Multi-Relational Probabilistic Event Representation Learning via Projected Gaussian Embedding

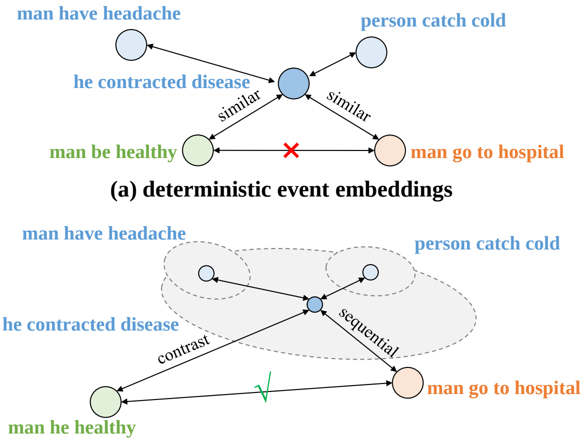
Linhai Zhang Congzhi Zhang Deyu Zhou[*]
School of Computer Science and Engineering, Key Laboratory of Computer Network
and Information Integration, Ministry of Education, Southeast University, China
{lzhang472, zhangcongzhi, d.zhou}@seu.edu.cn

Abstract

Event representation learning has been shown beneficial in various downstream tasks. Current event representation learning methods, which mainly focus on capturing the semantics of events via deterministic vector embeddings, have made notable progress. However, they ignore two important properties: the multiple relations between events and the uncertainty within events. In this paper, we propose a novel approach to learning multi-relational probabilistic event embeddings based on contrastive learning. Specifically, the proposed method consists of three major modules, a multi-relational event generation module to automatically generate multi-relational training data, a probabilistic event encoding module to model uncertainty of events by Gaussian density embeddings, and a relation-aware projection module to adapt unseen relations by projecting Gaussian embeddings into relation-aware subspaces. Moreover, a novel contrastive learning loss is elaborately designed for learning the multi-relational probabilistic embeddings. Since the existing benchmarks for event representation learning ignore relations and uncertainty of events, a novel dataset named MRPES is constructed to investigate whether multiple relations between events and uncertainty within events are learned. Experimental results show that the proposed approach outperforms other state-of-the-art baselines on both existing and newly constructed datasets.

1 Introduction

Events, carrying world knowledge, are the major research targets in Natural Language Processing (NLP) for decades. Distributed event representation learning has been shown beneficial in various NLP tasks, such as sentiment analysis (Zhou et al., 2021), event detection (Deng et al., 2021) and text generation (Chen et al., 2021).



(a) deterministic event embeddings

(b) multi-relational probabilistic event embeddings

Figure 1: An example in the MRPES dataset, where shaded areas represent confidence intervals of the density embeddings.

Early event representation learning methods mainly focused on the way of composing event components, such as by Multilayer Perceptrons (Granroth-Wilding and Clark, 2016), Recurrent Neural Networks (Modi, 2016), and Tensor Networks (Weber et al., 2018). Latter work tried to incorporate various external knowledge into event representation learning, such as knowledge graphs (Ding et al., 2016), extra event features (Lee and Goldwasser, 2018), or commonsense knowledge (Ding et al., 2019). Recently, Gao et al. (2022) showed the effectiveness of incorporating contrastive learning (Chen et al., 2020) in event representation learning by simultaneously utilizing weakly supervised contrastive learning and prototype-based clustering. So far, similar to word representation learning (Mikolov et al., 2013; Pennington et al., 2014), current approaches for event representation learning mainly aim to capture the semantics of events based on large-scale co-occurrence training data by making the semantically-similar events closer in embedding

[*] Corresponding author.

Findings of the Association for Computational Linguistics: ACL 2023, pages 6162–6174
July 9-14, 2023 ©2023 Association for Computational Linguistics

space.

Though notable progress has been made, most existing methods still have two limitations. On the one hand, they ignore the multiple relations between events, which means every event pair that occurs together will be pushed closer whatever the actual relation between them. As shown in Figure 1(a), both event $(man\ go\ to\ hospital)$ and event $(man\ be\ healthy)$ will be pushed closer to event $(he\ contracted\ disease)$, which means they will be pushed closer too. It is problematic as they are semantically different. On the other hand, the inherent uncertainty or polysemy of events is ignored. A more general event is used to describe more situations, reflecting higher uncertainty in its meaning (Athiwaratkun and Wilson, 2018; Zhang et al., 2021). As shown in Figure 1(a), event $(he\ contracted\ disease)$ is a general description for illness, which is semantically similar to two specific events $(man\ have\ headache)$ and $(person\ catch\ cold)$ which are different from each other. However, restricted by the triangle inequality, it is hard for a vector embedding to be close to the other two points that are apart from each other.

In this paper, we propose a Multi-relatiOnal pRobabilistic Event embedding method based on Contrastive Learning (MORE-CL) to solve the above limitations. Specifically, we utilize COMET (Bosselut et al., 2019), a commonsense generative model, to generate multi-relational positive samples for contrastive training. A probabilistic event encoder based on BERT (Devlin et al., 2019) is proposed to generate Gaussian event embeddings by estimating the mean vector and variance matrix. To deal with unseen relations, a relation-aware projector is employed to determine the relation-based event pair context automatically with an attention mechanism and project the density embeddings into relation-specific subspaces. Finally, the original InfoNCE loss (Oord et al., 2018) is modified to learn multi-relational probabilistic embedding. To investigate the effectiveness of the proposed method, a multi-relational probabilistic event similarity dataset named MRPES is constructed. Experimental results show that MORE-CL outperforms other baselines by a large margin on both original and new benchmark datasets.

In conclusion, our contributions are three-fold:

- A novel method, MORE-CL, is proposed to model the multiple relations between events and the uncertainty within events using projected Gaussian density embeddings with contrastive learning.
- A multi-relational probabilistic event similarity dataset named MRPES is constructed and annotated to evaluate whether the multiple relations between events and the uncertainty within events are learned.
- Experimental results show the effectiveness of the proposed method on both original and newly constructed benchmark datasets.

2 Related Work

Event Representation Learning. Most existing event representation learning methods aim to project textual event descriptions represented into a dense vector where the semantic information of events is preserved as much as possible. Previous works either explored ways to effectively compose event components such as by tensor network (Ding et al., 2015; Weber et al., 2018) or external knowledge to improve the learning of event embeddings (Ding et al., 2016; Lee and Goldwasser, 2018; Ding et al., 2019). Besides textual signal, Zhang et al. (2021) proposed to utilize event images as external knowledge. Generally, they can be categorized as methods learned by the margin loss based on a pair of a positive sample and a negative sample. Recently, Gao et al. (2022) explored contrastive learning (Chen et al., 2020) in event representation learning, which outperformed previous margin loss-based methods by a large margin, showing the effectiveness of contrastive learning in this task. It should be pointed out that SWCC proposed by (Gao et al., 2022) implicitly captures relation information by performing prototype-based clustering. However, SWCC is not trained on relational data explicitly and only captures one relation between the events. Our work follows this line of research and makes improvements by considering multiple relations between events and uncertainty within events.

Script Event Prediction. A task closely related to event representation learning is script event prediction or script learning. Script learning focuses on modeling a sequence of events and predicting what will happen next. Previous works on script learning mainly focused on different neural architectures to learn event embeddings and model the sequence, such as MLP (Modi, 2016; Granroth-Wilding and

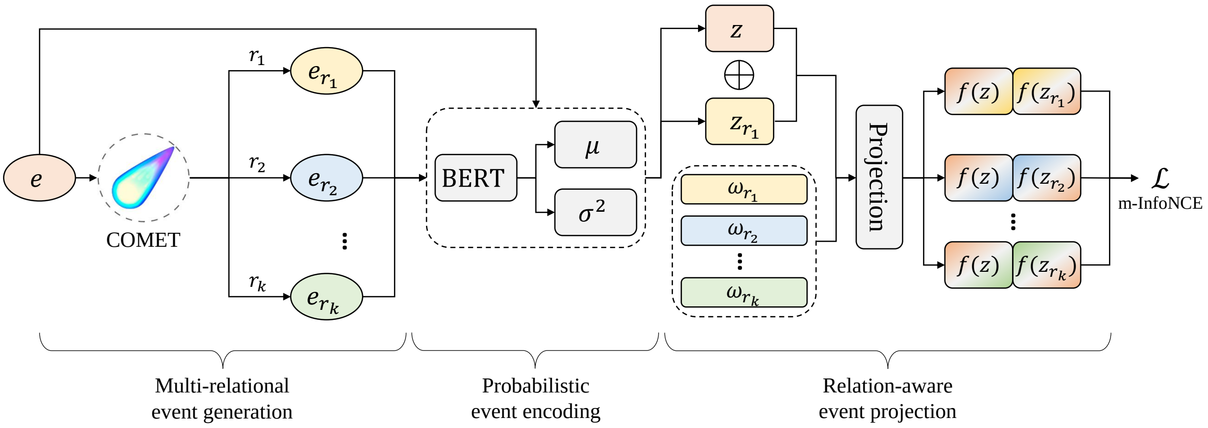


Figure 2: The overall architecture of MORE-CL, best viewed in colors.

Clark, 2016) and LSTM (Pichotta and Mooney, 2016b,a). Recently, some works tried to enhance script learning by incorporating knowledge of various discourse relations (Lee and Goldwasser, 2019; Zheng et al., 2020), which is similar to our work. However, our work mainly focuses on the modeling of the event itself instead of the sequence.

3 Method

As shown in Figue 2, MORE-CL consists of three modules. Firstly, the training events extracted from a large corpus are fed into the multi-relational event generation module to generate positive sample events for contrastive learning. Then, the training events as well as their multi-relational positive samples are encoded as multivariate Gaussian density embeddings by the probabilistic event encoding module. After that, the density embeddings are projected into relation-specific subspaces by the relation-aware event projection module. Finally, three modules are jointly optimized by the modified contrastive learning loss. The details of each step are discussed as follows.

3.1 Multi-relational Event Generating

It is critical for contrastive learning to construct positive samples for training data. The common practices to generate positive samples in NLP tasks are token replacement, token shuffling, or token removing (Yan et al., 2021). However, such methods are not suitable for generating positive event samples because events are sensitive to word change. Recently, the dropout mechanism is employed to generate positive samples (Wu et al., 2021), which is still not suitable for our scenario because the positive samples generated by dropouts cannot encode relational prior knowledge.

Therefore, we propose to employ $\mathbb{COMET}$ (Bosselut et al., 2019), a commonsense generative model, to automatically generate multi-relational positive samples for training events. Specifically, $\mathbb{COMET}$ is a transformer-based generative model trained on the commonsense knowledge graph, ATOMIC (Sap et al., 2019) for automatically commonsense knowledge graph construction. Given the head event as $X^s = \{x^s_0, ..., x^s_{|s|}\}$ and relation as $X^r = \{x^r_0, ..., x^r_{|r|}\}$, where xs are word tokens, $\mathbb{COMET}$ generates the tail event as $X^o = \{x^o_0, ..., x^o_{|o|}\}$ by:

$$X^o = \mathbb{COMET}([X^s, X^r]) \quad (1)$$

Given a set of training events $D = \{e_i = (x_1, x_2, ..., x_{|e_i|})\}_{i=1}^n$, each event e_i is fed into $\mathbb{COMET}$ as head event X^s. As for relations, the nine default training relations $\{r_j\}_{j=1}^k$ in $\mathbb{COMET}$ are employed. The details of relations used for training are listed in Appendix A. Then, relational positive samples are generated for each event under each relation and the multi-relational positive event sample set $\{e^i_{r_j} | i = 1, ..., n; j = 1, ..., k\}$ is obtained by:

$$e^i_{r_j} = \mathbb{COMET}([e_i, r_j]) \quad (2)$$

3.2 Probabilistic Event Encoding

Previous event representation learning methods usually adopt specific neural network architectures with static word embeddings as the event encoders (Granroth-Wilding and Clark, 2016; Modi, 2016). Recent work show the effectiveness of the pre-trained language model such as BERT (Devlin et al., 2019) in event encoding (Gao et al., 2022). In this paper, we also employ BERT as the backbone of the event encoder.

To represent the uncertainty and polysemy of events, we propose to learn density event embeddings instead of point event embeddings. In this paper, we choose multivariate Gaussian distribution as the density. The reasons are two-fold. On the one hand, Gaussian distributions only require two parameters and are easy to optimize. On the other hand, Gaussian distribution has an analytical form under many calculations such as Kullback–Leibler (KL) divergence.

In the probabilistic event encoding module, an event $e_i = \{x_1, ..., x_{|e_i|}\}$ (for both the original training data and the generated positive samples) is first fed into the BERT encoder to get their semantic representation by:

$$q_i = \{\text{[CLS]}, x_1, ..., x_{|e_i|}, \text{[SEP]}.\} \tag{3}$$

$$[\boldsymbol{v}^{(i)}_{\text{[CLS]}}, \boldsymbol{v}^{(i)}_{x_1}, ..., \boldsymbol{v}^{(i)}_{x_{|e_i|}}, \boldsymbol{v}^{(i)}_{\text{[SEP]}}] = \text{BERT}(q_i) \tag{4}$$

where $\boldsymbol{v}$s are the vector representations by BERT and q_i is the input query for the event e_i by concatenating the token sequence with the special token [CLS] and [SEP]. The vector representation of [CLS] token $\boldsymbol{v}^{(i)}_{\text{[CLS]}}$ is utilized as the semantic representation for e_i. For simplification, we assume the variance matrixes of density embeddings are diagonal. Therefore the variance matrixes can be fully specified by the variance vectors at the diagonal. Then the semantic representation for e_i is projected to the mean vector and variance vector of the Gaussian embedding by two specific Multilayer Perceptrons (MLPs):

$$\begin{aligned} \boldsymbol{\mu}_i &= \text{MLP}_{mean}(\boldsymbol{v}^{(i)}_{\text{[CLS]}}) \\ \boldsymbol{\sigma}^2_i &= \text{MLP}_{var}(\boldsymbol{v}^{(i)}_{\text{[CLS]}}) \end{aligned} \tag{5}$$

The density representation $\boldsymbol{z}_i$ of the event e_i is:

$$\boldsymbol{z}_i = \mathcal{N}(\boldsymbol{\mu}_i, diag(\boldsymbol{\sigma}^2_i)) \tag{6}$$

where $diag(\boldsymbol{v})$ means matrix taking $\boldsymbol{v}$ as diagonal.

3.3 Relation-aware Event Projecting

The original contrastive learning algorithm learns representations in a common embedding space, which requires the embeddings of positive sample pairs to be close and those of negative pairs to be separate. However, we argue that for multi-relational learning, the original contrastive learning may not be valid. Similar to knowledge graph embedding, an event may have multiple aspects and various relations may focus on different aspects of events, which makes a common space insufficient for modeling. Therefore, inspired by the knowledge graph embedding methods (Wang et al., 2014; Lin et al., 2015), we propose to perform contrastive learning of different relations at different relation-specific hyperplanes to make sure that different relations do not affect each other during learning.

Give a event embedding $\boldsymbol{z}_i$ and a relation r, we project $\boldsymbol{z}_i$ by:

$$f_r(\boldsymbol{z}_i) = \boldsymbol{z}_i - \boldsymbol{\omega}_r^T \boldsymbol{z}_i \boldsymbol{\omega}_r \tag{7}$$

where $\boldsymbol{\omega}_r$ denotes the normal vector for hyperplanes of r. Based on projection $f_r(\cdot)$, the Gaussian event embedding $\boldsymbol{z}_i$ is projected into a subspace with $\boldsymbol{\omega}_r$ as a normal vector. It should be noted that a linear transformation of Gaussian distribution is still Gaussian. Therefore the density embedding after projection is still Gaussian density.

Such transformation requires the normal vector for hyperplanes of relation, which is learned during training and unknown for an unknown relation. However, in the real world, the relations between events are various, which can not be enumerated during the model training. Therefore, it is necessary for our method to be generalized to unknown relations. To deal with this problem, an attention-based mechanism is proposed to learn the relation-specific normal vectors $\boldsymbol{\omega}$ automatically based on the context of event pairs. To be more specific, given an event pair with unknown relation $\{e_i, e_j\}$, we first obtain their context embedding $\boldsymbol{c}$ by concatenating them together and feeding the concatenation into the BERT encoder by:

$$q_{ij} = \{\text{[CLS]}, e_i, e_j, \text{[SEP]}.\} \tag{8}$$

$$[\boldsymbol{v}^{(ij)}_{\text{[CLS]}}, ..., \boldsymbol{v}^{(ij)}_{\text{[SEP]}}] = \text{BERT}(q_{ij}) \tag{9}$$

Again, the representation of token [CLS] is utilized as the event pair context embedding $\boldsymbol{c}_{ij} = \boldsymbol{v}^{(ij)}_{\text{[CLS]}}$. Then the attention mechanism is adopted to learn the context-aware relation normal vector $\boldsymbol{\omega}_a$ based on a set of relation hyperplane normal vectors $\{\boldsymbol{\omega}_j\}_{j=1}^k$ by:

$$a_r^{(ij)} = \frac{exp(\boldsymbol{c}_{ij} \cdot \boldsymbol{\omega}_r)}{\sum_{t=1}^k exp(\boldsymbol{c}_{ij} \cdot \boldsymbol{\omega}_t)} \tag{10}$$

$$\boldsymbol{\omega}_a^{(ij)} = \sum_{r=1}^{k} a_r^{(ij)} \boldsymbol{\omega}_r \tag{11}$$

$\boldsymbol{\omega}_a^{(ij)}$ is normalized by:

$$\boldsymbol{\omega}_a^{(ij)} \leftarrow \frac{\boldsymbol{\omega}_a^{(ij)}}{||\boldsymbol{\omega}_a^{(ij)}||^2} \tag{12}$$

The Gaussian density embeddings of e_i and e_j are projected as:

$$\begin{aligned} f_a(\boldsymbol{z}_i) &= \boldsymbol{\mu}_i - (\boldsymbol{\omega}_a^{(ij)})^T \boldsymbol{\mu}_i \boldsymbol{\omega}_a^{(ij)} \\ f_a(\boldsymbol{z}_j) &= \boldsymbol{\mu}_j - (\boldsymbol{\omega}_a^{(ij)})^T \boldsymbol{\mu}_j \boldsymbol{\omega}_a^{(ij)} \end{aligned} \tag{13}$$

3.4 Multi-relational Probabilistic Contrastive Learning

As stated before, contrastive learning is employed. However, the original InfoNCE loss (Chen et al., 2020) is designed for single-relational (similar or dissimilar) deterministic embedding, which is not suitable for multi-relational probabilistic embedding.

Therefore, we modify the original InfoNCE loss. One important component of the InfoNCE loss is the distance function, where the cosine similarity is usually employed. For Gaussian density embeddings, we utilize the symmetric KL divergence to serve as the distance function. The distance function $g(\cdot, \cdot)$ is:

$$g(\boldsymbol{a}, \boldsymbol{b}) = \exp\{\frac{1}{2\tau}(\mathrm{KL}(\boldsymbol{a}||\boldsymbol{b}) + \mathrm{KL}(\boldsymbol{b}||\boldsymbol{a}))\} \tag{14}$$

where $\boldsymbol{a}$ and $\boldsymbol{b}$ are two density embeddings, τ is the temperature parameter.

Then we set the multi-relation part of the loss function as:

$$\mathcal{L}_{mr} = -\sum_{i=1}^{n}\sum_{r=1}^{k} \log \frac{\epsilon_r \cdot g(f_a(\boldsymbol{z}_i), f_a(\boldsymbol{z}_r^{(i)}))}{g(f_a(\boldsymbol{z}_i), f_a(\boldsymbol{z}_r^{(i)})) + \sum_{j \in N(i)} g(f_r(\boldsymbol{z}_i), f_r(\boldsymbol{z}_j))} \tag{15}$$

where $\boldsymbol{z}_r^{(i)}$ is the density embedding of the positive sample under the relation r for the event e_i, $\boldsymbol{z}_i$ is the density embedding of e_i, $N(i)$ is the index set of in-batch negative sample of e_i and ϵ_r is the weight parameter for the relation r. Note that for the calculation of distance for negative samples, we use vanilla relational projection $f_r(\cdot)$ instead of attention-based relational projection $f_a(\cdot)$ to keep the negative pairs separate for every relation.

To capture the event semantics, we also introduce the dropout-based positive samples during contrastive training:

$$\mathcal{L}_{dp} = -\sum_{i=1}^{n}\sum_{r=1}^{k} \log \frac{\epsilon_r \cdot g(f_r(\boldsymbol{z}_i), f_r(\boldsymbol{z}_i^{(+)}))}{g(f_r(\boldsymbol{z}_i), f_r(\boldsymbol{z}_i^{(+)})) + \sum_{j \in N(i)} g(f_r(\boldsymbol{z}_i), f_r(\boldsymbol{z}_j))} \tag{16}$$

where $\boldsymbol{z}_i^{(+)}$ is the density embedding of the dropout-based positive sample for the event e_i. It should be noted that vanilla relational projection $f_r(\cdot)$ is used for the calculation of positive samples in this part as we want the dropout-based positive samples to be close to the training samples for every relation.

As discussed in Gao et al. (2022), introducing the original Mask Language Modeling (MLM) loss $\mathcal{L}_{mlm}$ into learning is beneficial for the backbone encoder. The final loss function is obtained by adding the above three terms together:

$$\mathcal{L}_{\text{m-InfoNCE}} = \beta \mathcal{L}_{mr} + \mathcal{L}_{dp} + \mathcal{L}_{mlm} \tag{17}$$

where β is the loss weight parameter.

4 Experiments

In this section, we investigate the effectiveness of MORE-CL by comparing it with several competitive baselines both on the conventional event similarity task and the proposed multi-relation probabilistic event similarly task.

4.1 Implementation Details

Following previous works (Weber et al., 2018; Ding et al., 2019; Gao et al., 2022), the training events are extracted from the *New York Times Gigaword Corpus* using Open Information Extraction system Ollie (Mausam et al., 2012). Specifically, we use the same filter setting as (Gao et al., 2022), which results in 4,029,877 distinct events. For each event, we use $\mathbb{COMET}$ to generate its positive samples under the 9 default relations. The details of generation are shown in Appendix A.

The backbone used in the encoder module is *BERT-base-uncased*. The learning rate is set as 4e-7. The model is trained with a batch size of 125 and total epochs of 2 by an Adam optimizer. The optimal dimension of Gaussian density embedding is chosen by experiments and set to 500. The loss weight parameter β is set to 0.01. The temperature parameter τ is set as 0.3 and the weight parameters for relations ϵ are set to 0.1. In practice, we assume the output of network $\text{MLP}_{var}(\cdot)$ is the log of variance vector which is taken exponential when used to keep it non-negative. At each batch, the calculated KL divergence values are normalized by min-max normalization to make the training process more stable. The model is implemented by PyTorch (Paszke et al., 2019).

4.2 Datasets

Hard Similarity Dataset. Weber et al. (2018) proposed a dataset of 115 samples to identify the semantically similar event pairs from the dissimilar pairs. To make the dataset more difficult, the positive samples are annotated to have little lexical overlaps with the anchor events while the negative samples are annotated to have high overlaps. Ding et al. (2019) extended this dataset to 1000 samples. For both datasets, accuracy is adopted as an evaluation metric, where a sample is successfully processed if and only if the similarity between the positive pair is higher than the similarity between the negative pair.

Transitive Sentence Similarity. Kartsaklis and Sadrzadeh (2014) proposed this fine-grained similarity dataset, which contains 108 pairs of transitive sentences that consist of a subject, a verb, and an object. Each pair of events is assigned with similarity scores from 1 to 7 by human annotators, where a higher value indicates more similar events. For this dataset, Spearman's correlation between the similarity score is predicted by each method and the average annotated similarity score is employed as the evaluation metric.

Multi-Relational Probabilistic Event Similarity (MRPES). The previous two datasets are designed to evaluate the single-relational deterministic event representations. To further investigate whether the knowledge of multiple relations between events and uncertainty within events is learned, we propose a new multi-relational probabilistic event similarity dataset (MRPES). MRPES is an extension of Weber's dataset, containing 115 samples. As shown in Table 1, each sample in MRPES contains 1 anchor event, 1 negative sample, 4 relational positive samples, and 2 probabilistic positive samples. The anchor events and negative samples are taken from (Weber et al., 2018), while the rest of the events are manually annotated. For relational positive samples, we choose two learned relations **oEffect** and **xNeed** and two unknown relations **contrast** and **sequential**. For probabilistic positive samples, we annotate each anchor event with two semantically-related events while these two events are semantically different. For the relational test, the setting is the same as the original **Hard Similarity Dataset**. For the probabilistic test, a sample is successfully processed if and only if the similarities between two positive samples are both greater than its similarity with the negative sample. The details of the dataset are listed in Appendix B.

Anchor	**Negative**
journalist capture animal	journalist capture image
oEffect	**xNeed**
animal is caught	person be a hunter
Contrast	**Sequential**
animal escaped	person sell animal
Probabilistic_1	**Probabilistic_2**
man hunt deer	kid catch insect

Table 1: An example in the MRPES dataset.

4.3 Baselines

Following (Gao et al., 2022), three types of methods are employed for comparison:

- **Event representation methods**: **EM Comp.**, **Role Factor Tensor** and **Predicate Tensor** are all proposed by (Weber et al., 2018) to learn the interactions of event components with tensor networks. **SWCC** (Gao et al., 2022) is the current SOTA method for the event similarity task by incorporating contrastive learning and prototypical clustering simultaneously.

- **Event representation methods with external knowledge**: **KGEB** (Ding et al., 2016) incorporates knowledge graph information. **FEEL** (Lee and Goldwasser, 2018) employs animacy and sentiment as extra features of events. **NTN-IntSent** (Ding et al., 2019) utilizes sentiment and intent of events to enhance event representations.

- **Multi-relational script learning methods**: **SAM-Net** (Lee and Goldwasser, 2019) incorporates discourse relations into script learning. **UniFA-S** (Zheng et al., 2020) utilizes scenario knowledge for event representations.

4.4 Main Results

Similarity datasets results. The experimental results for three similarity datasets are shown in Ta-

Method	Hard similarity%		Transitive sentence
	Original	Extended	similarity (ρ)
EM Comp. (Weber et al., 2018)	33.9	18.7	0.57
Predicate Tensor (Weber et al., 2018)	41.0	25.6	0.63
Role Factor Tensor (Weber et al., 2018)	43.5	20.7	0.64
SWCC (Gao et al., 2022)	80.9	72.1	**0.82**
KGEB (Ding et al., 2016)	52.6	49.8	0.61
FEEL (Lee and Goldwasser, 2018)	58.7	50.7	0.67
NTN-IntSent (Ding et al., 2019)	77.4	62.8	0.74
SAM-Net (Lee and Goldwasser, 2019)	51.3	45.2	0.59
UniFA-S (Zheng et al., 2020)	78.3	64.1	0.75
MORE-CL	**89.6**	**84.9**	0.81

Table 2: Results on the similarity datasets. The best results are bold. Part of results are taken from (Gao et al., 2022).

Method	known relation		unknown relation		probablistic test	
	oEffect	xNeed	Contrast	Sequential	one	both
SWCC	42.6	60.0	66.1	56.5	65.2	42.6
MORE-CL	**88.7**	**85.2**	**81.7**	**77.4**	**81.7**	**67.0**

Table 3: Results on the MRPES dataset. Best results are bold. "one" for the probabilistic test denotes at least one positive sample is successfully predicted, while "both" denotes both positive samples are successfully predicted.

ble 2. It can be observed that MORE-CL outperforms all the other baselines on two hard similarity datasets by a large margin except that on the transitive sentence similarity dataset, MORE-CL achieves similar results as SWCC. It might be attributed to the under-calibration of Gaussian density embeddings, which is also found in (Zhang et al., 2021). It can also be observed that methods with external knowledge or multi-relation knowledge generally outperform those without using external knowledge.

MRPES results. As SWCC generally outperformed other methods by a large margin in the similarity task, we only compare our method with SWCC on the MRPES dataset. As shown in Table 3, MORE-CL outperforms SWCC greatly. The reason is obvious that SWCC is a single-relational deterministic event representation method without modeling multiple relations and uncertainty of events. Moreover, the high accuracy scores for two known relations show that MORE-CL learns the training relations well. It can also be found that scores for unknown relations are slightly lower than known relations', showing that MORE-CL can generalize to unknown relations.

As for the probabilistic test, the result is interesting. For the case of at least one sample correct, the performance gap between SWCC and MORE-CL is 16.5% while for the case of both samples correct, the performance gap increases to 24.4%. It further verifies our assumption that it is hard for point embeddings to model that one embedding is close to the other two embeddings which should be separated.

4.5 Model Analysis

In this part, we remove or change three components of MORE-CL and generate four experiment settings to investigate their effects on the performance, where setting **S5** is the original model.

- **S1** is the setting where the probabilistic event encoding module is replaced with a normal BERT encoder, and the symmetric KL similarity is replaced with cosine similarity.

- **S2** is a model where the multi-relational event generation module is removed, and the dropout-based positive samples are utilized for contrastive learning.

- **S3** is the setting where the relation-aware event projection module is removed. For inference, the test samples are processed under all 9 training relations respectively and the final decision is made by averaging the results for 9 training relations.

Settings	Hard similarity%		Transitive sentence
	Original	Extended	similarity (ρ)
S1	79.1	65.9	**0.82**
S2	87.0	82.6	0.78
S3	88.7	83.2	0.80
S4	87.8	83.5	0.80
S5	**89.6**	**84.9**	0.81

Table 4: Ablation study on similarity dataset.

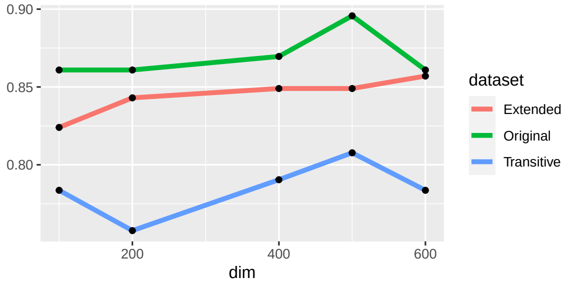


Figure 3: Results of MORE-CL with different embedding dimensions on three similarity datasets.

- **S4** is the setting where a simple version without attention replaces the relation-aware event projection module. For training, an extra relation-specific normal vector is introduced for projecting all event pairs with unknown relations, for inference, the normal vector for unknown relations is employed for projection. In practice, we utilize the co-occurrence data to learn this normal vector.

It can be observed from Table 4 that without the probabilistic encoding module, the performances of MORE-CL on hard similarity datasets drop dramatically while the performance on the transitive sentence similarity dataset increases slightly. The performance fluctuation over hard similarity datasets comes from two parts. On the one hand, the Gaussian density embeddings model the uncertainty within events. On the other hand, the relational positive event samples are generated automatically, which will certainly introduce noise into the model. The performance fluctuation over the transitive sentence similarity dataset further shows that the Gaussian density embeddings are under-calibrated.

To investigate the optimal dimension for MORE-CL, we perform experiments with different embedding dimensions on three similarity datasets. As shown in Figure 3, the performances of MORE-CL increase first and then decrease with the embedding dimension growth, which is concordant with the majority of event representation learning methods.

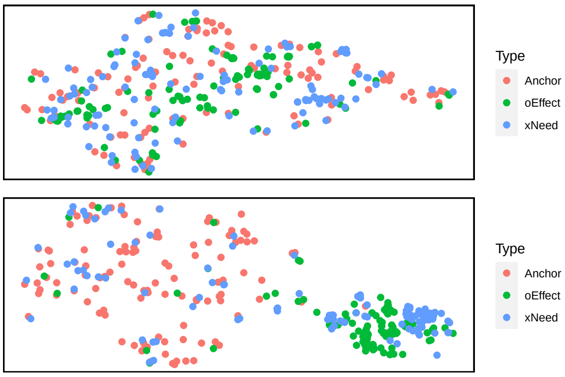


Figure 4: Visualization of the embeddings learned by SWCC (upper part) and MORE-CL (lower part) on the MRPES dataset. For MORE-CL, only the mean vectors are used for visualization.

It should be noticed that the optimal embedding dimension of MORE-CL is smaller than SWCC. The reason might be that the density embeddings can carry more information compared with point embeddings at the same embedding dimension.

4.6 Visualization

To get a more intuitional understanding of multi-relational embedding learning, we present a visualization of embeddings learned by SWCC and MORE-CL with T-SNE (Van der Maaten and Hinton, 2008). As shown in Figure 4, the embeddings learned by SWCC for three types of events are mixed up, while the embeddings learned by MORE-CL are separated.

5 Conclusion

In this paper, to model the multiple relations between events and uncertainty within events, we propose a multi-relational probabilistic event representation learning method, MORE-CL, based on the projected Gaussian embedding with contrastive learning. To be more specific, MORE-CL consists of three modules, a multi-relational event generation module to incorporate relational knowledge of events, a probabilistic event encoding module to model uncertainty with Gaussian density embeddings, and a relation-aware projection module to adapt to unseen relations. What's more, we also present a new dataset to test the knowledge of multiple relations and uncertainty learned by event representation methods. The experimental results for both existing and new datasets show the effectiveness of the proposed method.

Limitations

Though achieving promising results in the experiments, our work still has the following limitations.

- As shown in Table 2 and Table 3. The proposed Gaussian embedding may have a calibration problem leading to performing badly on fine-grained similarity tasks measured by Spearman's correlation.

- The proposed method assumes that all relations are symmetric and adopts a symmetric similarity measurement. However, not all the relations are symmetric. And the ability to deal with unsymmetric relations with unsymmetric measurement is one important advantage of density embeddings which point embeddings do not have.

- The proposed MRPES dataset should be improved in terms of quantity and quality. The number of test samples should be increased to over a thousand to get more statistically robust results. The types of unseen relations should be also increased to have a more comprehensive investigation of the ability to generalize on relations. The negative samples should be elaborately designed to provide the anchor event with different negative samples under different relations.

Acknowledgement

The authors would like to thank the anonymous reviewers for their insightful comments. This work is funded by the National Natural Science Foundation of China (62176053). This work is supported by the Big Data Computing Center of Southeast University.

References

Ben Athiwaratkun and Andrew Gordon Wilson. 2018. Hierarchical density order embeddings. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. 2019. COMET: Commonsense transformers for automatic knowledge graph construction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4762–4779, Florence, Italy. Association for Computational Linguistics.

Hong Chen, Raphael Shu, Hiroya Takamura, and Hideki Nakayama. 2021. GraphPlan: Story generation by planning with event graph. In *Proceedings of the 14th International Conference on Natural Language Generation*, pages 377–386, Aberdeen, Scotland, UK. Association for Computational Linguistics.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR.

Shumin Deng, Ningyu Zhang, Luoqiu Li, Chen Hui, Tou Huaixiao, Mosha Chen, Fei Huang, and Huajun Chen. 2021. OntoED: Low-resource event detection with ontology embedding. 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 2828–2839, Online. Association for Computational Linguistics.

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, Minneapolis, Minnesota. Association for Computational Linguistics.

Xiao Ding, Kuo Liao, Ting Liu, Zhongyang Li, and Junwen Duan. 2019. Event representation learning enhanced with external commonsense knowledge. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4894–4903, Hong Kong, China. Association for Computational Linguistics.

Xiao Ding, Yue Zhang, Ting Liu, and Junwen Duan. 2015. Deep learning for event-driven stock prediction. In *Twenty-fourth international joint conference on artificial intelligence*.

Xiao Ding, Yue Zhang, Ting Liu, and Junwen Duan. 2016. Knowledge-driven event embedding for stock prediction. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2133–2142, Osaka, Japan. The COLING 2016 Organizing Committee.

Jun Gao, Wei Wang, Changlong Yu, Huan Zhao, Wilfred Ng, and Ruifeng Xu. 2022. Improving event representation via simultaneous weakly supervised contrastive learning and clustering. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3036–3049, Dublin, Ireland. Association for Computational Linguistics.

Mark Granroth-Wilding and Stephen Clark. 2016. What happens next? event prediction using a compositional neural network model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30.

Dimitri Kartsaklis and Mehrnoosh Sadrzadeh. 2014. A study of entanglement in a categorical framework of natural language. *arXiv preprint arXiv:1405.2874*.

I-Ta Lee and Dan Goldwasser. 2018. Feel: Featured event embedding learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

I-Ta Lee and Dan Goldwasser. 2019. Multi-relational script learning for discourse relations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4214–4226, Florence, Italy. Association for Computational Linguistics.

Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015. Learning entity and relation embeddings for knowledge graph completion. In *Twenty-ninth AAAI conference on artificial intelligence*.

Mausam, Michael Schmitz, Stephen Soderland, Robert Bart, and Oren Etzioni. 2012. Open language learning for information extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 523–534, Jeju Island, Korea. Association for Computational Linguistics.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Ashutosh Modi. 2016. Event embeddings for semantic script modeling. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, pages 75–83, Berlin, Germany. Association for Computational Linguistics.

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

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Karl Pichotta and Raymond Mooney. 2016a. Learning statistical scripts with lstm recurrent neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30.

Karl Pichotta and Raymond J. Mooney. 2016b. Using sentence-level LSTM language models for script inference. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 279–289, Berlin, Germany. Association for Computational Linguistics.

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 *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 3027–3035.

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(11).

Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014. Knowledge graph embedding by translating on hyperplanes. In *Proceedings of the AAAI conference on artificial intelligence*, volume 28.

Noah Weber, Niranjan Balasubramanian, and Nathanael Chambers. 2018. Event representations with tensor-based compositions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Lijun Wu, Juntao Li, Yue Wang, Qi Meng, Tao Qin, Wei Chen, Min Zhang, Tie-Yan Liu, et al. 2021. R-drop: Regularized dropout for neural networks. *Advances in Neural Information Processing Systems*, 34:10890–10905.

Yuanmeng Yan, Rumei Li, Sirui Wang, Fuzheng Zhang, Wei Wu, and Weiran Xu. 2021. ConSERT: A contrastive framework for self-supervised sentence representation transfer. 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 5065–5075, Online. Association for Computational Linguistics.

Linhai Zhang, Deyu Zhou, Yulan He, and Zeng Yang. 2021. Merl: Multimodal event representation learning in heterogeneous embedding spaces. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 14420–14427.

Jianming Zheng, Fei Cai, and 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*, pages 249–258.

Deyu Zhou, Jianan Wang, Linhai Zhang, and Yulan He. 2021. Implicit sentiment analysis with event-centered text representation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6884–6893, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Relation	Explanation	Example
xIntent	Why does X cause the event	PersonX wanted to be nice
xNeed	What does X need to do before the event?	PersonX knows PersonY well
xAttr	How would X be described?	PersonX is caring
xEffect	What effects does the event have on X?	PersonX will want to chat with PersonY
xReact	How does X feel after the event?	PersonX will feel good
xWant	What would X likely want to do after the event?	PersonX will want to chat with PersonY
oEffect	What effects does the event have on others?	PersonY will smile
oReact	How do others feel after the event?	PersonY will feel flattered
oWant	What would others likely want to do after the event?	PersonY will compliment PersonX back

Table 5: Relations explained in Multi-relational event generation. Explanation and examples are taken from (Sap et al., 2019). The head entity is (*PersonX pays PersonY a compliment*). X denotes the subject of the head entity, and o denotes the subject of the tail entity.

Type		Explanation	Example
Anchor		Event that will be tested.	journalist capture animal
Negative		Event that is textually similar and semantically dissimilar to the anchor event.	journalist capture image
Seen relations	oEffect	What effects does the anchor event have on others?	animal is caught
	xNeed	What does X need to do before the anchor event?	person be a hunter
Unseen relations	Contrast	What is the opposite of the anchor event?	animal escaped
	Sequential	What is most likely to happen after the anchor event?	person sell animal
Prbabilistic test	Probabilistic_1	An event that is semantically similar to the anchor event.	man hunt deer
	Probabilistic_2	Another event that is semantically similar to the anchor event.	kid catch insect

Table 6: Relations in the MERPES dataset.

A Relations explained in Multi-relational event generation

COMET (Bosselut et al., 2019) is a transformer-based generative model trained on the commonsense knowledge graph, ATOMIC (Sap et al., 2019), which employs 9 types of relations of ATOMIC. ATOMIC constructs event triplets by asking *If-then* questions. For example, for a commonsense "*if* X pays Y a compliment, *then* then Y will likely return the compliment", the relation between these two events is "what would others likely want to do after the event?", which is denoted as *oWant* in ATOMIC, then this commonsense will be transformed as an event triplet {(*PersonX pays PersonY a compliment*), *oWant*, (*Y will compliment PersonX back*}. To capture more knowledge of relations between events, we also employ all 9 types of relations to generate positive samples. The details of the relations used are shown in Table 5.

B Relations explained in MERPES dataset

To further investigate the knowledge of multiple relations and uncertainty learned by the event representation learning methods, we propose a multi-relational probabilistic event similarity dataset (MRPES). Every sample in MRPES data consists of 8 events. The details of each event and its explanation is shown in Table 6.

ACL 2023 Responsible NLP Checklist

A For every submission:

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

☒ A2. Did you discuss any potential risks of your work?
Our work follows previous work with the same setting, therefore we believe no risks will be included in our work.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
the section abstract and the section 1 Introduction.

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

B ☑ Did you use or create scientific artifacts?

the section 4 Experiments.

☑ B1. Did you cite the creators of artifacts you used?
the section 3 Method and the section 4 Experiments.

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
the section 3 Method and the section 4 Experiments.

☑ 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)?
the section 4 Experiments.

☑ 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?
the section 4 Experiments.

☒ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
The dataset used for training and testing are regular English texts collected by pervious work.

☑ 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.
the section 4 Experiments.

C ☑ Did you run computational experiments?

the section 4 Experiments.

☒ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
The size of our proposed model is relatively small, therefore the total computational budget should be affordable for most of people in the community.

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?
the section 4 Experiments.

☑ 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?
the section 4 Experiments.

☑ 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.)?
the section 4 Experiments.

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.