# Incorporating Scenario Knowledge into A Unified Fine-tuning Architecture for Event Representation

Jianming Zheng, Fei Cai*, Honghui Chen
Science and Technology on Information Systems Engineering Laboratory, National University of Defense Technology
Changsha, China
{zhengjianming12,caifei,chenhonghui}@nudt.edu.cn

## ABSTRACT

Given an occurred event, human can easily predict the next event or reason the preceding event, yet which is difficult for machine to perform such event reasoning. Event representation bridges the connection and targets to model the process of event reasoning as a machine-readable format, which then can support a wide range of applications in information retrieval, e.g., question answering and information extraction. Existing work mainly resorts to a joint training to integrate all levels of training loss in event chains by a simple loss summation, which is easily trapped into a local optimum. In addition, the scenario knowledge in event chains is not well investigated for event representation. In this paper, we propose a unified fine-tuning architecture, incorporated with scenario knowledge for event representation, i.e., *UniFA-S*, which mainly consists of a unified fine-tuning architecture (*UniFA*) and a scenario-level variational auto-encoder (*S-VAE*). In detail, *UniFA* employs a multi-step fine-tuning to integrate all levels of training and *S-VAE* applies a stochastic variable to implicitly represent the scenario-level knowledge. We evaluate our proposal from two aspects, i.e., the representation and inference abilities. For the representation ability, our ensemble model *UniFA-S* can beat state-of-the-art baselines for two similarity tasks. For the inference ability, *UniFA-S* can outperform the best baseline, achieving 4.1%-8.2% improvements in terms of accuracy for various inference tasks.

## CCS CONCEPTS

• **Information systems** → *Content analysis and feature selection.*

## KEYWORDS

event representation; pre-training; fine-tuning; scenario knowledge

**ACM Reference Format:**
Jianming Zheng, Fei Cai, Honghui Chen. 2020. Incorporating Scenario Knowledge into A Unified Fine-tuning Architecture for Event Representation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '20), July 25–30, 2020, Virtual Event, China.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3397271.3401173

*Corresponding author.


*SIGIR '20, July 25–30, 2020, Virtual Event, China*

ACM ISBN 978-1-4503-8016-4/20/07…$15.00
https://doi.org/10.1145/3397271.3401173

## 1 INTRODUCTION

Given a snap observation of an event, people can easily predict and reason unobserved causes according to the observed event, i.e., what might have happened just before, what might occur afterwards, and how this pair of events are chained through causes and effects [30]. For instance, considering the fact that "A man broke the record", human can intuitively infer that "he got awarded" is more likely to happen afterwards in contrast with "he got arrested". However, representing such knowledge (so-called *scripts* in [29]) as a machine-readable format is challenging in the field of artificial intelligence, which can support a series of following applications, e.g., question answering [18], discourse understanding [12] and information extraction [20], etc.

Following the line of work for event representation [2, 3], our approach aims at learning the distributed representations for narrative event chains generated from text, by which the downstream tasks, e.g., event similarity [37], transitive sentence similarity [37] and script event prediction [10], can be fulfilled accordingly. Figure 1 presents an example of a narrative event chain with three levels, i.e., the intra-event level, the inter-event level and the scenario level. The intra-event level is represented in the purple box, including the intra-event elements; the inter-event level denoted in the green box concentrates on an event pair and their relation; the scenario level considers a whole event chain, which is indicated in the red box in Fig. 1. Existing works on event representation can be mainly categorized into three folds, i.e., intra-event-based [10, 37], inter-event-based [19, 21, 36] and external-knowledge-based approaches [6, 16]. These approaches either focus on the multiplicative interactions among intra-event elements or explore the complex event relationships or seek for external commonsense knowledge to enrich additional context for event representation. However, such existing models typically adopt either a single level loss (e.g., intra-event loss [37] and inter-event loss [10, 21, 37]) or a joint-training loss [6, 16] in the training phase. The simple additive loss could be easily trapped into a local optimum although the joint-training loss makes sense and is capable of capturing different level of losses.

In addition, previous work has not well investigated the scenario-level knowledge, i.e., so-called *event context*, for event presentation. We argue that the development of event chain is guided by the scenario of event chain. Under different scenarios of event chains, the same start event could be plotted to different directions. For example, considering two scenarios: (i) *robbery* and (ii) *first aid*, given the start event "Jim broke the car window," there are two

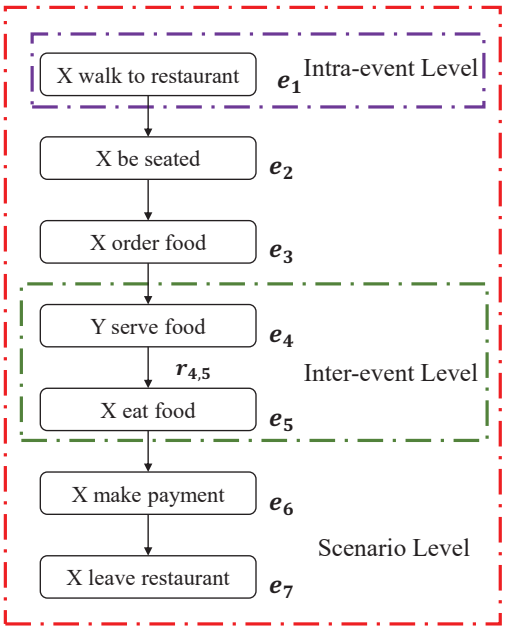



**Figure 1: An example of a narrative event chain.**

polar opposite subsequent event for these two scenarios, e.g., "Jim stole a bag" for *robbery* and "Jim rescued the comatose driver" for *first aid*. Although Weber et al. [37] first introduced the *scenario* conception and claimed that tensor-based models can capture the scenario-level semantics, the semantics they obtained was only for individual event yet with few concern for the whole event chain.

In this paper, we attempt to provide a solution for event representation by proposing a unified fine-tuning architecture incorporated with scenario knowledge for event representation, i.e., *UniFA-S*, which mainly consists of a unified fine-tuning architecture (*UniFA*) and a scenario-level variational auto-encoder (*S-VAE*). On the overall architecture side, our inspiration draws from the successful applications of inductive transfer learning in natural language processing [5, 11, 15, 26, 38]. Accordingly, we propose a **U**nified **F**ine-tuning **A**rchitecture (*UniFA*), which is a scalable and stackable framework consisting of four key components, i.e., bert-base-uncased, raw-text, intra-event and inter-event components, connected by a multi-step fine-tuning method. With the merits of cascaded training, the multi-step fine-tuning can not only contribute to minimize the loss, but can avoid being trapped in the local optimum. On the event-scenario side, we propose a **S**cenario-level **V**ariational **A**uto **E**ncoder (*S-VAE*) to implicitly represent the scenario-level knowledge. In particular, *S-VAE* regards the event chain as a dialog generation process, and further introduces a stochastic latent variable to guide the event representation.

To the best of our knowledge, we are the first to propose a novel fine-tuning framework to integrate each level of training and employ the variational auto encoder to implicitly represent the scenario-level knowledge for event representation. We evaluate our proposal from two aspects, i.e., the representation ability and the inference ability. The experimental results show that our ensemble model *UniFA-S* outperforms state-of-the-art baselines from the perspective of both representation and inference abilities. In particular, we find that the multi-step fine-tuning is the most effective fine-tuning method and modeling scenario knowledge has an outstanding ability in utilizing the context especially for dealing with the inference tasks.

In summary, our main contributions in this paper are in three folds.

(1) We propose a unified fine-tuning architecture based approach *UniFA-S* for event representation that integrates all levels of training by a multi-step fine-tuning method.
(2) We design a scenario-level variation auto encoder to implicitly represent the scenario-level knowledge that pilots the event generation in event chains.
(3) We evaluate the performance of our proposal from the perspective of representation and inference abilities. We find *UniFA-S* can outperform start-of-the art baselines on eight datasets for four main tasks.

## 2 RELATED WORK

We first introduce the history and modeling methods on narrative event chain in Section 2.1; then we describe the existing pre-training and fine-tuning methods in Section 2.2.

### 2.1 Narrative event chain

*Event chain* models human expectations about the relevant causal relationships among events. An event chain can be used to infer how events will unfold in a given scenario [29]. Restricted to the manual acquisition, early work on event chain shows a slow progression until narrative event chains introduced by [2]. Chambers and Jurafsky [2] assumed that although a narrative script had several participants, there was a central actor (i.e., protagonist) who characterized a narrative chain. In this assumption, probabilistic co-occurrence-based models combined with dependency parser can realize the automatic extraction of narrative event chains from raw text. They also casted narrative events as triplet of the from <*predicate, dependency_type*>, where the *predicate* was a verb lemma and the *dependency_type* denoted a grammatical dependency relation between the *predicate* and the protagonist, e.g., 'subj', 'obj' or 'iobj'. Besides, Pichotta and Mooney [27] explored a richer representation over multi-argument event format.

From modeling perspectives, existing event representation works can be classified into three main types, i.e., intra-event-based, inter-event-based and external-knowledge-based embeddings. Firstly, intra-event-based embeddings mainly concentrate on the multiplicative interaction among intra-event elements. Granroth-Wilding and Clark [10] simply concatenated on predicate and argument embeddings and fed them into a neural network to get the event representation. While Weber et al. [37] used the tensor-network-based model to capture more subtle semantic interactions. Secondly, inter-event-based embeddings mainly research on the complex and diverse relations between events. Wang et al. [36] utilized the LSTM hidden states to integrate the chain order information into event model. Li et al. [19] extended the narrative event chains into the narrative event evolutionary graph to model the dense connections among events. While Lee and Goldwasser [17] broadened the single relation (time-order relation) into the diverse ones based on the discourse relations from PDTB [28]. Besides, Lv et al. [21] exploited the

event segment relations instead of the event-pair relation. Thirdly, by jointly training the event representation model and external knowledge, external-knowledge-based embeddings intendedert to mine the potential connections between narrative event chains and external knowledge. Ding et al. [6] introduced ATOMIC [30] to obtain the sentiment and intent information of event.

Clearly, different modeling perspectives can capture different characteristics of narrative event chains. However, there is no unified framework to integrate characteristics of each level during training, which is consider in our work via *UniFA*. Additionally, the aforementioned approaches neglect the acquisition and representation of scenario-level knowledge that plays a vital role in directing the development of narrative event chains. Our proposed *S-VAE* can effectively solve this problem by introducing a latent variable.

## 2.2 Pretraining and fine-tuning methods

The pre-training models have been shown effective in the natural language processing tasks, e.g., question answering [23], textual entailment [26], semantic role labeling [5] sentimental analysis [4, 39], etc. These pre-training models can be mainly classified into two groups, i.e., feature-based models and fine-tuning models. The feature-based models generate the pre-trained embeddings from other tasks, where the output can be regarded as the additional features for the downstream tasks. Word2vec [24] and GloVe [25], two classical word embeddings models, focus on transforming words into the distributed representations and capturing the syntactics as well as the semantics by pre-training the neural language models on a large text corpora. In addition, McCann et al. [23] concentrated on the machine translation task to get the contextualize word vectors (CoVe). While Peters et al. [26] developed the sequence-level model, i.e., ELMo, to capture the complex word features across different linguistic contexts and then use ELMo to generate the context-sensitive word embeddings. Different from the feature-based strategy, the fine-tuning models first produce the contextual word representations which have been pre-trained from unlabeled text and then fine-tune for a supervised downstream task. Dai and Le [4] trained a sequence auto-encoder model on unlabeled text as an initialization of the other supervised network. Devlin et al. [5] primarily trained de-noise auto encoder on language modeling and next sentence predication tasks to get the pre-trained layers.

However, existing models still employ a general fine-tuning method (e.g., Bert [5]) and have not formed a task-specific fine-tuning method. Whereas our proposal, i.e., multi-step fine-tuning method, is specially designed for the representation of event in narrative event chains.

# 3 APPROACH

As shown in Fig. 2, our ensemble model *UniFA-S* consists of two main parts, i.e., *UniFA* and *S-VAE*. *UniFA* concentrates on how to integrate all levels of training with the raw-text sequence as input, where we can get the hidden representations of intra-event elements, event representation and relation representation. After that, such hidden representations are fine-tuned by *S-VAE* that applies a stochastic variable to pilot the event generation. We detail these two parts in Section 3.1 and Section 3.2, respectively.

## 3.1 Unified fine-tuning architecture

In this section, we design a unified fine-tuning architecture (*UniFA*) that combines the advantages of intra-event and inter-event trainings. As shown in the left part of Fig. 2, *UniFA* consists of four main components, i.e., bert-base-uncased component, raw-text component, intra-event component and inter-event component, that are connected by multi-step fine-tuning. We directly borrow the merits from the literature [5] for the bert-base-uncased component, thus we focus on the following three components.

*3.1.1* ***Raw-text component.*** In raw-text corpus selection, the bert-base-uncased model adopts the BooksCorpus (800M words) [40] and English Wikipedia (2500M words) as the pre-trained corpora [5]. We extract the narrative event chains from the New York Times portion of the Gigaword corpus [9]. As employing the bert-base-uncased model for event representation directly will suffer from the loss brought by the inconsistent between the source and target domains, we employ a fine-tuning process to minimize the loss brought by the difference of raw-text corporas.

We first employ a masked language model [5] to fine-tune the pre-trained bert-base-uncased model. We randomly mask some words in a text sequence with $[mask]$ tokens. Then, the loss of raw-text component $loss^{1^{st}}$ can be represented as

$$loss^{1^{st}} = -\sum_{\widetilde{w}\in\{[mask]\}} p(\widetilde{w}|bert)\log p(\widetilde{w}|bert) + \lambda L(\theta_1), \quad (1)$$

where $\{[mask]\}$ is a set of masked words, the probability $p(\widetilde{w}|bert)$ is computed using a softmax classifier on the bert-base-uncased model, $\lambda$ is a pre-defined weighted parameter, and $L(\theta_1)$ means $L2$ regularization on parameters $\theta_1$ in the raw-text component.

*3.1.2* ***Intra-event component.*** Given a chain of narrative events, i.e., $\{e_1, e_2, \cdots, e_n\}$, each event $e$ consists of three intra-event elements, i.e., predicate ($pred(e)$), subject ($subj(e)$) and object ($obj(e)$), which are extracted from the raw text. For example, given an example "Kevin is eating his favorite cookies," we can extract an event "(*eating*, *Kevin*, *favorite cookies*)." The detailed extraction steps will be illustrated in Section 4.3.2. Although the narrative event chains are extracted from the corpus, each event in a chain is a sparser and more discrete data format than the raw text. Hence, we employ a fine-tuning to reduce the gap between the intra-event sequences and the raw text.

Similarly, we adopt the masked language model to process the fine-tuning. The input format can be represented as follows:

$$[CLS]\, pred(e_1), subj(e_1), obj(e_2)\cdots, [mask], \cdots, obj(e_n),$$

where $[CLS]$ is a start token, $pred(e_i)$, $subj(e_i)$ and $obj(e_i)$ are the predicate, subject and object words in event $e_i$, respectively. We randomly mask some words in an intra-event sequence by a token $[mask]$.

In the intra-event encoder, we consider encoding the intra-event words from three aspects, i.e., the raw-text embeddings, the attribute embeddings and the position embeddings. For generating the raw-text embeddings, we directly feed the intra-event sequences into the fine-tuned bert-base-uncased model in the raw-text component. For generating the attribute embeddings, each word in an intra-event sequence is associated with three attribute types, i.e., predicate, subject and object, which can be regarded as the

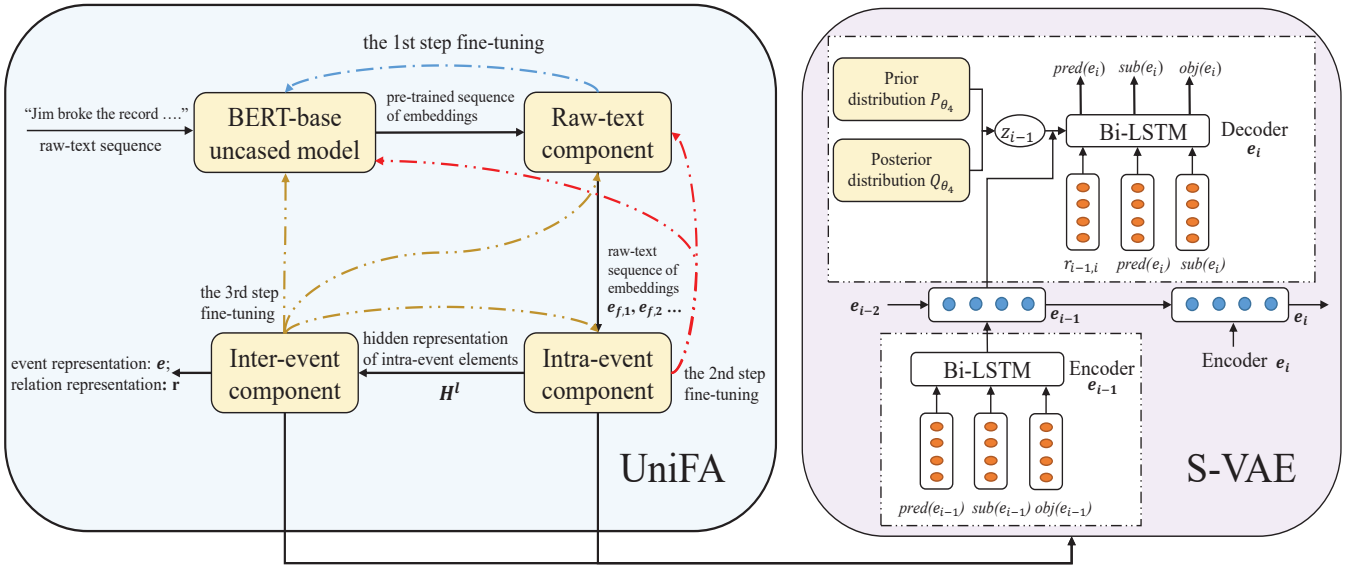



**Figure 2: The overall architecture of *UniFA-S***

additional features of word embeddings. Following [34], we embed positions of intra-event words for generating the position embeddings. By doing so, the final intra-event word representation $x_i$ is produced as follows:

$$x_i = e_{f,i} + e_{a,i} + e_{p,i}, \tag{2}$$

where $e_{f,i}$, $e_{a,i}$ and $e_{p,i}$ are the raw-text embeddings, the attribute embeddings and the position embeddings for the $i$-th word in an intra-event sequence, respectively.

After that, we feed the intra-event representation sequence $\boldsymbol{X}$ ($x_i \in \boldsymbol{X}$) into the multiple transformer blocks [34] to generate the hidden representations:

$$\begin{cases} \boldsymbol{H^1} = \text{Transformer}(\boldsymbol{X}) \\ \boldsymbol{H^i} = \text{Transformer}(\boldsymbol{H^{i-1}}) \end{cases}, \tag{3}$$

where Transformer $(\cdot)$ means a one-layer transformer block [34] and $\boldsymbol{H^i}(i \in \{1, 2, \cdots, l\})$ denotes the hidden representation of the $i$-th layer transformer block.

Accordingly, the loss of intra-event component $loss^{2nd}$ can be represented as

$$loss^{2nd} = -\sum_{\widetilde{w} \in \{[mask]\}} p(\widetilde{w}) \log \widetilde{w} + \lambda L(\theta_2), \tag{4}$$

where $\{[mask]\}$ is the set of masked words, $\lambda$ is the same parameter in Eq. (1), $L(\theta_2)$ is $L2$ regularization on parameters $\theta_2$ of the intra-event component. In addition, we have

$$p(\widetilde{w}) = p(\widetilde{w}|\boldsymbol{H^l}, a)p(a|\boldsymbol{H^l_{mask}}), \tag{5}$$

where $p(\widetilde{w}|\boldsymbol{H^l}, a)$ means the word probability conditioned by the hidden representation $\boldsymbol{H^l}$ and the attribute $a$ while $p(a|\boldsymbol{H^l_{mask}})$ denotes the attribute probability of the masked word.

*3.1.3* ***Inter-event component.*** The inter-event component explores the inter-event relations to enrich the event representation. Following [17], we adopt the discourse relations [28] to annotate the relation within two events. However, different from [17], we employ the Role-factor model [37] to encode event as it has been proven more effective than the event composition mode [37].

Given an event $e$, let $p(e), s(e), o(e)$ represent the embeddings of predicate, subject and object, respectively. Then, the event embeddings can be represented as follows:

$$e = W_s T(s(e), p(e)) + W_o T(o(e), p(e)) \tag{6}$$

where $W_s, W_o$ are the trade-off matrices for the subject role and object role, respectively. $T(a, b)$ denotes a tensor network [33] based on a 3-dimension weighted tensor $T$ with $a$ and $b$ as inputs.

Accordingly, the loss of inter-event component $loss^{3rd}$ can be represented as:

$$loss^{3rd} = \sum_{t \in T} \sum_{t^* \in T^*} \max(0, \delta + f(t) - f(t^*)) + \lambda L(\theta_3), \tag{7}$$

where $T$ and $T^*$ denote the set of positive and corrupted relational triplets, respectively, $\delta$ is the margin, $L(\theta_3)$ is $L2$ regularization on parameters $\theta_3$ of the inter-event component, and $\lambda$ is the same parameter in Eq. (1), In addition, $f(t)$ is formulated as

$$f(t) = f_{transe}((e_h, e_t, r_{h,t})) = ||e_h + r_{h,t} - e_t||_p^p, \tag{8}$$

where $e_h$, $r_{h,t}$ and $e_t$ are the head event, relation and tail event representation in a relation triplet, and $f_{transe}(a, b, c)$ is a *TransE* function [1] with $a$, $b$ and $c$ as inputs. In the knowledge graph representation, $a$, $b$ and $c$ denote head entity, relation and tail entity, respectively; $||\cdot||_p^p$ is a function of Euclidean distance with p-dimension.

*3.1.4* ***Multi-step fine-tuning.*** As shown in Fig. 2, *UniFA* is featured with a cascaded structure, that is, the output of a low component is the input of a high component. Accordingly, we argue that a lower component may contain more general knowledge than a higher component.

In view of this, we design a multi-step fine-tuning method to fine-tune *UniFA* as shown in Fig. 2:

**Algorithm 1** *Stage heating strategy.*

**Input:** The list of required fine-tuning components in a strict structure order from the bottom to the top: $C_r = \{C_1, C_2, \cdots, C_n\}$.
**Output:** The fine-tuned parameters $\psi$.

```
The heated list C_h = {}.
while C_r ≠ ∅ do
    Pop out the highest component C_temp in C_r.
    Push C_temp into C_h.
    Fine-tune (heating) all components in C_h at exponential-descent learning rates.
end while
```

- The $1^{st}$ step fine-tuning narrows the gap between the pre-trained bert-base-uncased model and the extracted corpus;
- The $2^{nd}$ step fine-tuning first decreases the domain difference between the intra-event sequences and the extracted corpus, and then promotes the $1^{st}$ step fine-tuning;
- The $3^{rd}$ step fine-tuning first reduces the gap between the inter-event sequences and the intra-event sequences, and then improves the $2^{nd}$ step fine-tuning.

Compared with the joint-training, this catastrophic fine-tuning can not only better capture the relations within each component, but avoid the local optimum introduced by the manual additive loss function. However, this catastrophic fine-tuning may cause a catastrophic forgetting if we fine-tune all layers and all components at once. To deal with it, we propose a *stage heating* fine-tuning strategy that simulates the *stage heating* process in chemistry [31]. We detail the main process in Algorithm 1. For instance, we have a heated list $C_h$ and a list of components $C_r$ to fine-tune, that are lined in a strict structure order from low to high. At each fine-tuning epoch, we first pop out the highest component $C_{temp}$ from $C_r$ in the current stage, and then push $C_{temp}$ into $C_h$. During the fine-tuning process, only the components in the heated list will be fine-tuned and the others will be remained unchanged. Different from "*gradual unfreezing*" [11] and "*chain-thaw*" [7] that push a layer into the "thaw layers" or train a layer simultaneously, our fine-tuning strategy adds an individual component at one time, which can improve the efficiency of fine-tuning while maintaining the component integrity.

In addition, as the more general the knowledge is, the slower the fine-tuning would be, we introduce the exponential-descent learning rates to grasp different layers of information. That is, when the networks reduce one layer, the learning rate will be dropped with a specific rate, i.e.,

$$\begin{cases} \eta_i^j = \eta_i^{j-1}/\alpha \\ \eta_i^j = \eta_{i-1}^j/\beta \end{cases}, \tag{9}$$

where $\eta_i^j$ denotes the learning rate for the $j$-th layer in the $i$-th component, $\alpha$ and $\beta$ are the learned parameters. Here, superscripts and subscripts are arranged in an increasing order from the bottom to the top.

## 3.2 Scenario variational auto-encoder

Variational auto-encoder (VAE) models have shown great generative ability in text generation tasks like dialogue generation [32] as it can capture a high-level feature by sampling a stochastic variable for text generation. Similarly, we design a scenario-level variation auto-encoder (*S-VAE*) to implicitly represent the scenario knowledge of an event chain.

As shown in the right part of Fig. 2, we adopt the Bi-LSTM networks in the encoder stage to capture the relations among intra-event elements, which is similar to Pair-LSTM [36] excluding the event composition and dynamic memory networks. Given an event chain $(e_1, \cdots, e_i)$, the prior distribution $P_{\theta_4}$ and the posterior distribution $Q_{\theta_5}$ of the stochastic latent variable $z_i$ can be formulated as follows:

$$\begin{cases} P_{\theta_4}(z_{i-1}|e_1, \cdots, e_{i-1}) = \mathbb{N}(\mu_p(e_1, \cdots, e_{i-1}), \Sigma_p(e_1, \cdots, e_{i-1})) \\ Q_{\theta_5}(z_{i-1}|e_1, \cdots, e_i) = \mathbb{N}(\mu_q(e_1, \cdots, e_i), \Sigma_q(e_1, \cdots, e_i)) \end{cases},$$

where $\mathbb{N}(\mu, \Sigma)$ is a multivariate normal distribution with the mean $\mu$ and the covariance matrix $\Sigma$. The prior distribution is conditioned on the historical events while the posterior distribution is conditioned on the historical events as well as the next event.

In the decoder stage, the event transition from $e_{i-1}$ to $e_i$ is bridged by a relation representation $r_{i-1,i}$. In addition, the decoder for event $e_i$ is conditioned on the previous observed event sequence $\{e_1, \cdots, e_{i-1}\}$ and the stochastic variable $z_{i-1}$ drawn from the prior distribution $P_{\theta_4}$. Such factors are denoted as *context*, i.e., $context = \{z_{i-1}, r_{i-1,i}, e_1, \cdots, e_{i-1}\}$. Therefore, we can design the decoder for the next event as

$$\begin{aligned} P_{\theta_4}(e_i|context) =& P_{\theta_4}(c_1|context) \cdot P_{\theta_4}(c_2|context, c_1) \\ & \cdot P_{\theta_4}(c_3|context, c_1, c_2), \end{aligned} \tag{10}$$

where $c_1$,$c_2$ and $c_3$ are $pred(e_i)$, $sub(e_i)$ and $obj(e_i)$, respectively. Actually, this decoder is similar to the variational encoder-decoder for dialog generation [32].

For training S-VAE, the evidence lower bound $L^{ELBO}(\theta_4, \theta_5)$ is often maximized, i.e.,

$$\begin{aligned} L^{ELBO}(\theta_4, \theta_5) = & \sum_{i=1}^{n} -KL[Q_{\theta_5}(z_i|e_1, \cdots, e_{i+1})||P_{\theta_4}(z_i|e_1, \cdots, e_i)] \\ & + \mathbb{E}_{Q_{\theta_5}(z_i|e_1, \cdots, e_{i+1})}[logP_{\theta_4}(e_i|z_{i-1}, e_1, \cdots, e_i)], \end{aligned} \tag{11}$$

where $KL[Q||P]$ is the Kullback-Leibler divergence between the distributions $Q$ and $P$. Hence, the loss of *S-VAE* can be formulated as follows:

$$loss^{4^{th}} = -L^{ELBO}(\theta_4, \theta_5) + \lambda L(\theta_4, \theta_5), \tag{12}$$

where $L(\theta_4, \theta_5)$ is $L2$ regularization on parameters $\theta_4$ and $\theta_5$ of the *S-VAE* component.

So far, our proposal *UniFA-S* in this paper is fully illustrated.

# 4 EXPERIMENTS SETUP

## 4.1 Evaluation tasks and datasets

We examine the performance of our proposal from two aspects: the representation ability and the inference ability.

*4.1.1* ***Representation ability.*** This ability aims to measure how well the event representation model can classify similar or dissimilar event pairs. It has two main types of tasks, hard similarity task [37] and transitive sentence similarity [6].

*Hard similarity task*, introduced by Weber et al. [37], consists of two types of event pairs: one with events that are close to each other but have little lexical overlap (e.g., *police catch robber / authorities apprehend suspect* ) and the other with events that are further apart but have a high overlap (e.g., *police catch robber / police catch disease*). This corresponding dataset (denoted as *Original* in this paper) contains 230 event pairs. Similar and dissimilar event pairs account for 50%, respectively. After that, Ding et al. [6] extended this dataset to 1000 event pairs with the same similar/dissimilar percentage (denoted as *Extended* in this paper). For evaluation metric, following [6, 37], we use Accuracy in this paper, which measures the percentage of cases where the similar pair receives a higher cosine value than the dissimilar pair.

*Transitive sentence similarity*, introduced by Ding et al. [6], is another version for event similarity evaluation. It is extracted from 108 pairs of transitive sentences on the transitive sentence similarity dataset [13]. Each pair is manually annotated by a similarity score from 1 to 7. A larger score indicates a higher similarity. For evaluation metrics, following [6], we adopt the Spearmans correlation ($\rho \in [-1, 1]$) to measure the cosine similarity between the scores generated by each model and the annotated ground-truth score given by human.

*4.1.2* ***Inference ability***. This ability concentrates on measuring how well the event representation can predict the next event in an event chain. According to the inference step, this ability can be classified into two types: one-step inference task and multi-step inference task. One-step inference task, e.g., multiple-choice narrative cloze (MCNC) [10], aims to predict a missing event given its context.

Extended from the one-step inference task, the multi-step inference task concentrates on predicting a sequence of events with some prior knowledge provided. In the process of constructing dataset, based on the differences of provided prior knowledge and the selection strategy for event chains, the multi-step inference task has four versions, i.e., MCNS-V, MCNE-V, Base and Sky. In particular, MCNS-V and MCNE-V denote multiple-choice narrative sequence (MCNS) [16] and multiple-choice narrative explanation (MCNE) [16] constructed by the *Viterbi* algorithm that considers the integrity of event chain and finds the most probable event chain. While Base is constructed by greedily picking the best transition and then moving to the next time stamp. Sky breaks down a sequence of prediction into individual decisions which applies the golden states of all contextual events. Furthermore, the above inference tasks are all in a multiple-choice setting, i.e., the event representation model should choose a positive event from one golden choice and four corrupted choices for each-step inference.

## 4.2 Model summary

According to the model taxonomy in Section 2.1, for each category, we select two recent proposals as baselines including the best baseline in the corresponding category, respectively.

**Event-comp:** an intra-event-based method that consists of intra-event elements based on a fully connected network [10].

**Role-factor:** an intra-event-based method that models multiplicative interactions among intra-event elements based on a tensor network [37].

**SAM-Net:** an inter-event-based method that explores the event-segment relations [21].

**EventTransE:** an inter-event-based method that explores the inter-event relation based on the discourse relations [17].

**FEEL:** an external-knowledge-based method that introduces the sentiment and animacy information [16].

**IntSent:** an external-knowledge-based method that introduces the intent and sentiment information [6].

Next, we list the models proposed in this paper for comparison.

***UniFA-S:*** an ensemble model that adds *S-VAE* into *UniFA* as the component.

***UniFA-S*$_{[\mathrm{FT}]}$:** a variant of *UniFA-S* applying a fine-tuning method [*FT*], e.g., jointly-training (JT), *chain-thaw* (CT), *gradual unfreezing* (GU) and multi-step fine-tuning (MF).

***UniFA-S*$_{w/o[C]}$:** a variant of *UniFA-S* without a particular component [*C*], e.g., the raw-text component (RT), the intra-event component (ItR), the inter-event component(ItE) and the scenario component (Sc).

## 4.3 Research questions and configurations

*4.3.1* **Research question.** We examine the effectiveness of our proposal and focus on the following research questions to guide our experiments:

**RQ1:** Dose our proposal improve the representation ability compared to the existing baselines?

**RQ2:** Dose our proposal promote the inference ability compared to the previous baselines?

**RQ3:** How is the performance of event representation model impacted by different fine-tuning methods, e.g., joint-training and multi-step fine-tuning? Does our proposed multi-step fine-tuning boost the representation and inference ability?

**RQ4:** How is the performance of our proposal with different number of event steps as context in the one-step inference task?

**RQ5:** How is the performance of our proposal with the increase of inference steps in the multi-step inference tasks?

**RQ6:** Which component contributes the most to the representation and inference abilities?

*4.3.2* **Model configuration.** Following [10, 16, 17], we choose the New York Times portion of the Gigaword corpus [1] as the raw-text corpus. In addition, we use the Stanford CoreNLP [22] to extract the dependency parses and coreference chains. Based on the coreference chains, we creat the event chains in the form of $(pred, subj, obj)$. For the extraction of intra-event words, we keep the complete mention spans rather than only headwords [17]. Finally, we select 1.4M event chains as the training set, 10K event chains as the development set and 10K event chains as the test set.

During training, we set batch size to 128 and regularization weight to $10^{-5}$. We adopt the Adam Optimizer [14] with exponential-descent learning rate to optimizer our models. Specially, we find it work well when $\alpha = 2.6$, $\beta = 3.0$ and $\eta_0^0 = 5 \times 10^{-5}$. As for word embeddings, we adopt the pre-trained bert-base-uncased version to initialize the model and refer readers to Devlin et al. [5] for details. Other weighted or trade-off matrices are initialized with Xavier Initialization [8].

[1] https://catalog.ldc.upenn.edu/LDC2003T05

**Table 1: Representation performance in terms of Accuracy (%) and $\rho$. The results produced by the best baseline and the best performer in each column are underlined and boldfaced, respectively. Statistical significance of pairwise differences of *UniFA-S* vs. the best baseline is determined by a $t$-test ($^{\blacktriangle}$ for $\alpha$ = .01, or $^{\triangle}$ for $\alpha$ = .05).**

| | Hard similarity (Accuracy %) | | Transitive sentence |
|---|---|---|---|
| Model | *Original* [37] | *Extended* [6] | similarity ($\rho$) |
| Event-comp | 33.9 | 18.7 | 0.57 |
| Role-factor | 43.5 | 20.7 | 0.64 |
| SAM-Net | 51.3 | 45.2 | 0.59 |
| EventTransE | 53.7 | 48.1 | 0.65 |
| FEEL | 58.7 | 50.7 | 0.67 |
| IntSent | 77.4 | 62.8 | 0.74 |
| *UniFA* | 75.8 | 61.2 | 0.71 |
| *S-VAE* | 68.9 | 59.3 | 0.69 |
| ***UniFA-S*** | **78.3**$^{\blacktriangle}$ | **64.1**$^{\triangle}$ | **0.75**$^{\triangle}$ |

**Table 2: Inference performance in terms of Accuracy (%). The results produced by the best baseline and the best performer in each column are underlined and boldfaced, respectively. Statistical significance of pairwise differences of *UniFA-S* vs. the best baseline is determined by a $t$-test ($^{\blacktriangle}$ for $\alpha$ = .01, or $^{\triangle}$ for $\alpha$ = .05).**

| | One-step inference | Multi-step inference | | | |
|---|---|---|---|---|---|
| Model | MCNC | MCNS-V | Base | Sky | MCNE-V |
| Event-comp | 46.3 | 29.9 | 27.8 | 38.4 | 32.5 |
| Role-factor | 48.8 | 28.6 | 28.3 | 39.6 | 32.5 |
| SAM-Net | 54.3 | 46.2 | 43.2 | 50.4 | 49.2 |
| EventTransE | 63.7 | 59.5 | 51.2 | 64.5 | 60.9 |
| FEEL | 51.6 | 41.6 | 38.5 | 46.0 | 44.8 |
| IntSent | 56.4 | 44.7 | 42.2 | 49.6 | 48.5 |
| *UniFA* | 65.2 | 63.2 | 55.0 | 65.8 | 63.4 |
| *S-VAE* | 60.2 | 56.6 | 50.5 | 59.7 | 60.1 |
| ***UniFA-S*** | **66.3**$^{\blacktriangle}$ | **64.0**$^{\blacktriangle}$ | **55.4**$^{\triangle}$ | **67.2**$^{\triangle}$ | **64.5**$^{\blacktriangle}$ |

# 5 RESULTS AND ANALYSIS

## 5.1 Representation ability

To answer **RQ1**, we examine the representation ability of our proposal as well as the baselines. We present the representation performance of discussed models in Table 1.

For the baselines as shown in Table 1, we can find that the external-knowledge-based models, i.e., FEEL and IntSent, present an obvious superiority over the other models for the representation tasks, i.e., hard similarity and transitive sentence similarity. For instance, IntSent shows near 44.1% and 30.6% improvements against EventTransE for the hard similarity task on the *Original* and *Extended* datasets, respectively. For the transitive sentence similarity task, IntSent presents a 13.9% improvement in terms of $\rho$ against EventTransE. In addition, improvements of IntSent against the intra-event based approaches are relatively higher than that of IntSent against EventTransE. Such dominant performance could be explained by the fact that the incorporation of external knowledge can provide additional context information, which deals with the sparseness issue well in representing the semantically deficient event and thus helps to substantially boost the representation ability. Following the external-knowledge based models, the inter-event based models generally outperform the intra-event based models for both tasks. It indicates that exploring the inter-event relation can better distinguish the difference of event than the intra-event interactions.

Next, we zoom in on the comparisons of our proposal against the best baseline IntSent. First of all, we can find that, in general, *UniFA* outperforms *S-VAE* for both tasks presented in Table 1. For instance, *UniFA* presents an improvement of 10.0%, 3.2% and 2.9% against *S-VAE* for the respective hard similarity task and the transitive sentence similarity task, which means that compared with the scenario-level training, the intra-and-inter event trainings can better generate the event representation, returning higher scores of Accuracy and $\rho$. However, comparing the results of these two individual models with that of the best baseline IntSent, our proposed *UniFA* and *S-VAE* lose the competitions. It implies that only modeling the intra-and-inter relationship of events cannot catch up with the benefit brought by external knowledge. However, our ensemble model *UniFA-S* can finally beat IntSent, showing 1.2%, 2.1%, 1.4% improvements in terms of accuracy and $\rho$ against IntSent for both tasks, respectively.

## 5.2 Inference ability

For answering **RQ2**, we examine the inference ability of our proposal as well as the baselines for the one-step task (i.e., MCNC) and the multi-step (i.e., MCNS-V, Base, Sky and MCNE-V) inference task, respectively. For comparison, we present the experimental results in Table 2.

Similarly, we first zoom in the baselines. Different from the results of representation ability in Table 1, the dominating baseline for evaluating the inference ability is EventTransE, i.e., an inter-event based model. In particular, in terms of Accuracy, EventTransE presents near 30.5% − 108.0% improvements against Role-factor (i.e., the winner in the intra-event based group) and 12.9% − 33.1% improvements against IntSent (i.e., the winner in the external-knowledge based group) for various tasks. It could be explained the fact that the inter-event relations may guide the event transition and then boost the inference ability. In addition, we find that the external-knowledge based models can keep superiority over the intra-event based models.

Next, we focus on comparing the results of our proposal against that of the baselines. Compared with the best baseline, i.e., EventTransE, *UniFA* shows 2.4%, 6.2%, 7.4%, 2.0%, 4.1% improvements in terms of Accuracy for the corresponding MCMC, MCNS-V, Base, Sky and MCNE-V tasks, respectively. The largest improvement in Base may be due to the following factors: Base is constructed by greedily picking the best transition; in this dataset, our *UniFA* can magnify the potential of the inter-event component. Regarding to *S-VAE*, it wins the competitions against the intra-event based models but loses against EventTransE, implying that only representing

the scenario-level knowledge cannot better improve the inference ability than exploring the inter-event relations. However, our ensemble model *UniFA-S* can generally beat EventTransE, presenting near $4.1-8.2\%$, $0.7-2.1\%$, $7.3-13.1\%$ improvements in terms of Accuracy against EventTransE, *UniFA* and *S-VAE*, respectively. It demonstrates that modeling the scenario-level knowledge makes sense when incorporated with the proposed unified fine-tuning architecture, i.e., *UniFA*.

## 5.3 Utility of multi-step fine-tuning

Next, we zoom in on **RQ3**, in order to verify the utility of multi-step fine-tuning, which connects all components in *UniFA-S*, we examine the representation and inference performance of *UniFA-S* under different fine-tuning methods, i.e., existing fine-tuning approaches like *joint training*, *chain-thaw layers* [7], *gradual unfreezing* [11] as well as our proposed *multi-step fine-tuning* approach. Table 3 presents the representation and inference performance of our propossal and other baseline fine-tuning methods.

Let us first examine the representation ability. As shown in Table 3, we find that *UniFA-S* with *joint training* shows the worst performance, which may be due to the fact that a simple summation of different levels of training losses is easily trapped in the local optimist. Regarding to other existing fine-tuning methods, *gradual unfreezing* is more effective than *chain-thaw*. In other words, *gradual unfreezing* is the best one among the baseline fine-tuning methods. Furthermore, in the comparison between our proposal (*UniFA-S* with *mulit-step fine-tuning*) and the best baseline (*UniFA-S* with *gradual unfreezing*), our proposal has 1.4% and 2.1% improvements in terms of Accuracy for the hard similarity task on the respective *Original* and *Extended* datasets, and 2.7% promotion in terms of $\rho$ for the transitive sentence similarity task.

Next, we move to the inference ability. Similar findings can be returned for all inference tasks. Again, the performance of *UniFA-S* with *joint training* fall behind that of *UniFA-S* with other fine-tuning methods. In addition, the performance of *UniFA-S* with *gradual unfreezing*, exceeds that of *chain-thaw*, presenting $2.8\%-6.9\%$ improvements in terms of Accuracy. Regarding to our proposed multi-step fine-tuning, *UniFA-S* with *multi-step fine-tuning* can consistently keep superiority over *UniFA-S* with other fine-tuning methods. In particular, *UniFA-S* with multi-step fine-tuning achieves $0.4-1.7\%$ improvement in terms of Accuracy against *UniFA-S* with *gradual unfreezing*.

## 5.4 Utility in one-step inference task

Next, for answering **RQ4** to see the impact of the length of event chain for the one-step inference task, we group the prediction results made by our proposal as well as the best baseline, i.e., EventTransE, that is found by the experiments in Section 5.2, according to the length of event chain denoted as $l_c$ ($l_c \in \{1, 2, \cdots, 6\} \cup l_c \geq 7$). We plot the results in Fig. 3; where event steps 7 represents $l_c \geq 7$.

Clearly, as shown in Fig. 3, when the number of event steps increases, the performance of these discussed models mainly keeps the same trend overall. That is, the accuracy will first increase obviously and then reach a plateau. It indicates that the accuracy for event prediction will keep unchanged when the context involves too many event steps. In other words, the discussed models can utilize limited information from multiple event steps as context to predict the next event. Interestingly, comparing the growth-stagnation point $p$ and growth range $r$, we can find that the growth mode of these models can be broadly classified into three groups. The first mode is the early-and-small growth mode that has an earlier stagnation point (e.g., $p \leq 3$) and a small growth range (e.g., $r \leq 1$), which corresponds to *UniFA*. The second mode is the just-and-medium growth mode that stops in time point ($p = 4$) and has a medium growth ($1 < r \leq 2$), which corresponds to *UniFA-S* and EventTransE. The third mode is the late-and-large growth mode that has a late stagnation point ($p \geq 5$) and a large growth range ($r > 2$), which corresponds to *S-VAE*. Although *S-VAE* has a worst start point, the stability and sustainability of growth mode makes *S-VAE* competitive for the inference task. Specially, the combination between the early-and-small growth mode of *UniFA* and the late-and-large growth mode of *S-VAE* generates the state-of-the-art model *UniFA-S* with the just-and-medium growth mode.

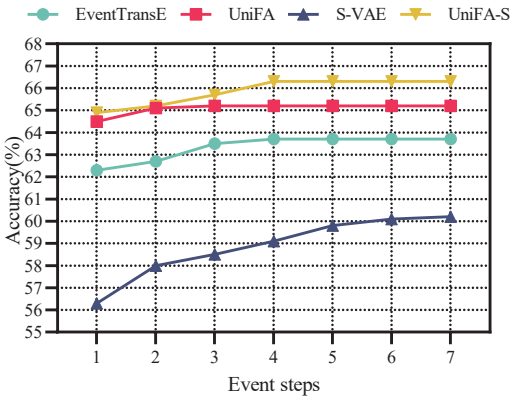



**Figure 3: Effect on performance of our proposals and the best baseline EventTransE in terms of Accuracy brought by different number of event steps as context for MCNC.**

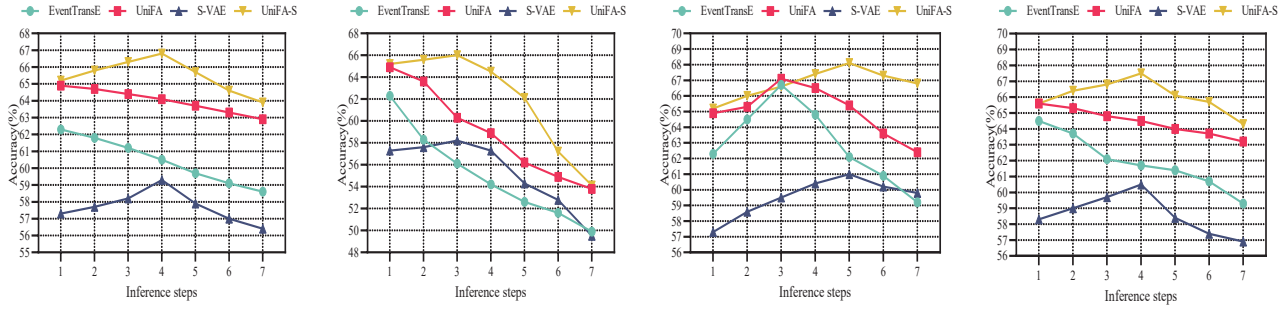
## 5.5 Utility in multi-step inference task

For answering **RQ5** to see the impact of the number of inference steps for four multi-step inference tasks, we compare the performance of our proposals and the best baseline EventTransE. Similarly, we group the prediction results according to the number of inference steps denoted as $s_t$ ($s_t \in \{1, 2, \cdots, 6\} \cup s_t \geq 7$) and plot the results in Fig. 4 for four versions of the multi-step inference tasks, where inference steps 7 represents $s_t \geq 7$.

As shown in Fig. 4, in general, *UniFA-S* achieves the best performance for different number of inference steps among four multi-step inference tasks. Besides, performance of each discussed model has an obvious descending trend despite some fluctuations, which stems from the fact that the more step the model infer, the more complicated the prediction will be. For MCNS-V, when the number of inference steps increases, the performance of *UniFA* and EventTransE shows a descending trend; while the performance of *S-VAE* and *UniFA-S* begins to arise at first and then decreases, which maintains a descending trend in general. These differences may be due to the fact that more inferred event steps may lead to generate more abundant context for inference tasks. Therefore the late-and-large

**Table 3: Utility of multi-step fine-tuning. The results produced by the best existing fine-tuning method and the best performer in each column are underlined and boldfaced, respectively. Statistical significance of pairwise differences of *UniFA-S*$_{[MF]}$ vs. the best baseline fine-tuning is determined by a $t$-test ($^{\blacktriangle}$ for $\alpha$ = .01, or $^{\triangle}$ for $\alpha$ = .05).**

| Model | Representation ability | | | Inference ability | | | | |
|---|---|---|---|---|---|---|---|---|
| | Hard similarity (Accuracy %) | | Transitive sentence | One-step inference | Multi-step inference | | | |
| | *Original* [37] | *Extended* [6] | similarity $\rho$ | MCNC | MCNS-V | Base | Sky | MCNE-V |
| *UniFA-S*$_{[JT]}$ | 69.2 | 59.4 | 0.69 | 60.7 | 57.3 | 51.6 | 60.5 | 60.3 |
| *UniFA-S*$_{[CT]}$ | 72.7 | 60.1 | 0.70 | 63.8 | 59.4 | 52.9 | 62.6 | 60.9 |
| *UniFA-S*$_{[GU]}$ | 77.2 | 62.8 | 0.73 | 65.6 | 63.5 | 55.2 | 66.1 | 63.0 |
| ***UniFA-S*$_{[MF]}$** | **78.3**$^{\blacktriangle}$ | **64.1**$^{\triangle}$ | **0.75**$^{\triangle}$ | **66.3**$^{\blacktriangle}$ | **64.0**$^{\triangle}$ | **55.4**$^{\triangle}$ | **67.2**$^{\triangle}$ | **63.8**$^{\triangle}$ |




**(a) MCNS-V** **(b) Base** **(c) Sky** **(d) MCNE-V**

**Figure 4: Effect on the performance of our proposals as well as the best baseline EventTransE in terms of Accuracy brought by different number of event steps as context for MCNS-V, Base, Sky and MCNE-V, respectively.**

growth mode (*S-VAE*) can alleviate the prediction decline temporarily, resulting in a growing accuracy. Specially, *UniFA-S* can resist the downward trend by depending on the *S-VAE* part, despite the just-and-medium growth mode. Similarly, the performance of all discussed models in MCNE-V has similar change trends except the overall accuracy. Since MCNE-V provides an extra event (i.e., the ending event) compared with MCNS-V, the same model in MCNE-V has a higher accuracy performance than that in MCNS-V.

For Base, all discussed models have a steadily descending trend of performance, which may be caused by the characteristic of dataset itself. That is, Base, constructed by picking the best transition of event pair, relies on the previous event to predict the next event rather than the context of event chains. Hence, when the number of inference steps increases, all discussed models cannot resist the downward trend. Different from Base, Sky is constructed by the best local decision according to the event context, which implies that the resistant ability brought by event context for all discussed models can be amplified. Therefore, all discussed model have a temporary rising performance to reach the highest accuracy and then keep the descending trend in general.

## 5.6 Ablation study

To answer **RQ6**, we perform an ablation study to get a deep insight into each component of our proposed *UniFA-S*. We remove a certain component of *UniFA-S* and examine the corresponding performance of the incomplete *UniFA-S* for all discussed tasks. In particular, there are four individual *UniFA-S* models we want to investigate, i.e., *UniFA-S*$_{w/o[Sc]}$ that removes the scenario component, *UniFA-S*$_{w/o[ItE]}$ that replaces the inter-event component with the event composition model [10] to directly generate the event representation, *UniFA-S*$_{w/o[ItR]}$ that removes the intra-event component and directly uses the fine-tuned bert-uncased model to get the word embeddings, and *UniFA-S*$_{w/o[RT]}$ that replace the raw-text component with the pre-trained Bert embeddings. We compute the relative performance change rates of the incomplete *UniFA-S* model against the integrated *UniFA-S* model. Each ablation result is computed using the average of relative change rates for the representation ability and the inference ability, respectively [35]. We present the results in Table 4.

**Table 4: Ablation performance in terms of the representation and inference abilities. The most important component in each column are boldfaced.**

| Model | Representation ability | Inference ability |
|---|---|---|
| *UniFA-S*$_{w/o[Sc]}$ | -4.6% | -10.6% |
| *UniFA-S*$_{w/o[ItE]}$ | -5.7% | **-11.2%** |
| *UniFA-S*$_{w/o[ItR]}$ | **-8.5%** | -5.6% |
| *UniFA-S*$_{w/o[RT]}$ | -2.9% | -3.3% |

As shown in Table 4, we can find that the intra-event component contributes the most to the representation ability, which is followed by the inter-event component and the scenario component. The least influential component for the representation ability is the raw-text component. These findings demonstrate that enriching the intra-event representation is the most effective way to boost the representation ability. As to the inference ability, the most influential component in our proposal is the inter-event component, which is closely followed by the scenario component. The close

relative change rate verifies that the effectiveness of our proposed scenario component.

## 6 CONCLUSION AND FUTURE WORK

Our work aims at learning the distributed event representation. We propose a unified fine-tuning architecture incorporated with scenario knowledge, consisting of a unified fine-tuning architecture (*UniFA*) and a scenario-level variational auto-encoder (*S-VAE*). In detail, *UniFA*, which consists of four key components, employs a multi-step fine-tuning to consider all levels of training. In addition, in order to tackle the existing models lack of modeling the event context, we design a *S-VAE* method to implicitly represent the scenario-level knowledge. Experimental results on eight individual datasets show the advantages of our proposal on improving the representation and inference abilities.

As to future work, on the one hand, we plan to investigate how to incorporate the external knowledge into our proposed event architecture, as external knowledge has been proven effective in terms of boosting the representation ability in Section 5.1. On the other hand, we plan to extend the *S-VAE* model with diverse event relations to generate more informative scenario knowledge.

## 7 ACKNOWLEDGMENTS

This work was partially supported by the National Natural Science Foundation of China under No. 61702526, the Defense Industrial Technology Development Program under No. JCKY2017204B064. All content represents the opinion of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.

## REFERENCES

[1] Antoine Bordes, Nicolas Usunier, Alberto García-Durán, Jason Weston, and Oksana Yakhnenko. 2013. Translating Embeddings for Modeling Multi-relational Data. In *NIPS*. MIT Press, 2787–2795.
[2] Nathanael Chambers and Daniel Jurafsky. 2008. Unsupervised Learning of Narrative Event Chains. In *ACL*. ACL, 789–797.
[3] Nathanael Chambers and Dan Jurafsky. 2009. Unsupervised Learning of Narrative Schemas and their Participants. In *ACL*. ACL, 602–610.
[4] Andrew M. Dai and Quoc V. Le. 2015. Semi-supervised Sequence Learning. In *NIPS*. MIT Press, 3079–3087.
[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT*. ACL, 4171–4186.
[6] Xiao Ding, Kuo Liao, Ting Liu, Zhongyang Li, and Junwen Duan. 2019. Event Representation Learning Enhanced with External Commonsense Knowledge. In *EMNLP-IJCNLP*. ACL, 4893–4902.
[7] Bjarke Felbo, Alan Mislove, Anders Søgaard, Iyad Rahwan, and Sune Lehmann. 2017. Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. In *EMNLP*. ACL, 1615–1625.
[8] Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *AISTATS (JMLR Proceedings)*, Vol. 9. JMLR.org, 249–256.
[9] David Graff, Kong Junbo, and Chenand Kazuaki Maeda Ke. 2003. English Gigaword. *Linguistic Data Consortium, Philadelphia* 4(2) (2003), 34.
[10] Mark Granroth-Wilding and Stephen Clark. 2016. What Happens Next? Event Prediction Using a Compositional Neural Network Model. In *AAAI*. AAAI Press, 2727–2733.
[11] Jeremy Howard and Sebastian Ruder. 2018. Universal Language Model Fine-tuning for Text Classification. In *ACL*. ACL, 328–339.
[12] Rongtao Huang, Bowei Zou, Hongling Wang, Peifeng Li, and Guodong Zhou. 2019. Event Factuality Detection in Discourse. In *NLPCC*, Vol. 11839. Springer, 404–414.
[13] Dimitri Kartsaklis and Mehrnoosh Sadrzadeh. 2014. A Study of Entanglement in a Categorical Framework of Natural Language. In *QPL (EPTCS)*, Vol. 172. 249–261.
[14] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *ICLR*. OpenReview.net.
[15] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. *CoRR* abs/1909.11942 (2019). arXiv:1909.11942
[16] I-Ta Lee and Dan Goldwasser. 2018. FEEL: Featured Event Embedding Learning. In *AAAI*. AAAI Press, 4840–4847.
[17] I-Ta Lee and Dan Goldwasser. 2019. Multi-Relational Script Learning for Discourse Relations. In *ACL*. ACL, 4214–4226.
[18] Feng-Lin Li, Kehan Chen, Yan Wan, Weijia Chen, Qi Huang, and Yikun Guo. 2019. Using Event Graph to Improve Question Answering in E-commerce Customer Service. In *ISWC*, Vol. 2456. CEUR-WS.org, 327–328.
[19] Zhongyang Li, Xiao Ding, and Ting Liu. 2018. Constructing Narrative Event Evolutionary Graph for Script Event Prediction. In *IJCAI*. ijcai.org, 4201–4207.
[20] Xiao Liu, Heyan Huang, and Yue Zhang. 2019. Open Domain Event Extraction Using Neural Latent Variable Models. In *ACL*. ACL, 2860–2871.
[21] Shangwen Lv, Wanhui Qian, Longtao Huang, Jizhong Han, and Songlin Hu. 2019. SAM-Net: Integrating Event-Level and Chain-Level Attentions to Predict What Happens Next. In *AAAI*. AAAI Press, 6802–6809.
[22] Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP Natural Language Processing Toolkit. In *ACL*. ACL, 55–60.
[23] Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. 2017. Learned in Translation: Contextualized Word Vectors. In *NIPS*. MIT Press, 6297–6308.
[24] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. In *ICLR*. OpenReview.net.
[25] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global Vectors for Word Representation. In *EMNLP*. ACL, 1532–1543.
[26] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep Contextualized Word Representations. In *NAACL-HLT*. ACL, 2227–2237.
[27] Karl Pichotta and Raymond J. Mooney. 2014. Statistical Script Learning with Multi-Argument Events. In *EACL*. ACL, 220–229.
[28] Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind K. Joshi, and Bonnie L. Webber. 2008. The Penn Discourse TreeBank 2.0. In *LREC*. European Language Resources Association.
[29] Schank Roger, C and Abelson Robert, P. 1977. Scripts, plans, goals, and understanding: An inquiry into human knowledge structures. *Lawrence Erlbaum Associates* (1977).
[30] Maarten Sap, Ronan Le Bras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A. Smith, and Yejin Choi. 2019. ATOMIC: An Atlas of Machine Commonsense for If-Then Reasoning. In *AAAI*. AAAI Press, 3027–3035.
[31] Ali Sarhadi, Jesper Henri Hattel, Hans Nørgaard Hansen, Cem Celal Tutum, Lasse Lorenzen, and Peter M.W. Skovgaard. 2012. Thermal modelling of the multi-stage heating system with variable boundary conditions in the wafer based precision glass moulding process. *Journal of Materials Processing Technology* 212, 8 (2012), 1771–1779.
[32] Iulian Vlad Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron C. Courville, and Yoshua Bengio. 2017. A Hierarchical Latent Variable Encoder-Decoder Model for Generating Dialogues. In *AAAI*. AAAI Press, 3295–3301.
[33] Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. 2013. Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank. In *ACL*. ACL, 1631–1642.
[34] 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 *NIPS*. MIT Press, 5998–6008.
[35] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2018. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In *EMNLP*. ACL, 353–355.
[36] Zhongqing Wang, Yue Zhang, and Ching-Yun Chang. 2017. Integrating Order Information and Event Relation for Script Event Prediction. In *EMNLP*. ACL, 57–67.
[37] Noah Weber, Niranjan Balasubramanian, and Nathanael Chambers. 2018. Event Representations With Tensor-Based Compositions. In *AAAI*. AAAI Press, 4946–4953.
[38] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. XLNet: Generalized Autoregressive Pretraining for Language Understanding. *CoRR* abs/1906.08237 (2019). arXiv:1906.08237
[39] Jianming Zheng, Fei Cai, Wanyu Chen, Chong Feng, and Honghui Chen. 2019. Hierarchical Neural Representation for Document Classification. *Cognitive Computation* 11, 2 (2019), 317–327.
[40] Yukun Zhu, Ryan Kiros, Richard S. Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning Books and Movies: Towards Story-Like Visual Explanations by Watching Movies and Reading Books. In *ICCV*. IEEE Computer Society, 19–27.