UniEX: An Effective and Efficient Framework for Unified Information Extraction via a Span-extractive Perspective

Ping Yang[1*] **Junyu Lu**[12*] **Ruyi Gan**[1*] **Junjie Wang**[3] **Yuxiang Zhang**[3]
Jiaxing Zhang[1†] **Pingjian Zhang**[2†]
[1]International Digital Economy Academy [2]South China University of Technology
[3]Waseda University
{yangping, lujunyu, ganruyi}@idea.edu.cn
wjj1020181822@toki.waseda.jp, joel0495@asagi.waseda.jp
pjzhang@scut.edu.cn

Abstract

We propose a new paradigm for universal information extraction (IE) that is compatible with any schema format and applicable to a list of IE tasks, such as named entity recognition, relation extraction, event extraction and sentiment analysis. Our approach converts the text-based IE tasks as the token-pair problem, which uniformly disassembles all extraction targets into joint span detection, classification and association problems with a unified extractive framework, namely UniEX. UniEX can synchronously encode schema-based prompt and textual information, and collaboratively learn the generalized knowledge from pre-defined information using the auto-encoder language models. We develop a traffine attention mechanism to integrate heterogeneous factors including tasks, labels and inside tokens, and obtain the extraction target via a scoring matrix. Experiment results show that UniEX can outperform generative universal IE models in terms of performance and inference-speed on 14 benchmarks IE datasets with the supervised setting. The state-of-the-art performance in low-resource scenarios also verifies the transferability and effectiveness of UniEX.

1 Introduction

Information extraction (IE) aims at automatically extracting structured information from unstructured textual sources, covering a wide range of subtasks such as named entity recognition, relation extraction, semantic role labeling, and sentiment analysis (Muslea et al., 1999; Grishman, 2019). However, the variety of subtasks build the isolation zones between each other and form their own dedicated models. Fig 1 (a) presents that the popular IE approaches handle structured extraction by the addition of task-specific layers on top of pre-trained language models (LMs) and a subsequent

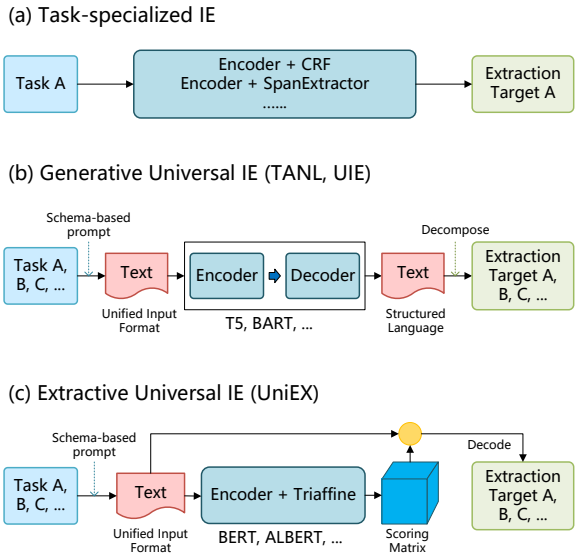

Figure 1: (a) Task-specific IE methods: isolated structures and schemas. (b) Typical generative universal IE: unified modeling via text or structure generation. (c) Our extractive universal IE: unified modeling via traffine attention mechanism and auto-encoder LMs.

fine-tuning of the conjoined model (Lample et al., 2016; Luo et al., 2020; Wei et al., 2020; Ye et al., 2022). The isolated architectures and chaotic situation prevents enhancements from one task from being applied to another, which hinders the effective latent semantics sharing such as label names, and suffer from inductive bias in transfer learning (Paolini et al., 2020).

With powerful capabilities in knowledge sharing and semantic generalization, large-scale LMs bring the opportunity to handle multiple IE tasks using a single framework. As shown in Fig 1 (b), by developing sophisticated schema-based prompt and structural generation specification, the IE tasks can be transformed into text-to-text and text-to-structure formats via large-scale generative LMs (Dong et al., 2019; Paolini et al., 2020; Lu et al., 2022) such as T5 (Raffel et al., 2020a). Moreover, the universal IE frameworks can learn general knowledge from multi-source prompts, which is

*Equal Contribution.
†Corresponding Author.

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 16424–16440

beneficial for perceiving unseen content in low-resource scenarios. Despite their success, these generative frameworks suffer from their inherent problems, which limit their potential and performance in universal modeling. Firstly, the schema-based prompt and contextual information are synthetically encoded for generating the target structure, which is not conducive to directly leveraging the position information among different tokens. Secondly, the generative architecture utilizes the token-wise decoder to obtain the target structure, which is extremely time-consuming.

The aforementioned issues prompt us to rethink the foundation of IE tasks. Fundamentally, we discover that the extraction targets of different IE tasks involve the determination of semantic roles and semantic types, both of which can be converted into span formats by the correlation of the inside tokens in the passage. For instance, an entity type is the boundary detection and label classification of a semantic role, while a relation type can be regarded as the semantic association between specific semantic roles. From this perspective, the IE tasks can be decoded using a span-extractive framework, which can be uniformly decomposed as several atomic operations: i) Span Detection, which locates the boundaries of the mentioned semantic roles; ii) Span Classification, which recognizes the semantic types of the semantic roles; iii) Span Association, which establishes and measures the correlation between semantic roles to determine semantic types. According to the above observation, we propose a new paradigm for universal IE, called **Uni**fied **Ex**traction model (UniEX) as Figure 1 (c). Specifically, we first introduce a rule-based transformation to bridge various extraction targets and unified input formats, which leverages task-specific labels with identifiers as the schema-based prompt to learn general IE knowledge. Then, recent works (Liu et al., 2019a; Yang et al., 2022) state that the auto-encoder LMs with bidirectional context representations are more suitable for natural language understanding. Therefore, We employ BERT-like LMs to construct an extractive architecture for underlying semantic encoding. Finally, inspired by the successful application of span-decoder and biaffine network to decode entity and relation with a scoring matrix (Yu et al., 2020b; Li et al., 2020; Yuan et al., 2022), we introduce a triaffine attention mechanism for structural decoding, which jointly considers high-order interactions among multiple factors, including tasks, labels and inside tokens. Each triaffine scoring matrix is assigned to a demand-specific prompt for obtaining span-extractive objectives.

Through extensive experiments on several challenging benchmarks of 4 main IE tasks (entity/relation/event/sentiment extraction), we demonstrate that compared with the state-of-the-art universal IE models and task-specific low-resource approaches, our UniEX achieves a substantial improvement in performance and efficiency with supervised, few-shot and zero-shot settings.

Our main contributions are summarized as:

- We develop an efficient and effective universal IE paradigm by converting all IE tasks into joint span classification, detection and association problem.
- We introduce UniEX, a new unified extractive framework that utilizes the extractive structures to encode the underlying information and control the schema-based span decoding via the triaffine attention mechanism.
- We apply our approach in low-resource scenarios, and significant performance improvements suggest that our approach is potential for attaching label information to generalized objects and transfer learning. Our code will be made publicly available.

2 Related Work

Unified NLP Task Formats Since the prompt-tuning can improve the ability of language models to learn common knowledge and fix the gap across different NLP tasks, recent studies show the necessity of unifying all NLP tasks in the format of a natural language response to natural language input (Raffel et al., 2020b; Sanh et al., 2022; Wei et al., 2021). Previous unified frameworks usually cast parts of text problems as question answering (McCann et al., 2018) or span extraction (Keskar et al., 2019) tasks. TANL (Paolini et al., 2020) frames the structured prediction tasks as a translation task between augmented natural languages. By developing a text-to-text architecture, T5 (Raffel et al., 2020b) makes prompts to effectively distinguish different tasks and provide prior knowledge for multitask learning. UIE (Lu et al., 2022) uniformly models IE tasks with a text-to-structure framework, which encodes different

extraction structures via a structured extraction language, adaptively generates varying targets via a structural schema instructor. Although effective, such methods focus on generative styles and thus cannot be adapted to the knowledge selection for vast label-based models. It motivates us to design an efficient and effective universal IE method, where we develop unified Extraction (EX) formats and triaffine attention mechanism.

Label Information Label semantics is an important information source, which carries out the related meaning induced from the data (Hou et al., 2020; Ma et al., 2022a; Mueller et al., 2022). The L-TapNet (Hou et al., 2020) introduces the collapsed dependency transfer mechanism to leverage the semantics of label names for few-shot tagging tasks. LSAP (Mueller et al., 2022) improves the generalization and data efficiency of few-shot text classification by incorporating label semantics into the pre-training and fine-tuning phases of generative LMs. Together, these successful employments of label knowledge in low-resource setting motivates us to introduce label semantics into our unified inputs to handle few-shot and zero-shot scenarios.

3 Approaches

Generally, there are two main challenges in universally modeling different IE tasks via the extractive architecture. Firstly, IE tasks are usually demand-driven, indicating that each pre-defined schema should correspond to the extraction of specific structural information. Secondly, due to the diversity of IE tasks, we need to resolve appropriate structural formats from the output sequence to accommodate different target structures, such as entity, relation and event. In this section, we outline how the UniEX exploits a shared underlying semantic encoder to learn the prompt and text knowledge jointly, and conduct various IE tasks in a unified text-to-structure architecture via the triaffine attention mechanism.

3.1 The UniEX Framework

3.1.1 Unified Input

Formally, given the task-specific pre-defined schema and texts, the universal IE model needs to adaptively capture the corresponding structural information from the text indicated by the task-relevant information. To achieve this, we formulate a unified input format consisting of task-relevant schema and text, as shown in Figure 2. To promote the sharing of generalized knowledge across different IE tasks, we choose to simply use the task-based and label-based schemas as prompt rather than elaborate questions, fill-in blanks or structural indicators. To achieve proper prompt representation, we introduce several special tokens `[D-TOK]`, `[C-TOK]` and `[A-TOK]` as identifiers, uniformly replacing the corresponding schema representations in the input sentence. Here, `[D-TOK]` inherits the ability of `[CLS]` to capture the global semantic information. `[C-TOK]` and `[A-TOK]` inherit the ability of `[SEP]`, thus remaining to use token representation to symbolize the connotation of subsequent schemas. Consider an input set denoted as (s, x), includes the following: i) task-based schema s_d for span detection, ii) label-based schemas s_c for span classification and s_a for span association, iii) one passage $x = \{x_1, \ldots, x_{N_x}\}$. The input sentence with $N_s = N_{sd} + N_{sc} + N_{sa}$ schemas and N_x inside tokens can be denoted as:

$$x_{inp} = \Big\{[\,\text{D-TOK}]^i\ s_d^i\Big\}_{i=1}^{N_{sd}} \Big\{[\,\text{C-TOK}]^i\ s_c^i\Big\}_{i=1}^{N_{sc}} \Big\{[\,\text{A-TOK}]^i\ s_a^i\Big\}_{i=1}^{N_{sa}} [\text{SEP}]\ x\ [\text{SEP}]. \quad (1)$$

3.1.2 Backbone Network

In our UniEX framework, we employ the BERT-like LMs as the extractive backbone, such as RoBERTa (Liu et al., 2019b) and ALBERT (Lan et al., 2020), to integrate the bidirectional modeled input x_{inp}. Note that the unified input contains multiple labels, resulting in undesired mutual influence across different labels and leading to a misunderstanding of the correspondence between the label and its structural format during the decoding phase. Meanwhile, in some tasks, the large number of labels allows schemas to take up excessive locations, squeezing the space for text. Referring to the embedding methods in the UniMC (Yang et al., 2022), we address these issues from several perspectives, including position id and attention mask. Firstly, to avoid the information interference caused by the mutual interaction within label-based schemas, we constantly update the position id pos to tell apart intra-information in the label. In this way, the position information of label-relevant tokens is coequally treated based on their position embedding, and the refreshed location information for the first token of each label-based schema avoids the natural increase of the location id. Then, as shown in Figure 3, due to the detailed correlation among schema-based prompts in the IE tasks, we further

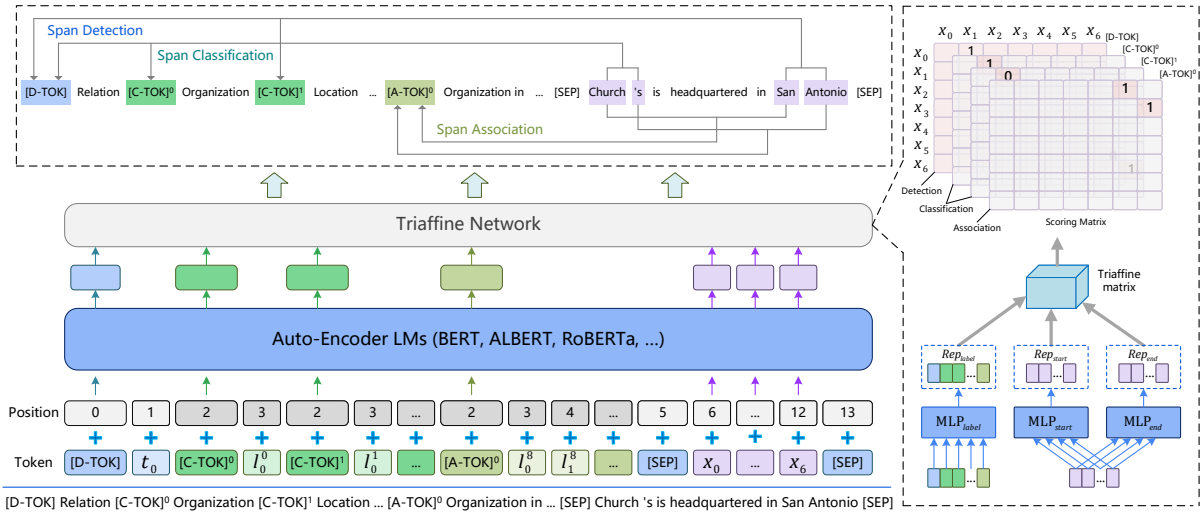

Figure 2: The overall architecture of UniEX. The sample text comes from CoNLL04 (Roth and Yih, 2004).

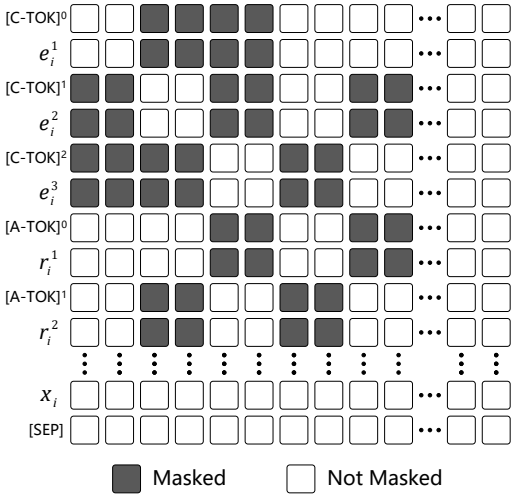

Figure 3: Schema-based Attention Mask Matrix of the relation extraction task with triplet type (e^1, r^1, e^2) and (e^1, r^2, e^3). The relation and entity types are internally invisible, whereas the paired relation and entity types can attend to each other.

introduce a schema-based attention mask matrix M_{mask} in the self-attention calculation to control the flow of labels, ensuring that unrelated labels are invisible to each other. In particular, different entity, relation and event types are invisible to each other, while relation and event types can contact their bound entity types.

Furthermore, we take the encoded hidden vector from the last Transformer-based layer, where we combine the special tokens part as the schema representations $H_s \in \mathbb{R}^{N_s \times d}$ and the passage tokens part as the text representations $H_x \in \mathbb{R}^{N_x \times d}$ with hidden size d.

$$H_s, H_x = \text{Encoder}\left(x_{inp}, pos, M_{\text{mask}}\right) \quad (2)$$

3.1.3 Triaffine Attention for Span Representation

After obtaining the schema representations and text representations from the auto-encoder LM, the following challenge is *how to construct a unified decoding format that is compatible with different IE structures, with the goal of adaptively exploiting schemas to control various extraction targets.* Take the example in Figure 4, for the event extraction system, we locate the start and end indices of the words boundary "Dariues", "Ferguson" and "injure" as the semantic roles, categorized as the *Agent*, *Victim* and *Trigger* semantic types (entity/trigger) respectively, and collectively to the *Injure* semantic type (event). For the relation extraction system, we associate the semantic roles "Betsy Ross" and "Philadelphia" by attaching their intersecting information to the *Live in* semantic type (relation). In conjunction with the discussions in the Introduction, we consider two elements for universally modeling IE tasks as joint span detection, classification and association: I) Different extraction targets are presented in the form of span, relying on unified information carriers to accommodate various semantic roles and semantic types. II) The span-extractive architecture is necessary for establishing schema-to-text information interaction, which can adaptively extract schema-related semantic information from text.

For the first proposal, we introduce two information carriers for decoding heterogeneous IE structures in a unified span format:

1. **Structural Table** indicates a rank-2 scoring matrix corresponding to a particular schema, which accommodates the semantic information required

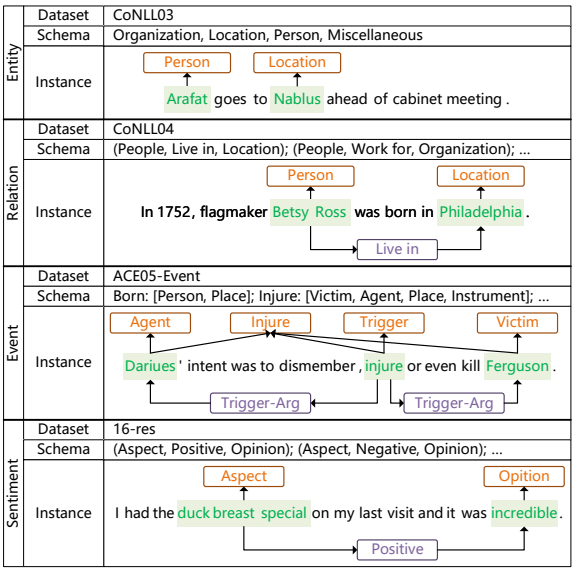

Figure 4: Uniformly modeling different extraction targets as joint span detection, classification and association with sampling from selected datasets.

for span-extractive parsing.

2. **Spotting Designator** indicates the location of spans in the preceding structural table, which represent extraction targets corresponding to the particular schema.

For the second proposal, we attempt to explore the internal interaction of the inside tokens by converting the text representation into span representation. Then, we apply two separate FFNs to create different representations (H_x^s / H_x^e) for the start/end positions of the inside tokens. To further interact such multiple heterogeneous factors simultaneously, we define the deep triaffine transformation with weighted matrix $\mathcal{W} \in \mathbb{R}^{d\times d\times d}$, which apply the triaffine attention to aggregate the schema-wise span representations by considering schema as queries as well as start/end of the inside tokens as keys and values. In this process, the triaffine transformation injects each schema information into the span representations and resolves the corresponding extraction targets. It creates a $N_s \times N_x \times N_x$ scoring tensor S by calculating continuous matrix multiplication as following:

$$\begin{aligned} H_x^s &= \mathrm{FFN}_s\left(H_x\right), \\ H_x^e &= \mathrm{FFN}_e\left(H_x\right), \\ S &= \sigma(\mathcal{W} \times_1 H_s \times_2 H_x^s \times_3 H_x^e), \end{aligned} \tag{3}$$

where $\times_k$ is the matrix multiplication between input tensor and dimension-k of $\mathcal{W}$. $\sigma(*)$ denotes the Sigmoid activation function.

At this point, the tensor S provides a mapping score from the schema to internal spans of the text, where each rank-2 scoring matrix corresponding to a specific schema is the structural table. For the r-th structural table, the affine score of each span (p, q) that starts with p and ends with q can be denoted as $S_{r,p,q} \in [0, 1]$, while the affine score of a valid span in the structural table is the spotting designator. We divide all N_s structural tables into three parts according to the distribution of the schemas, among them, N_{sd} for span detection, N_{sc} for span classification, and N_{sa} for span association. For different schemas, we develop their spotting designators by following strategies:

Span Detection: In particular, we usually use the structural table derived from the task-based schema representation for span detection, which can be obtained from the hidden state of the special token `[CLS]`. Since the `[CLS]` token is mutually visible to other schemas, the task-based schema representation can capture the span-related semantic information of the semantic roles from the task and label names. The spotting designators identify the start and end indices of the i-th semantic roles as (s_i, e_i) using the axes.

Span Classification: The label-based schema representations for entity/argument/trigger/event types are used for span classification. The spotting designators are identical with the span positions of the semantic roles, indicating that the semantic type of the i-th span can be identified by attaching to the (s_i, e_i) position in the corresponding structural table.

Span Association: The label-based schema representations for relation/sentiment types are used for span association. In this process, we model the potentially related semantic roles and correlate them to corresponding semantic types. The spotting designators locate at two interleaved positions associated with the semantic roles of the semantic type, that is, for the i-th and j-th spans, the extraction target is transformed to the identification of the (s_i, s_j) and (e_i, e_j) positions in the corresponding structural table.

Note that all span values in the structural table for label-based schemas are masked except for the spotting designators, because we only need to observe the semantic types and semantic association among the detected spans. Specifically, the spotting designators for span detection are the spans with $q \geq p$, and the spotting designators for span classification and association are defined by the position consistency and interleaving of valid spans

with $S_{r,p,q} = 1$ in span detection.

3.2 EX Training Procedure

Given the input sentence x_{inp}, We uniformly reformat different output targets as a rank-3 matrix Y, sharing the same spotting designators as the triaffine scoring matrix. Similarly, we denote the value of each valid span as $Y_{r,p,q} \in \{0, 1\}$, with $Y_{r,p,q} = 1$ denoting the desirable span for a ground-truth and $Y_{r,p,q} = 0$ denoting the meaningless span for semantic role or semantic type. Hence it is a binary classification problem and we optimize our models with binary cross-entropy:

$$\mathrm{BCE}(y, \hat{y}) = -(y \cdot \log(\hat{y}) + (1 - y) \cdot \log(1 - \hat{y})), \quad (4)$$

$$\mathcal{L} = \sum_{r=1}^{N_s} \sum_{p=1}^{N_x} \sum_{q=1}^{N_x} \mathrm{BCE}\left(Y_{r,p,q}, S_{r,p,q}\right). \quad (5)$$

4 Experiments

To verify the effectiveness of our UniEX, we conduct extensive experiments on different IE tasks with supervised (high-resource), few-shot and zero-shot (low-resource) scenarios.

4.1 Experimental Setup

For the supervised setting, we follow the preparation in TANL (Paolini et al., 2020) and UIE (Lu et al., 2022) to collect 14 publicly available IE benchmark datasets and cluster the well-representative IE tasks into 4 groups, including entity, relation, event and structured sentiment extraction. In particular, for each group, we design a corresponding conversion regulation to translate raw data into the unified EX format.

Then, for the few-shot setting, we adopt the popular datasets FewNERD (Ding et al., 2021) and Cross-Dataset (Hou et al., 2020) in few-shot entity extraction and domain partition as (Ma et al., 2022b). For the zero-shot setting, we use the common zero-shot relation extraction datasets WikiZSL (Chen and Li, 2021) and FewRel (Han et al., 2018) and follow the same process of data and label splitting as (Chia et al., 2022). Following the same evaluation metrics as all previous methods, we use span-based offset Micro-F1 with strict match criteria as the primary metric for performance comparison. Please refer to Appendix A for more details on dataset descriptions, unified EX input formats, metrics and training implementation.

4.2 Experiments on Supervised Settings

In our experiment, under the high-resource scenario, we compare our approach with the state-of-the-art generative universal IE architectures that provide a universal backbone for IE tasks based on T5 (Raffel et al., 2020a), including TANL (Paolini et al., 2020) and UIE (Lu et al., 2022). For a fair comparison, We only consider results without exploiting large-scale contexts and external knowledge beyond the dataset-specific information, and present the average outcomes if the baseline is conducted in multiple runs. The main results of UniEX and other baselines on 14 IE datasets are shown in Table 1. We can observe that: 1) By modeling IE as joint span detection, classification and association, and encoding the schema-based prompt and input texts with the triaffine attention mechanism, UniEX provides an effective universal extractive backbone for all IE tasks. The UniEX outperforms the universal IE models with approximate backbone sizes, achieving new state-of-the-art performance on almost all tasks and datasets. 2) The introduction of label-based schema facilitates the model learning task-relevant knowledge, while the triaffine scoring matrix establishes the correspondence between each schema and extraction targets. Obviously, the UniEX can better capture and share label semantics than using generative structures to encode underlying information. Meanwhile, triaffine transformation is a unified and cross-task adaptive operation, precisely controlling where to detect and which to associate in all IE tasks. Compared with the TANL and UIE, our approach achieves significant performance improvement on most datasets, with nearly 1.36% and 1.52% F1 on average, respectively.

4.3 Experiments on Low-resource Scenarios

To verify the generalization and transferability of UniEX in low-resource scenarios, we evaluate models under few-shot and zero-shot settings, respectively. In order to reduce the influence of noise caused by random sampling on the experiment results, we repeat the data/label selection processes for five different random seeds and report the averaged experiment results as previous works (Hou et al., 2020; Chia et al., 2022). We use the BERT-base (Devlin et al., 2019) as the UniEX backbone to align with other low-resource results.

Firstly, we compare the UniEX with the competitive few-shot entity extraction models. For FewNERD, we compare the proposed approach to De-

Task	Dataset	Domain	Metric	TANL 220M	UniEX 132M	UIE 770M	UniEX 372M
Entity Extraction	ACE04	News, Speech	Entity F1	-	-	86.52	**87.12**
	ACE05-Ent	News, Speech	Entity F1	84.90	**85.96**	85.52	**87.02**
	CoNLL03	News	Entity F1	91.70	92.13	92.17	**92.65**
	GENIA	Biology	Entity F1	76.40	**76.69**	-	-
Relation Extraction	ACE05-Rel	News, Speech	Relation Strict F1	**63.70**	63.64	64.68	**66.06**
	CoNLL04	News	Relation Strict F1	71.40	**71.79**	73.07	**73.40**
	SciERC	Scientific	Relation Strict F1	-	-	33.36	**38.00**
	ADE	Medicine	Relation Strict F1	80.60	83.81	-	-
Event Extraction	ACE05-Evt	News, Speech	Event Trigger F1	68.40	**70.86**	72.63	**74.08**
			Event Argument F1	47.60	**50.67**	**54.67**	53.92
	CASIE	Cybersecurity	Event Trigger F1	-	-	68.98	**71.46**
			Event Argument F1	-	-	60.37	**62.91**
Sentiment Extraction	14-res	Review	Sentiment Triplet F1	-	-	73.78	**74.77**
	14-lap	Review	Sentiment Triplet F1	-	-	63.15	**65.23**
	15-res	Review	Sentiment Triplet F1	-	-	66.10	**68.58**
	16-res	Review	Sentiment Triplet F1	-	-	73.87	**76.02**

Table 1: Overall results of universal IE approaches on different datasets for entity/relation/event/sentiment extraction tasks. **Base** refers to TANL and UniEX respectively using T5-base and RoBERTa-base as the backbone. **Large** refers to UIE and UniEX respectively using T5-large and RoBERTa-large as the backbone.

Models	Intra				Inter			
	1∼2-shot		5∼10-shot		1∼2-shot		5∼10-shot	
	5 way	10 way	5 way	10 way	5 way	10 way	5 way	10 way
ProtoBERT†	23.45±0.92	19.76±0.59	41.93±0.55	34.61±0.59	44.44±0.11	39.09±0.87	58.80±1.42	53.97±0.38
NNShot†	31.01±1.21	21.88±0.23	35.74±2.36	27.67±1.06	54.29±0.40	46.98±1.96	50.56±3.33	50.00±0.36
ESD	41.44±1.16	32.29±1.10	50.68±0.94	42.92±0.75	66.46±0.49	59.95±0.69	**74.14±0.80**	67.91±1.41
DecomMeta	52.04±0.44	43.50±0.59	63.23±0.45	**56.84±0.14**	68.77±0.24	63.26±0.40	71.62±0.16	68.32±0.10
UniEX	**53.92±0.39**	**45.67±0.53**	**63.26±0.14**	56.65±0.27	**69.37±0.19**	**64.53±0.05**	73.79±0.32	**69.63±0.45**

Table 2: F1 scores with standard deviations on FewNERD. † denotes the results reported from Ding et al. (2021).

comMeta (Ma et al., 2022b), ESD (Wang et al., 2022), and methods from (Ding et al., 2021), e.g., ProtoBERT, NNShot. For Cross-Dataset, we compare the UniEX to DecomMeta (Ma et al., 2022b) and baselines reported by (Hou et al., 2020), e.g., TransferBERT, Matching Network, ProtoBERT and L-TapNet+CDT.

Table 2 and 3 illustrates the main results on FewNERD and Cross-Dataset of our approach alongside those reported by previous methods. It can be seen that UniEX achieves the best performance under different type granularity and domain divisions, and outperforms the prior methods with a large margin. Compare with DecomMeta on Cross-Dataset, UniEX achieves a performance improvement up to 6.94% and 5.63% F1 scores on average in 1-shot and 5-shot, which demonstrates the effectiveness of our approach in learning general IE knowledge. It indicates that even without pre-training on large-scale corpus, our approach can still sufficiently excavate the semantic information related with objective entities from label names, which enhances the understanding of task-specific information when data is extremely scarce.

Secondly, we compare UniEX with the latest baselines TableSequence (Wang and Lu, 2020) and RelationPrompt (Chia et al., 2022) on zero-shot relation triplet extraction task for Wiki-ZSL and Few-Rel datasets in Table 4. In both single-triplet and multi-triplet evaluation, UniEX consistently outperforms the baseline models in terms of Accuracy and overall F1 score respectively, which demonstrates the ability of our approach to handle unseen labels. Although we observe a lack of advantage in recall score for multi-triplet evaluation, the significant improvement in precision allowed our approach to achieve a balanced precision-recall ratio. The reason for such difference is probably

Models	1-shot				5-shot			
	News	Wiki	Social	Mixed	News	Wiki	Social	Mixed
TransferBERT‡	4.75±1.42	0.57±0.32	2.71±0.72	3.46±0.54	15.36±2.81	3.62±0.57	11.08±0.57	35.49±7.60
Matching Network‡	19.50±0.35	4.73±0.16	17.23±2.75	15.06±1.61	19.85±0.74	5.58±0.23	6.61±1.75	8.08±0.47
ProtoBERT‡	32.49±2.01	3.89±0.24	10.68±1.40	6.67±0.46	50.06±1.57	9.54±0.44	17.26±2.65	13.59±1.61
L-TapNet+CDT‡	44.30±3.15	12.04±0.65	20.80±1.06	15.17±1.25	45.35±2.67	11.65±2.34	23.30±2.80	20.95±2.81
DecomMeta	46.09±0.44	17.54±0.98	25.14±0.24	34.13±0.92	58.18±0.87	**31.36±0.91**	31.02±1.28	45.55±0.90
UniEX	**58.51±0.14**	**18.20±0.45**	**34.67±0.25**	**39.28±0.55**	**66.08±0.42**	29.68±0.32	**38.64±1.29**	**54.25±0.35**

Table 3: F1 scores with standard deviations on Cross-Dataset. ‡ denotes the results reported from Hou et al. (2020).

Dataset	Model	Single-Triplet	Multi-Triplet		
		Acc.	*P.*	*R.*	*F1*
Wiki-ZSL	TableSequence	14.47	43.68	3.51	6.29
	RelationPrompt	16.64	29.11	**31.00**	30.01
	UniEX	**26.84**	**58.22**	25.85	**34.94**
FewRel	TableSequence	11.82	15.23	1.91	3.40
	RelationPrompt	22.27	20.80	**24.32**	22.34
	UniEX	**27.30**	**44.46**	15.72	**23.13**

Table 4: Result for zero-shot relation triplet extraction under the setting of unseen label set size $m = 5$. We use the Micro-F1, Precision (P.) and Recall (R.) to evaluate the multiple triplet extraction. Evaluating single triplet extraction involves only one possible triplet for each sentence, hence we only use the Accuracy (Acc.) metric.

Dataset	CoNLL03	CoNLL04		CASIE		16-res
F1	Ent	Rel-S		Evt-Tri	Evt-Arg	Rel-S
W/O SAM	28.47	0		4.03	0	0
W/O TriA	58.58	49.40		6.97	1.51	29.77
W/O Label	92.59	70.94		71.18	62.29	74.64
UniEX	92.65	73.40		71.46	62.91	76.02

Table 5: Experiment results of UniEX with different ablation strategies on the test set of four downstream datasets: CoNLL03 (entity), CoNLL04 (relation), CASIE (event) and 16-res (sentiment).

because the directional matching in the triaffine transformation will tend to guide the model to predict more credible targets.

4.4 Ablation Study

In this section, we intend to verify the necessity of key components of the UniEX, including the flow controlling and triaffine transformation. Table 5 shows ablation experiment results of UniEX on four downstream tasks.

W/O SAM: removing the schema-based attention mask matrix that controls the flowing of labels. We find that model performance is almost zero on many tasks, which demonstrates the importance of eliminating intra-information of labels. AMM makes the labels unreachable to each other, effectively avoiding the mutual interference of label semantics.

W/O TriA: replacing the triaffine transformation with the multi-head selection network, which multiplies the schema and the head-to-tail span of the text respectively, and then replicates and adds them to get the scoring matrix. The significant performance decline demonstrates the important role of triaffine attention mechanism in establishing dense correspondence between schemas and text spans.

W/O Label: replacing the label names with the special token `[unused n]`, which eliminates label semantics while allowing the model to still distinguish between different labels. We find a slight degradation of model performance in small datasets CoNLL03 and 16-res, indicating that the prior knowledge provided by label names can effectively compensate for the deficiency of training data. As the correspondence between schema and extraction targets is not affected, model performance in large datasets tends to stabilize.

Model	CoNLL03 (sent/s)	CoNLL04 (sent/s)	CASIE (sent/s)	16-res (sent/s)
UIE	2.1(×1.0)	1.0(×1.0)	1.1(×1.0)	1.4(×1.0)
UniEX	16.5(×7.9)	16.6(×16.6)	14.9(×13.5)	19.7(×14.1)

Table 6: The efficiency comparison of UIE and UniEX with batch_size=1. $(\times k)$ is the relative inference-speed.

4.5 Efficiency Analysis

To verify the computation efficiency of our approach on universal IE, we compare inference-speed with UIE (Lu et al., 2022) on the four standard datasets mentioned in section 4.4. As shown in Table 6, we can find that since generating the target structure is a token-wise process, the inference-speed of UIE is slow and limited by the length of the target structure. On the contrary, UniEX can

decode all the target structures at once from the scoring matrices obtained by triaffine transformation, with an average speedup ratio of 13.3 to UIE.

5 Conclusion

In this paper, we introduce a new paradigm for universal IE by converting all IE tasks into joint span detection, classification and association problems with a unified extractive framework. UniEX collaboratively learns the generalized knowledge from schema-based prompts and controls the correspondence between schema and extraction targets via the triaffine attention mechanism. Experiments on both supervised setting and low-resource scenarios verify the transferability and effectiveness of our approaches.

Limitations

In this paper, our main contribution is an effective and efficient framework for universal IE. We aim to introduce a new unified IE paradigm with extractive structures and triaffine attention mechanism, which can achieve better performance in a variety of tasks and scenarios with more efficient inference-speed. However, it is non-trivial to decide whether a sophisticated and artificial prompt is required for complex datasets and large label sets. In addition, we only compare with limited baselines with specific datasets configurations when analyzing the performance of the UniEX in supervised, few-shot and zero-shot settings. In experiments, we implement only a few comparative experiments between BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019b) due to the limit of computational resources.

Ethical Considerations

As an important domain of natural language processing, information extraction is a common technology in our society. It is necessary to discuss the ethical influence when using the extraction models (Leidner and Plachouras, 2017). In this work, We develop a new universal IE framework, which enhances the generalization ability in various scenarios. As discussed (Schramowski et al., 2019, 2022; Blodgett et al., 2020), pre-trained LMs might contain human-made biases, which might be embedded in both the parameters and outputs of the open-source models. In addition, we note the potential abuse of universal IE models, as these models achieve excellent performance in various domains and settings after adapting to pre-training on large-scale IE datasets, which allows the models to be integrated into applications often without justification. We encourage open debating on its utilization, such as the task selection and the deployment, hoping to reduce the chance of any misconduct.

References

Su Lin Blodgett, Solon Barocas, Hal Daumé III, and Hanna Wallach. 2020. Language (technology) is power: A critical survey of "bias" in nlp. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5454–5476.

Chih-Yao Chen and Cheng-Te Li. 2021. Zs-bert: Towards zero-shot relation extraction with attribute representation learning. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3470–3479.

Yew Ken Chia, Lidong Bing, Soujanya Poria, and Luo Si. 2022. Relationprompt: Leveraging prompts to generate synthetic data for zero-shot relation triplet extraction. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 45–57.

Leon Derczynski, Eric Nichols, Marieke van Erp, and Nut Limsopatham. 2017. Results of the wnut2017 shared task on novel and emerging entity recognition. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 140–147.

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

Ning Ding, Guangwei Xu, Yulin Chen, Xiaobin Wang, Xu Han, Pengjun Xie, Haitao Zheng, and Zhiyuan Liu. 2021. Few-nerd: A few-shot named entity recognition dataset. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3198–3213.

George R 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)*.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified language model pre-training for natural language understanding and generation. *Advances in Neural Information Processing Systems*, 32.

Jenny Rose Finkel and Christopher D Manning. 2009. Nested named entity recognition. In *Proceedings of the 2009 conference on empirical methods in natural language processing*, pages 141–150.

Ralph Grishman. 2019. Twenty-five years of information extraction. *Natural Language Engineering*, 25(6):677–692.

Pankaj Gupta, Hinrich Schütze, and Bernt Andrassy. 2016. Table filling multi-task recurrent neural network for joint entity and relation extraction. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2537–2547.

Harsha Gurulingappa, Abdul Mateen Rajput, Angus Roberts, Juliane Fluck, Martin Hofmann-Apitius, and Luca Toldo. 2012. Development of a benchmark corpus to support the automatic extraction of drug-related adverse effects from medical case reports. *Journal of Biomedical Informatics*, 45(5):885–892.

Xu Han, Hao Zhu, Pengfei Yu, Ziyun Wang, Yuan Yao, Zhiyuan Liu, and Maosong Sun. 2018. Fewrel: A large-scale supervised few-shot relation classification dataset with state-of-the-art evaluation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4803–4809.

Yutai Hou, Wanxiang Che, Yongkui Lai, Zhihan Zhou, Yijia Liu, Han Liu, and Ting Liu. 2020. Few-shot slot tagging with collapsed dependency transfer and label-enhanced task-adaptive projection network. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1381–1393.

Nitish Shirish Keskar, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Unifying question answering, text classification, and regression via span extraction. *arXiv preprint arXiv:1904.09286*.

Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *ICLR (Poster)*.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. Albert: A lite bert for self-supervised learning of language representations. In *International Conference on Learning Representations*.

Jochen L Leidner and Vassilis Plachouras. 2017. Ethical by design: Ethics best practices for natural language processing. In *Proceedings of the First ACL Workshop on Ethics in Natural Language Processing*, pages 30–40.

Xiaoya Li, Jingrong Feng, Yuxian Meng, Qinghong Han, Fei Wu, and Jiwei Li. 2020. A unified mrc framework for named entity recognition. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5849–5859.

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*, pages 7999–8009.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019a. Multi-task deep neural networks for natural language understanding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4487–4496.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019b. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

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)*, pages 5755–5772.

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*, pages 3219–3232.

Yi Luan, Dave Wadden, Luheng He, Amy Shah, Mari Ostendorf, and Hannaneh Hajishirzi. 2019. A general framework for information extraction using dynamic span graphs. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3036–3046.

Ying Luo, Fengshun Xiao, and Hai Zhao. 2020. Hierarchical contextualized representation for named entity recognition. In *AAAI*, pages 8441–8448. AAAI Press.

Jie Ma, Miguel Ballesteros, Srikanth Doss, Rishita Anubhai, Sunil Mallya, Yaser Al-Onaizan, and Dan Roth. 2022a. Label semantics for few shot named entity recognition. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1956–1971.

Tingting Ma, Huiqiang Jiang, Qianhui Wu, Tiejun Zhao, and Chin-Yew Lin. 2022b. Decomposed meta-learning for few-shot named entity recognition. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1584–1596.

Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. 2018. The natural language decathlon: Multitask learning as question answering. *arXiv preprint arXiv:1806.08730*.

Aaron Mueller, Jason Krone, Salvatore Romeo, Saab Mansour, Elman Mansimov, Yi Zhang, and Dan Roth. 2022. Label semantic aware pre-training for few-shot text classification. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8318–8334.

Ion Muslea et al. 1999. Extraction patterns for information extraction tasks: A survey. In *The AAAI-99 workshop on machine learning for information extraction*, volume 2. Orlando Florida.

Tomoko Ohta, Yuka Tateisi, and Jin-Dong Kim. 2002. The genia corpus: an annotated research abstract corpus in molecular biology domain. In *Proceedings of the second international conference on Human Language Technology Research*, pages 82–86.

Giovanni Paolini, Ben Athiwaratkun, Jason Krone, Jie Ma, Alessandro Achille, RISHITA ANUBHAI, Cicero Nogueira dos Santos, Bing Xiang, and Stefano Soatto. 2020. Structured prediction as translation between augmented natural languages. In *International Conference on Learning Representations*.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Hwee Tou Ng, Anders Björkelund, Olga Uryupina, Yuchen Zhang, and Zhi Zhong. 2013. Towards robust linguistic analysis using ontonotes. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 143–152.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020a. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020b. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21:1–67.

Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling relations and their mentions without labeled text. In *Proceedings of the 2010 European conference on Machine learning and knowledge discovery in databases: Part III*, pages 148–163.

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) at HLT-NAACL 2004*, pages 1–8.

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

Victor Sanh, Albert Webson, Colin Raffel, Stephen Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, et al. 2022. Multitask prompted training enables zero-shot task generalization. In *The Tenth International Conference on Learning Representations*.

Taneeya Satyapanich, Francis Ferraro, and Tim Finin. 2020. CASIE: extracting cybersecurity event information from text. In *AAAI*, pages 8749–8757. AAAI Press.

Patrick Schramowski, Cigdem Turan, Nico Andersen, Constantin A Rothkopf, and Kristian Kersting. 2022. Large pre-trained language models contain human-like biases of what is right and wrong to do. *Nature Machine Intelligence*, 4(3):258–268.

Patrick Schramowski, Cigdem Turan, Sophie Jentzsch, Constantin Rothkopf, and Kristian Kersting. 2019. Bert has a moral compass: Improvements of ethical and moral values of machines. *arXiv preprint arXiv:1912.05238*.

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

Jue Wang and Wei Lu. 2020. Two are better than one: Joint entity and relation extraction with table-sequence encoders. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1706–1721.

Peiyi Wang, Runxin Xu, Tianyu Liu, Qingyu Zhou, Yunbo Cao, Baobao Chang, and Zhifang Sui. 2022. An enhanced span-based decomposition method for few-shot sequence labeling. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5012–5024.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

Zhepei Wei, Jianlin Su, Yue Wang, Yuan Tian, and Yi Chang. 2020. A novel cascade binary tagging framework for relational triple extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1476–1488.

Lu Xu, Hao Li, Wei Lu, and Lidong Bing. 2020. Position-aware tagging for aspect sentiment triplet extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2339–2349.

Hang Yan, Junqi Dai, Tuo Ji, Xipeng Qiu, and Zheng Zhang. 2021. A unified generative framework for aspect-based sentiment analysis. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2416–2429.

Ping Yang, Junjie Wang, Ruyi Gan, Xinyu Zhu, Lin Zhang, Ziwei Wu, Xinyu Gao, Jiaxing Zhang, and Tetsuya Sakai. 2022. Zero-shot learners for natural language understanding via a unified multiple choice perspective. *arXiv preprint arXiv:2210.08590*.

Deming Ye, Yankai Lin, Peng Li, and Maosong Sun. 2022. Packed levitated marker for entity and relation extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4904–4917.

Bowen Yu, Zhenyu Zhang, Xiaobo Shu, Tingwen Liu, Yubin Wang, Bin Wang, and Sujian Li. 2020a. Joint extraction of entities and relations based on a novel decomposition strategy. In *ECAI*.

Juntao Yu, Bernd Bohnet, and Massimo Poesio. 2020b. Named entity recognition as dependency parsing. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6470–6476.

Zheng Yuan, Chuanqi Tan, Songfang Huang, and Fei Huang. 2022. Fusing heterogeneous factors with triaffine mechanism for nested named entity recognition. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 3174–3186.

Amir Zeldes. 2017. The gum corpus: creating multilayer resources in the classroom. *Language Resources and Evaluation*, 51(3):581–612.

A Experiment Details

This section describes the details of experiments, including the dataset descriptions, unified EX input formats, metrics and training implementation.

A.1 Details of Downstream Tasks

A.1.1 Supervised Setting

For the supervised setting, We conduct downstream tasks on 4 IE tasks, 14 datasets, and the detailed statistic of each dataset is shown in Table 7.

Entity We conduct entity extraction experiments on four datasets, including the flat entity dataset extraction dataset CONLL03 (Sang and De Meulder, 2003), and nested entity extractions datasets ACE04 (Doddington et al., 2004), ACE05-Ent (Walker et al., 2006) and GENIA (Ohta et al., 2002). For the CONLL03, ACE04 and ACE05-Ent, We use the same processing and splits as (Li et al., 2020). For the GENIA, we follow the pre-processing steps and data split as (Finkel and Manning, 2009).

Relation We conduct relation extraction experiments on five joint entity-relation extraction datasets across several languages and domains, including CONLL04 (Roth and Yih, 2004), ACE05-Rel (Walker et al., 2006), NYT (Riedel et al., 2010), SciERC (Luan et al., 2018) and ADE (Gurulingappa et al., 2012). We follow the pre-processing versions and data split of previous works (Gupta et al., 2016; Yu et al., 2020a; Luan et al., 2019).

Event For ACE05-Evt, we follow the same types, data splits, and pre-processing steps as (Lin et al., 2020). For CASIE (Satyapanich et al., 2020), we remove three incomplete annotated documents, then split the remaining documents into three sets as (Lu et al., 2022).

Sentiment We conduct sentiment extraction experiments on the sentiment triplet extraction (Xu et al., 2020) of SemEval 14/15/16 aspect sentiment analysis datasets. We employ the pre-processing datasets of the previous work (Yan et al., 2021).

A.1.2 Few-shot Setting

For the few-shot setting, we conduct downstream tasks on 2 few-shot named entity recognition datasets:

Few-NERD (Ding et al., 2021). It is annotated with a hierarchy of 8 coarse-grained and 66 finegrained entity types. Two tasks are considered on this dataset: i) Intra, where all entities in train/dev/test splits belong to different coarsegrained types. ii) Inter, where train/dev/test splits may share coarse-grained types while keeping the fine-grained entity types mutually disjoint.

	|Ent|	|Rel|	|Evt|	#Train	#Val	#Test
ACE04	7	-	-	6,202	745	812
ACE05-Ent	7	-	-	7,299	971	1,060
CoNLL03	4	-	-	14,041	3,250	3,453
GENIA	5	-	-	14,824	1,855	1,854
ACE05-Rel	7	6	-	10,051	2,420	2,050
CoNLL04	4	5	-	922	231	288
NYT	3	24	-	56,196	5,000	5,000
SciERC	6	7	-	1,861	275	551
ADE	2	1	-	3,417	427	428
ACE05-Evt	-	-	33	19,216	901	676
CASIE	21	-	5	11,189	1,778	3,208
14res	2	3	-	1,266	310	492
14lap	2	3	-	906	219	328
15res	2	3	-	605	148	322
16res	2	3	-	857	210	326

Table 7: Detailed datasets statistics. |*| indicates the number of categories, and # is the number of sentences in the specific subset. We take sentiment types as special relation type: positive, negative, and neutral; and each sentiment triplet holds a aspect and a opinion.

Cross-Dataset (Hou et al., 2020). Four datasets focusing on four domains are used here: CoNLL2003 (Sang and De Meulder, 2003) (news), GUM (Zeldes, 2017) (Wiki) , WNUT-2017 (Derczynski et al., 2017) (social), and Ontonotes (Pradhan et al., 2013) (mixed). Among them, we take two domains for training, one for validation, and the remaining for test.

A.1.3 Zero-shot Setting

For the zero-shot setting, we conduct downstream tasks on 2 zero-shot named entity recognition datasets:

FewRel (Han et al., 2018) is hand-annotated for few-shot relation extraction, we further made it suitable for the zero-shot setting after data splitting into disjoint relation label sets for training, validation and testing as (Chia et al., 2022).

Wiki-ZSL (Chen and Li, 2021) is constructed through distant supervision over Wikipedia articles and the Wikidata knowledge base.

To partition the data into seen and unseen label sets, we follow the same process as (Chia et al., 2022) to be consistent. For each dataset, a fixed

Backbone	RoBERTa-large/RoBERTa-base				BERT-base			
Task	Entity	Relation	Event	Sentiment	Cross Dataset		Wiki-ZSL	FewRel
Phase	finetuning	finetuning	finetuning	finetuning	pretraining	finetuning	pretraining	pretraining
Learning Rate	2E-5	2E-5	2E-5	2E-5	2E-5	2E-5	2E-05	2E-5
Batch Size	32	32	32	32	32	2	32	32
Schedule	linear	linear	linear	linear	linear	linear	linear	linear
Warmup Rate	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06
Epoch	200	400	200	200	100	100	4	4

Table 8: Hyper-parameters for UniEX-base and UniEX-large on different tasks and datasets.

Hyper-parameter	UniEX-base	UniEX-large
Backbone	Roberta-large	Roberta-base
Layers of Encoder	12	24
Hidden Dimension	768	1,024
FF hidden size	3072	4096
Layer Normalize ϵ	1e-5	1e-5
Attention head	12	16

Table 9: Model architectures.

number of labels are randomly selected as unseen labels while the remaining labels are treated as seen labels during training. The unseen label set size is set to m=5 in our experiments. In order to reduce the effect of experimental noise, the label selection process is repeated for five different random seeds to produce different data folds. For each data fold, the test set consists of the sentences containing unseen labels. Five validation labels from the seen labels are used to select sentences for early stopping and hyperparameter tuning. The remaining sentences are treated as the train set. Hence, the zero-shot setting ensures that train, validation and test sentences belong to disjoint label sets.

A.2 Evaluation Metric

We use span-based offset Micro-F1 as the primary metric to evaluate the model as (Lu et al., 2022)

- **Entity**: an entity mention is correct if its offsets and type match a reference entity.
- **Relation Strict**: relation with strict match, a relation is correct if its relation type is correct and the offsets and entity types of the related entity mentions are correct.
- **Relation Triplet**: relation with boundary match, a relation is correct if its relation type is correct and the string of the subject/object are correct.
- **Event Trigger**: an event trigger is correct if its offsets and event type matches a reference trigger.
- **Event Argument**: an event argument is correct if its offsets, role type, and event type match a reference argument mention.

Task	Dataset	UIE 770M	UniEX 372M
Entity Extraction	ACE04	1.23	18.29
	ACE05-Ent	1.62	18.16
	CoNLL03	2.06	16.45
Relation Extraction	ACE05-Rel	1.64	18.69
	CoNLL04	1.00	16.60
	SciERC	1.02	17.09
Event Extraction	ACE05-Evt	1.55	12.93
	CASIE	1.55	12.93
Sentiment Extraction	14-res	1.45	18.60
	14-lap	1.49	19.78
	15-res	1.41	18.37
	16-res	1.38	19.71

Table 10: The average number of sentences generated per second by UIE and UniEX in the decoding phase.

- **Sentiment Triplet**: a correct triplet requires the offsets boundary of the target, the offsets boundary of the opinion span, and the target sentiment polarity to be all correct at the same time.

A.3 Training Implementation

To make a fair comparison, we first initialize UniEX-base and UniEX-large with RoBERTa-base and RoBERTa-large checkpoints (Liu et al., 2019b) for the supervised setting, and use the BERT-base checkpoint (Devlin et al., 2019) as the backbone for the few-shot and zero-shot settings. The model architectures are shown in Table 9. We employ Adam optimizer (Kingma and Ba, 2015) as the optimizer with 1e-8 weight decay. Table 8 shows the detailed hyper-parameters for downstream tasks. We truncate the concatenated overall length of schema-based prompt s and raw text x to 512 during training.

A.4 Unified Input

Inspired by template examples in UIE (Lu et al., 2022), we design a simple rule to transform the

original text to a unified EX format. In addition, we present four examples for different tasks:

An example of CONLL03 (Entity Extraction):
Raw text: {x: "Arafat goes to Nablus ahead of cabinet meeting .", *entity type*: [Location, Organization, Person, Miscellaneous], *extraction target*: [(Arafat, 1, 1, Person), (Nablus, 4, 4, Location)]}
Transformed Input: `[CLS]` Entity Extraction `[R-LEP]`^{1} Location `[R-LEP]`^{2} Organization `[R-LEP]`^{3} Person `[R-LEP]`^{4} Miscellaneous `[SEP]` Arafat goes to Nablus ahead of cabinet meeting . `[SEP]`

An example of CONLL04 (Relation Extraction):
Raw text: {x: "In 1752 , flagmaker Betsy Ross was born in Philadelphia .", *entity-relation type*: [(Organization, organization based in, Location), (Location, location in, Location), (Person, live in, Location), (Person, work for, Organization), (Person, kill, Person)], *extraction target*: [(Betsy Ross, 5, 6, Person), (Philadelphia, 10, 10, Location), (Betsy Ross, live in, Philadelphia)]}
Transformed Input: `[CLS]` Relation Extraction `[R-LEP]`^{1} Location `[R-LEP]`^{2} Organization `[R-LEP]`^{3} Person `[R-LEP]`^{4} Miscellaneous `[R-LEP]`^{5} work for `[R-LEP]`^{6} organization based in `[R-LEP]`^{7} location in `[R-LEP]`^{8} live in `[R-LEP]`^{9} kill `[SEP]` In 1752 , flagmaker Betsy Ross was born in Philadelphia . `[SEP]`

An example of ACE05-Evt (Event Extraction):
Raw text: {x: "Sergeant Chuck Hagel was seriously wounded twice in Vietnam .", *event-trigger-argument type*: [(Born, Trigger, Person, Place), (Injure, Trigger, Victim, Agent, Place, Instrument), (Convict, Trigger, Defendant, Adjudicator, Place), ...], *extraction target*: [(Chuck Hagel, 2, 3, Victim), (wounded, 6, 6, Trigger), (Vietnam, 9, 9, Place), (Injure, wounded, Chuck Hagel, Vietnam)]}
Transformed Input: `[CLS]` Event Extraction `[R-LEP]`^{1} Trigger `[R-LEP]`^{2} Person `[R-LEP]`^{3} Place ... `[R-LEP]`^{i} Born `[R-LEP]`$^{i+1}$ Injure ... `[R-LEP]`^{n} Trigger-Argument `[SEP]` Sergeant Chuck Hagel was seriously wounded twice in Vietnam . `[SEP]`

An example of 16-res (Sentiment Extraction):
Raw text: {x: "I had the duck breast special on my last visit and it was incredible .", *entity-relation-entity type*: [(Aspect, Positive, Opinion), (Aspect, Negative, Opinion), (Aspect, Neutral, Opinion)], *extraction target*: [(duck breast special, 4, 6, Aspect), (incredible, 14, 14, Opinion), (Positive, duck breast special, incredible)]}
Transformed Input: `[CLS]` Sentiment Extraction `[R-LEP]`^{1} Aspect `[R-LEP]`^{2} Opinion `[R-LEP]`^{3} Positive `[R-LEP]`^{4} Negative `[R-LEP]`^{5} Neutral `[SEP]` I had the duck breast special on my last visit and it was incredible . `[SEP]`

A.5 Unified Decoding

As shown in Figure 5, in order to depict the training and inference processes in more detail, we show the structural tables and spotting designators of the examples in figure 4 in the entity/relation/event extraction tasks.

A.6 Decoding Efficiency

As shown in Figure 10, to explicitly compare the structural decoding efficiency of different universal IE models, we illustrate the average number of sentences generated per second by UIE and UniEX during the decoding phase.

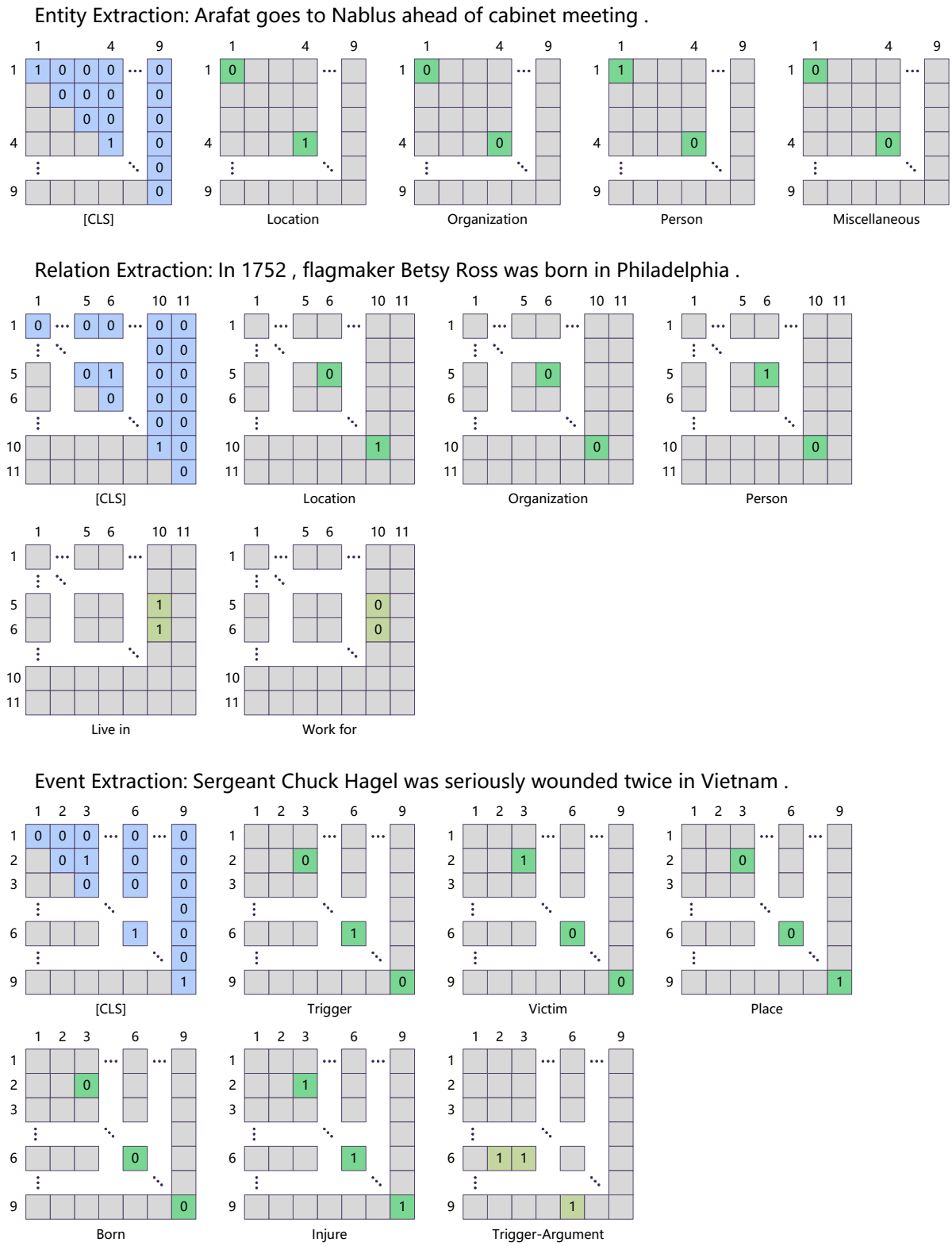

Figure 5: The decoding process of the UniEX. Each schema corresponds to a structural table, and each rectangle in the structural table represents an internal span, the gray spans are the invalid spans that and do not participate in model training. Other spans are spotting designators, among them, water-blue spans for span detection, viridis spans for span classification and atrovirens spans for span association.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
"Limitations" section

☑ A2. Did you discuss any potential risks of your work?
"Ethical Considerations" section

☑ A3. Do the abstract and introduction summarize the paper's main claims?
"Abstract" and "Introduction" sections

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

Sections 4.1, 4.2, 4.3, 4.4 and 4.5 Appendix A.1, A.2, A.3

☑ B1. Did you cite the creators of artifacts you used?
Sections 4.1, 4.2, 4.3 Appendix A.1, A.2, A.3

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Sections 4.1, 4.2, 4.3 Appendix A.1, A.2, A.3

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Sections 4.1, 4.2, 4.3 Appendix A.1, A.2, A.3

☑ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Sections 4.1, 4.2, 4.3 Appendix A.1, A.2, A.3

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Sections 4.1, 4.2, 4.3 Appendix A.1, A.2, A.3

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Table 7, 8, 9

C ☑ Did you run computational experiments?

Sections 4.2, 4.3, 4.4 and 4.5

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Appendix A.1, A.2, A.3

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Appendix A.1, A.2, A.3

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Sections 4.1, 4.2, 4.3, 4.4 and 4.5

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Sections 4.1, 4.2, 4.3

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
Not applicable. Left blank.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
Not applicable. Left blank.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
Not applicable. Left blank.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Not applicable. Left blank.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Not applicable. Left blank.