prompt.

Here are two sentences:

1.**[sentence1]** 2.**[sentence2]**

Here are the two syntactic parse trees of the sentences:

1.**[tree1]** 2.**[tree2]**

Assign a score for syntactic diversity for the two sentences on a scale of 0 to 1, where 0 is the lowest and 1 is the highest based on the Evaluation Criteria.

Evaluation Criteria: In the syntactic tree, each triple consists of the first element representing the word in the original text, the second element representing the headword on which the word depends, and the third element representing the dependency relationship between them. Syntactic Diversity (0-1) - The richness of syntax between two sentences. If two sentences express the same meaning semantically but have different dependency relationships in their syntactic structures, the higher the score, the greater the difference in dependency relationships.

---

[8]https://openai.com/gpt-4

[9]https://spacy.io/