MERL: Multimodal Event Representation Learning in Heterogeneous Embedding Spaces

Linhai Zhang[1] , **Deyu Zhou**[1*] , **Yulan He**[2] , **Zeng Yang**[1]

[1] School of Computer Science and Engineering, Key Laboratory of Computer Network and Information Integration, Ministry of Education, Southeast University, China

[2] Department of Computer Science, University of Warwick, UK

{lzhang472, d.zhou, yangzeng}@seu.edu.cn, yulan.he@warwick.ac.uk

Abstract

Previous work has shown the effectiveness of using event representations for tasks such as script event prediction and stock market prediction. It is however still challenging to learn the subtle semantic differences between events based solely on textual descriptions of events often represented as (subject, predicate, object) triples. As an alternative, images offer a more intuitive way of understanding event semantics. We observe that event described in text and in images show different abstraction levels and therefore should be projected onto heterogeneous embedding spaces, as opposed to what have been done in previous approaches which project signals from different modalities onto a homogeneous space. In this paper, we propose a Multimodal Event Representation Learning framework (MERL) to learn event representations based on both text and image modalities simultaneously. Event textual triples are projected as Gaussian density embeddings by a dual-path Gaussian triple encoder, while event images are projected as point embeddings by a visual event component-aware image encoder. Moreover, a novel score function motivated by statistical hypothesis testing is introduced to coordinate two embedding spaces. Experiments are conducted on various multimodal event-related tasks and results show that MERL outperforms a number of unimodal and multimodal baselines, demonstrating the effectiveness of the proposed framework.

Introduction

In Natural Language Processing (NLP), it is important to construct an event structure which represents what is going on in text since it is crucial for language understanding and reasoning. Transferring events into machine-readable form is important for many NLP tasks such as question answering, discourse understanding, and information extraction. By now, the mainstream practice is to represent prototype events as low-dimensional dense vectors.

Notable progresses have been made in learning event representations or embeddings from structured event triples expressed as *(subject, predicate, object)*. Event embeddings are typically learned in a compositional manner from their constituent embeddings. Two types of commonly-used composition methods are additive-based and tensor-based. For additive-based methods, the concatenation or addition of event components (i.e. the *subject*, *predicate* and *object*) embeddings are projected onto an event embedding space with a parameterized function using a neural network (Ashutosh and Ivan 2014; Granroth-Wilding and Clark 2016; Modi 2016). For tensor-based methods, the event components are composed by the tensor operation, where the multiplicative interactions between event components could be captured (Ding et al. 2015; Weber et al. 2018; Ding et al. 2019).

Figure 1: Upper part: Event (i) and (iii) share the same *subject* and *predicate* and yet bear different semantic meanings. On the contrary, Event (i) and (ii), despite having no event element in common, are semantically more similar. Event semantics could be distinguished more easily from event-associated images. Lower part: an event triple can be associated with multiple images.

Despite the success of event representation learning approaches, it is still challenging for models to learn the subtle semantic differences between events based solely on the text modality. For example, as shown in the upper part of Figure 1, there are three events: (i) *(she, attend, baby)*, (ii) *(he, take care, kid)* and (iii) *(she, attend, meeting)*. Both Event (i) and (ii) share the same subject and predicate, however expressing totally different meanings. On the contrary, Event (i) and (iii) do not have any event elements in common, yet they describe the same event. In such situations, event semantics can be better captured by event-associated images rather than the text description of event triples as shown in Figure

* Corresponding author.

1. It is thus crucial to utilize the image modality to enhance event representation learning.

Multimodal representation learning method aims to learn a unified representation of semantic units or information conveyed in different modalities (e.g., text and image). Previous approaches often project information from different modalities onto a homogeneous embedding space such as a point embedding space or a density embedding space (Vendrov et al. 2016; Ben and Andrew 2018). However, in event embedding learning, events depicted in images often convey much more information than their counterpart text descriptions. As illustrated in the lower part of Figure 1, the event *(he, play, soccer)* can be depicted by more than one image. In this example, we can view the event triple, expressed in a more concise way, as an abstraction of its associated event images; while each event image is an instantiation of the event triple, which may contain some details beyond what have been expressed in text.

In this paper, we propose a multimodal event representation learning framework (MERL) to project event triples and their associated images onto heterogeneous embedding spaces. More concretely, for each event, its event triple is projected onto a Gaussian density embedding space with a dual-path Gaussian triple encoder. The mean and variance of the Gaussian distribution is estimated using different composition methods. For an event image, a visual event components-aware image encoder is proposed to extract visual event components and generate an image point embedding. As we assume that an event image is an instantiation of an event triple, we want to ensure that the image point embeddings behave as if they were sampled from the event triple associated Gaussian embedding. To this end, a novel score function motivated by statistical hypothesis test is proposed, which is theoretically guaranteed and flexible to extend. Various experiments including multimodal event similarity, script event prediction and multimodal event retrieval have been conducted to evaluate the effectiveness of the proposed method.

The main contributions of the paper are listed as follows:

- A novel multimodal representation learning framework to project event triples and images onto heterogeneous embedding spaces is proposed, along with a statistically-motivated score function to coordinate event triple embeddings and images embeddings residing in the heterogeneous embedding spaces.
- Two novel encoders are designed for modeling event information from different modalities. The event triple encoder composes event components to estimate mean and variance of Gaussian embeddings by two different encoding paths. The image encoder extracts visual event features to generate image point embeddings.
- Experimental results on various tasks demonstrate that the proposed framework outperforms a number of competitive event representation learning and multimodal representation learning methods, verifying the effectiveness of multimodal learning of event representations.

Related Work

This paper is related to the following two lines of research:

Event Representation Learning Event representation learning approaches project prototype events represented as triples or sentence into dense vectors using neural networks, where similar events are embedded close to each other while distinct events are separated. Ding et al. (2015) proposed a tensor-based event embedding model for stock market prediction where components of structured events are composed by 3-dimensional tensors. Granroth-Wilding and Clark (2016) and Modi (2016) concatenated the embeddings of subject, predicate and object and fed them into a neural network to generate event embeddings. Ding et al. (2016) proposed to incorporate a knowledge graph into a tensor-based event embedding model. Pichotta and Mooney (2016) frame event prediction as a sequence to sequence problem where the components of structured events are fed into an LSTM model sequentially to predict the components of next event. Weber, Balasubramanian, and Chambers (2018) proposed another tensor-based event representation model, which learns to generate tensors based on embeddings of predicates. Lee and Goldwasser (2018) introduced sentiment polarity and animacy of events as additional event components and learn event embeddings. Ding et al. (2019) proposed a multi-task learning framework to inject sentiment and intent of events into the event embedding. However, all aforesaid methods only considered a single text modality and did not take into account other modalities such as images.

Multimodal Representation Learning Multimodal representation learning aims to learn representations of objects from multiple information sources, such as text, image and audio. Multimodal representation learning methods can be categorized into two types, joint representation and coordinated representation. Joint representation methods map unimodal signals from different modalities onto the same representation space. Silberer and Lapata (2014) proposed to concatenate text embeddings and image embeddings to predict object labels with stacked auto-encoders. Rajagopalan et al. (2016) proposed to explicitly model the view-specific and cross-view interactions over time for structured outputs. Coordinated representation methods learn separate representations for each modality but coordinate them through a constraint. Frome et al. (2013) proposed to constrain textual embeddings and visual images with a similarity inner product. Vendrov et al. (2016) proposed an order-preserving embedding framework to learn the partial order relationships between texts and images. Ben and Andrew (2018) extended this framework by replacing point embeddings with density embeddings. Li et al. (2020) introduced the multimedia event extraction task, which aims to extract events and their arguments from multimedia documents. However, all aforesaid methods represent images and texts in homogeneous spaces. To the best of our knowledge, this work represents the first attempt that considers multimodal representation learning in heterogeneous embedding spaces.

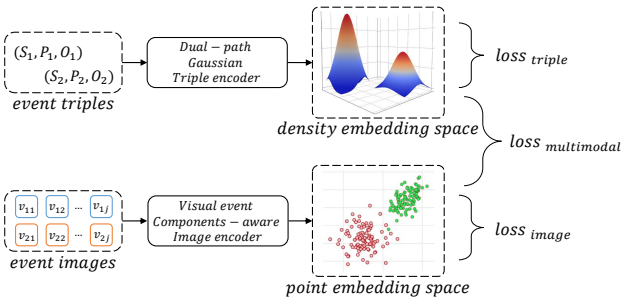

Figure 2: The architecture of Multimodal Event Representation Learning framework (MERL) with heterogeneous embedding spaces.

Methodologies

In this section, we will first introduce the overall architecture of the proposed Multimodal Event Representation Learning framework (MERL), followed by the details of the triple encoder and the image encoder. Finally, we describe the training procedure of the whole framework.

Multimodal Event Representation Learning on Heterogeneous Embedding Spaces

As shown in Figure 2, the proposed framework can be categorized as a coordinated multimodal representation method. In MERL, event triples and event images are projected onto heterogeneous embedding spaces by the carefully-designed triple encoder and image encoder with two intra-modal learning objectives. To fuse knowledge from multiple modalities, a cross-modal constraint is proposed to align the triple embedding space and image embedding space.

Problem Setting Firstly, we formulate the problem of multimodal event representation learning below. We assume each event e is paired with one event triple $t = \{S, P, O\}$ and one or more image descriptions $\{v_j|j = 1, ..., k\}$. MERL aims to achieve the following:

- Semantically similar event triples are projected into nearby locations in the embedding space;
- Images of the same event are projected and clustered together in the embedding space, while semantic relations of their corresponding event triples are preserved;
- For the same event, event image point embeddings are distributed as if they were sampled from the event triple density embedding.

Heterogeneous Embedding Spaces As stated before, we assume that an event triple is an abstraction of its associated images while event images are instantiations of the event triple. As such, MERL projects event data from different modalities onto heterogeneous embedding spaces. An event triple t is projected onto a density embedding space $\mathbb{D}$:

$$t \xrightarrow[\text{encoder}]{\text{triple}} \boldsymbol{t} \in \mathbb{D} \tag{1}$$

That is, each event triple is associated with a density. Density embedding learning is therefore equivalent to the estimation of density parameters. An event images v, on the contrary, is projected onto a point embedding space $\mathbb{P}$:

$$v \xrightarrow[\text{encoder}]{\text{image}} \boldsymbol{v} \in \mathbb{P} \tag{2}$$

Coordinate the Heterogeneous Spaces Coordinated multimodal representation methods usually employ homogeneous measurements such as cosine similarity or KL divergence to align embeddings in homogeneous embedding spaces. Such measurements are, however, not applicable here since we are dealing with heterogeneous embedding spaces. To measure between distributions and points, a natural way is to employ hypothesis testing. In statistics, a hypothesis is an assumption about the population parameter and a hypothesis testing is to assess the plausibility of the hypothesis with sample data. A common choice of hypothesis testing is the likelihood-ratio test (Casella and Berger 2002). For the null hypothesis H_0 and an alternative hypothesis H_1:

$$H_0 : \theta \in \Theta_0 \quad v.s. \quad H_1 : \theta \in \Theta_0^C \tag{3}$$

the test statistics of likelihood ratio test is:

$$\lambda(\boldsymbol{x}) = \frac{sup_{\Theta_0} L(\theta|\boldsymbol{x})}{sup_{\Theta} L(\theta|\boldsymbol{x})} \tag{4}$$

where $\boldsymbol{x}$ denotes samples, θ denotes parameters, $L(\cdot|\cdot)$ denotes the likelihood, Θ_0 denotes the parameter space of the null hypothesis, and Θ denotes the full parameter space. When the null hypothesis holds, the difference between the maximum likelihood over Θ_0 and the maximum likelihood over Θ should not be greater than the random sampling error. That is, the null hypothesis will be rejected when $\lambda(\boldsymbol{x}) < c$ ($c \in (0, 1)$). In other words, the larger the test statistics $\lambda(\boldsymbol{x})$ is, the more likely the null hypothesis H_0 is true.

Motivated by the likelihood ratio test, we propose a score function to measure between our density embedding $\boldsymbol{t}$ and multiple point embeddings $\boldsymbol{v}_s$. Taking log of both sides of Equation (4), we have:

$$s(\boldsymbol{t}, \boldsymbol{v}_s) = log(sup_{\Theta_0} L(\boldsymbol{t}|\boldsymbol{v}_s)) - log(sup_{\Theta} L(\boldsymbol{t}|\boldsymbol{v}_s)) \tag{5}$$

The larger value of Equation (5) is, the more likely the hypothesis between distribution $\boldsymbol{t}$ and data point $\boldsymbol{v}_s$ to be true.

In MERL, we assume that event images are instantiations of the event triple, which could be expressed as:

$$H_0 : \boldsymbol{v}_{j|j=1,...,k} \sim \mathcal{N}(\mu_t, \sigma_t^2) \tag{6}$$

Considering that the optimal parameters of $L(\theta|\boldsymbol{x})$ over Θ could be estimated by maximum likelihood estimation (MLE), which is also statistics of $T(\boldsymbol{x})$ of samples $\boldsymbol{x}$, for MERL, Equation (5) can be rewritten as:

$$s(\boldsymbol{t}, \boldsymbol{v}_s) = log(\mathcal{N}(\boldsymbol{v}_s|\mu_t, \sigma_t^2)) - log(\mathcal{N}(\boldsymbol{v}_s|T(\boldsymbol{v}_s)) \tag{7}$$

where $\mathcal{N}(x|\mu, \sigma^2)$ is likelihood of Gaussian distribution, $T(\boldsymbol{x}) = \{\bar{\boldsymbol{x}}, S^2(\boldsymbol{x})\}$ are the average and standard deviation of sample $\boldsymbol{x}$.

The score function can be interpreted as follows: the first term should be maximized, which is intuitive because this log likelihood measures the goodness of fit between a distribution and samples; the second term should be minimized,

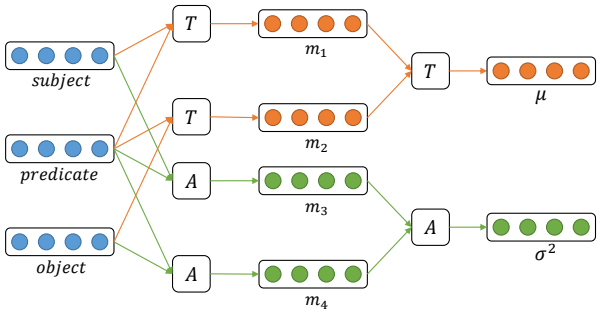

Figure 3: Dual-path Gaussian Triple Encoder.

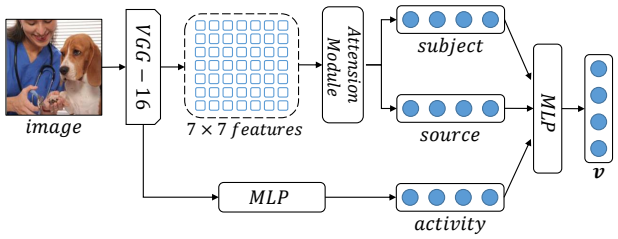

Figure 4: Visual Event Components-aware Image Encoder.

which can be interpreted as a penalty term that prevents samples to be clustered and encourages them to be dispersed and therefore more representative.

With the embedding spaces and the measurement between spaces defined, we can derive the intra-modal loss to learn the geometry within each modality and the cross-modal loss to coordinate between the modalities.

Dual-path Gaussian Triple Encoder

The goal of the triple encoder is to map an event triple $t = (S, P, O)$ to a Gaussian embedding $\boldsymbol{t}$, where S refers to subject or actor, P refers to predicate or action and O is object of the action. In this paper, to simplify the model, we set the co-variance matrix of a Gaussian distribution diagonal. The task is to compose the event components to calculate the mean vector and variance vector $\boldsymbol{t} = (\mu, \sigma^2)$ of an event density embedding.

Previous methods on neural Gaussian embeddings often calculate mean and variance vector with a shared encoder (Kendall and Gal 2017; Oh et al. 2018; Xiao and Wang 2019). However, we argue that the mean vector and the variance vector describe different aspects of the corresponding Gaussian distribution, that the mean vector determines the location of the Gaussian distribution in embedding space, while the variance vector captures the shape and spread of the Gaussian distribution. As such, we proposed a dual-path Gaussian event triple encoder, shown in Figure 3, which predicts the mean vector and variance vector through different paths.

For the mean vector, it is important to model the interaction between event components so as to capture any subtle change of semantics. Therefore, we introduce a tensor-composition-based path to calculate the mean vector of an event Gaussian embedding. The input to the triple encoder is word embeddings of S, P and O. For the event components comprising of multiple words, the average of their constituent word embeddings is applied. As shown in Figure 3, S and P, P and O are firstly composed to produce the intermediate representation m_1 and m_2:

$$\begin{aligned} m_1 &= T(s, p) = f(s^T U_1 p + b_1), \\ m_2 &= T(p, o) = f(p^T U_2 o + b_2), \end{aligned} \tag{8}$$

where $f = tanh$ is a non-linear function applied element-wise, $b \in \mathbb{R}^k$ is the bias vector, $U \in \mathbb{R}^{k \times d \times d}$ are tensors, which are sets of matrices, each with $d \times d$ dimensions. The tensor product $s^T U p$ will produce a vector $h \in \mathbb{R}^k$, where $h_i = \sum_{j,k} U_{ijk} s_j p_k$. Then the mean vector μ is calculated by the tensor composition of the intermediate representations m_1 and m_2:

$$\mu = T(m_1, m_2) = f(m_1^T U_3 m_2 + b_3) \tag{9}$$

For the variance vector, we assume that it mainly captures the variety of expressions relating to the same event, which is mainly determined by the predicates of events, not the multiplicative interactions between s and p or p and o. We thus introduce an additive-composition-based path to model the variance vector of an event Gaussian embedding. First, the intermediate representation m_3 and m_4 are calculated in a similar way as that in mean vector calculation:

$$\begin{aligned} m_3 &= A(s, p) = f(W_1[s, p] + b_4) \\ m_4 &= A(p, o) = f(W_2[p, o] + b_5) \end{aligned} \tag{10}$$

where $W \in \mathbb{R}^{k \times 2d}$ is weight matrix. The variance vector σ^2 is calculated by the additive composition of the intermediate representations m_3 and m_4:

$$\sigma^2 = A(m_3, m_4) = f(W_3[m_3, m_4] + b_6) \tag{11}$$

Visual Event Components-aware Image Encoder

The goals of image encoder is to map an event image v to a point embedding $\boldsymbol{v}$, where the image embeddings of similar triples should be clustered and image embeddings of dissimilar triples should be separated. To capture the event semantics in the image, we proposed to detect visual event components in the image and compose their representations to get event image embedding. The architecture of our proposed event image encoder is shown in Figure 4.

To enable the model to detect visual event components, We first train an image classifier on the ImSitu dataset, which introduces the task of situation recognition (Yatskar, Zettlemoyer, and Farhadi 2016). In ImSitu, every image is annotated with an activity and roles (called attributes) which are participants in the activity. For example, an image of vet clipping dog's claw will be annotated as activity *clipping* with *vet* as agent and *dog* as source. The task is to identify each attribute given an image. In this paper, we pre-train the image encoder to identify three attributes in ImSitu, *subject*, *activity* and *source*, which are corresponding to *subject*, *predicate* and *object* in an event triple.

Since by now most of object detection methods can only deal with a limited set of object types, thus failing to detect a large variety of objects in real-world event images.

Inspired by (Li et al. 2020), we employ the attention mechanism to extract open-vocabulary event components. In our model, we use a VGG-16 CNN to extract an overall image feature g and a 7×7 convolutional feature map for each image v: $k_{i,j} = CNN(v)$, which can be regarded as attention keys for 7×7 local regions. Taking the $subject$ as an example, the role query vector q_s is constructed by concatenating the role embedding s with the image feature g as context:

$$q_s = f(W_q[s, g] + b_q) \tag{12}$$

Then for an event image v, we calculate the attention score of role subject s to each of the local regions:

$$h_{ij} = \frac{exp(q_s k_{ij})}{\sum_{m,n} exp(q_s k_{mn})} \tag{13}$$

The representation of subject s in an image v is obtained by:

$$r_s = \sum_{i,j} h_{ij} t_{ij} \tag{14}$$

The representation of source (object) r_o is obtained with the same procedure as that for subject. The representation of activity (predicate) is obtained by feeding the overall image feature g into an MLP: $r_p = MLP(g)$. In the pre-training stage, we feed the extracted representations of subject and source as well as the representation g of the whole image i into the classification layers to perform situation recognition task. After pre-training, we replace the classification layer with a fully-connected layer to compose the representations of event arguments and obtain the final representation of an image e_i:

$$\boldsymbol{v} = f(W_v[r_s, r_p, r_o] + b_v) \tag{15}$$

Training

In this part, we describe how the triple encoder and the image encoder are jointly trained under the proposed MERL framework. The training objective function consists of three terms: the intra-triple loss the intra-image loss and the cross-modal loss.

For the intra-triple loss, we introduce a max-margin loss to encourage similar events having higher similarity score than the negative pairs. The similarity score is measured by the Bhattacharyya distance between two distributions:

$$l_1 = \sum_j d(t_j, t_j^p) + \max\{0, \alpha - d(t_j, t_j^n)\} \tag{16}$$

where t_j denotes an event triple, t_j^p denotes a positive sample (an event similar to t_j) and t_j^n denotes a negative sample (an event dissimilar to t_j). For diagonal Gaussian distribution, the Bhattacharyya distance can be simplified as:

$$d(t_1, t_2) = \frac{1}{8}(\mu_1 - \mu_2)^T \Sigma^{-1} (\mu_1 - \mu_2) + \frac{1}{2} \log\left(\frac{\det \Sigma}{\sqrt{\det \Sigma_1 \det \Sigma_2}}\right) \tag{17}$$

where $\Sigma = \frac{\Sigma_1 + \Sigma_2}{2}$. In our cases, all Σs are diagonal matrices.

For the intra-image loss, the objective function is similar to that of the triple encoder. A max-margin loss is introduced to encourage the encoder persevere the semantic relationships of event images in the embeddings space:

$$l_2 = \sum_{j,k} \|v_{jk}, v_{jk}^p\|_2 + \max\{0, \alpha - \|v_{jk}, v_{jk}^n\|_2\} \tag{18}$$

where $\|\cdot\|_2$ denotes Euclidean distance, v_{jk} denotes an event image, v_{jk}^p denotes a positive sample (i.e. an image similar to v_{jk}) and v_{jk}^n denotes a negative sample (an image dissimilar to v_{jk}).

The last loss is to align the results produced by the two encoders, so that the knowledge from two modalities could be utilized to mutually enhance the event density embeddings and image embeddings learned. Under our hypothesis, Equation (7) could be rewritten as:

$$l_3 = \sum_j log\mathcal{N}(\boldsymbol{v}_{j1}, ., \boldsymbol{v}_{jk} | \mu_j, \sigma_j^2) - \sum_{j,k} log\mathcal{N}(\boldsymbol{v}_{jk} | \bar{x}, S^2) \tag{19}$$

where $\bar{x}$ is the average of $\boldsymbol{v}_s$ and S^2 is sample deviation. They are MLE of the parameters of the Gaussian distribution.

The final objective function is the weighted sum of aforementioned three losses plus the l_2 norm of all parameters:

$$l = \alpha l_1 + \beta l_2 + \gamma l_3 + \lambda \|\Theta\|_2^2 \tag{20}$$

The whole training procedure is shown in Algorithm 1.

Algorithm 1: Training MERL

Input : multimodal event dataset: $T = \{t_j\}, V = \{v_{jk}\}$
Output: multimodal event embeddings: $\{\boldsymbol{t}_j\}, \{\boldsymbol{v}_{jk}\}$
for $t_j \in T$ **do**
  update loss l_1 with positive and negative event triple: t_j^p, t_j^n;
  for $v_{jk} \in V_j$ **do**
    sample positive event image v_{jk}^p;
    sample negative event image v_{jk}^n;
    update loss l_2.
  end
  update loss l_3.
end

For joint training, we build a multimodal dataset by extending the hard similarity events dataset proposed in (Ding et al. 2019). The original dataset consists of 1,000 event triples, each event is paired with a positive sample and negative sample. The positive samples are events having strong semantic relationships but with very little lexical overlap (e.g., *police catch robber / authorities apprehend suspect*), and the negative samples are events from distinct scenarios but with high overlap (e.g., *police catch robber / police catch disease*). To extend the dataset to multimodal, each event triple is used to query Google Image to retrieve 20 candidate images, which are filtered by human annotators to keep the top 10 most relevant ones. Our final multimodal event similarity dataset consists of 3,000 event triples paired with 30,000 event images.

Method		Multimodal Hard Similarity (Accuracy%)	Transitive Sentence Similarity (ρ)
Text	Additive	33.0	0.63
	Tensor	40.0	0.60
	Predicate Tensor	43.5	0.64
	Role Factored Tensor	41.0	0.63
	MERL_triple (unimodal training)	44.3	0.61
	MERL_triple (multimodal training)	**52.2**	**0.68**
Image	VGG-16	25.2	-
	MERL_image (unimodal training)	40.9	-
	MERL_image (multimodal training)	**47.0**	-

Table 1: Experimental results on multimodal event similarity task. The best results are in bold.

Experiments

We evaluate our proposed MERL on a variety of downstream tasks, including multimodal event similarity, script event prediction and cross-media event retrieval. We will report the datasets, baselines for comparison, evaluation metrics and results in detail.

Multimodal Event Similarity

The main purpose of event representation learning is to preserve the semantic information in events as much as possible, so that similar events are projected close to each other, while dissimilar events are far away from each other in the embedding space.

Datasets To evaluate MERL, we perform experiments on the following datasets:
Multimodal hard similarity dataset: Weber et al. (2018) propose a event similarity dataset which contains 230 pairs of events (115 pairs of similar types and 115 pairs of dissimilar types). We extend the Weber's hard similarity dataset to multimodal with the same procedure as described earlier for the hard similarity events dataset, resulting in 345 event triples paired with 3,450 event images. For each method, we calculate the similarity score of event pairs under representation, and report the fraction of cases that a similar pair having a higher score than a dissimilar one.
Transitive sentence similarity dataset, (Kartsaklis et al., 2014), contains 108 pairs of transitive sentences (i.e. short sentences contain a single subject, verb and object). Every pair is annotated by several annotators with a similarity score ranging from 1 to 7. For each method, we used the Spearman's correlation ρ between the similarity scores and average annotation scores as evaluation metrics.

Baselines The following baselines are included in the multimodal event similarity experiments:

- **Additive Compositional Model** (**Additive**)
- **Tensor Compositional Model** (**Tensor**)
- **Predicate Tensor Model** (**Predicate Tensor**) (Weber et al. (2018)): represents predicate as a tensor then compose subject and object based on this tensor.
- **Role Factored Tensor Model** (**Role Factored Tensor**) (Weber et al. (2018)): replaces the tensor composition between intermediate representations of subject & predicate and predicate & object with a one-layer neural network to obtain the final embedding.
- **VGG-16 CNN** (**VGG-16**) (Simonyan and Zisserman (2014)): serves as the feature extractor of our event image encoder.
- **MERL**: the triple encoder and image encoder of MERL are used in the text and image modal evaluation respectively.

Results From the results in Table 1 we can observe that after multimodal training, our triple encoder and image encoder both outperform other baselines and the variant of MERL trained with unimodal data only. This verifies our hypothesis that leveraging knowledge from both modalities enhances the quality of event embeddings learned in each modality. We also notice that some methods based on external knowledge bases achieve remarkable performance (Ding et al. 2016, 2019). (Ding et al. 2019) even achieves a accuracy over 70% on hard similarity task. However, multimodal learning of event representation is not conflict with these methods. Combining multimodal event data and external knowledge base may achieve better performance.

Script Event Prediction

Events carry world knowledge and play an important role in natural language inference tasks. A common-sense reasoning task could be used to evaluate how much the world knowledge information the learned event embeddings capture. Chambers and Jurafsky (2008) proposed the *narrative cloze task*. In this task, a series of events are extracted from one document, but one of the events is masked. The reasoning model is asked to predict the masked one out of two candidate events. Granroth-Wilding and Clark (2016) extended this task to multiple-choices and proposed the *multiple choice narrative cloze* (MCNC) task. Following Li, Ding, and Liu (2018), we evaluate the proposed method and other baselines on the standard MCNC dataset.

Multiple choice narrative cloze dataset To perform script event prediction, Li, Ding, and Liu (2018) extracted event chains from the New York Gigaword corpus with the same procedure as that in Granroth-Wilding and Clark (2016). The dataset contains 140k samples for training and 10k samples for testing. For each event chain, 5 candidate events are provided with 1 correct answer.

Method	Accuracy%
Additive	49.57
PairLSTM	50.83
SGNN	52.45
SGNN(MERL)	**53.47**
SGNN+Additive	54.15
SGNN+PairLSTM	52.71
SGNN+Additive+PairLSTM	54.93
SGNN(MERL)+Additive+PairLSTM	**55.51**

Table 2: Experimental results on script event prediction task. + denotes a combination of methods by aggregating the results of different methods statistically. The best results are in bold.

Baselines Script event prediction requires the modeling of sequential relations between events. As the Scaled Graph Neural Network (SGNN) model proposed by Li, Ding, and Liu (2018) provides a strong baseline of event sequence modeling, we use the framework of SGNN and replace their event embeddings with the ones generated by MERL.

- **Scaled Graph Neural Network** (**SGNN**) (Li, Ding, and Liu (2018)): constructs a narrative event graph and employs a graph network to model event chains.
- **PairLSTM** (Wang et al. (2017)): simultaneously models pairwise relationship and sequential relationship of events.
- **SGNN(MERL)**: replaces the event embeddings of SGNN with the mean vector of Gaussian distributions of event triples.

Results The top half of Table 2 shows the results of each individual method, and the bottom half of the table shows the aggregated results of multiple methods as has been previously done in (Li, Ding, and Liu 2018). We can observe that MERL outperforms other baselines by a small margin. Nevertheless, as have been previously reported in (Li, Ding, and Liu 2018), MCNC is a difficult task that even 1% improvement is considerable. We can conclude that multimodal learning of event embeddings indeed generates better event embeddings which benefit script event prediction.

Cross-modal Event Retrieval

Multimodal representation learning methods project information from two modalities and learn their relationships, making cross-modal retrieval possible. In this subsection, we evaluate the performance of MERL on cross-modal event retrieval.

Cross-modal event retrieval dataset We modify the multimodal hard similarly dataset (Weber) to perform multimodal event retrieval. For event triple retrieval, we use a randomly selected image as a query and aim to retrieve the best-matched event triple out of a randomly-constructed event triple set, which consists of the originally paired event triple and 19 randomly sampled event triples. For event image retrieval, we follow a similar set up that given an event triple, we aim to retrieve the best-matched event image out of a randomly-constructed image set in which one of the images is the desired one. We report the recall@10 results returned by each retrieval method.

Methods	Triple Retrieval	Image Retrieval
Order Embedding	54.8	51.3
Hierarchical Order Embedding	**61.0**	56.2
MERL	53.7	**69.4**

Table 3: Experimental results on multimodal event retrieval. The best results are in bold.

Baselines We assume an event triple is a more abstract description of its paired images. Such an assumption essentially implies a hierarchical relationship between an event triple and its associated images. In this set of experiments, we mainly compare MERL with multimodal representations learning methods that consider hierarchical relationships between text and image:

- **Order Embedding**, (Vendrov et al. 2016), defines a partial order relationship between text and image and represents texts and images only based on this asymmetric relationship. In the evaluation, each event triple and image pair is scored and ranked based on their respective order embeddings.
- **Hierarchical Order Embedding**, (Ben and Andrew 2018), extends the framework proposed by Vendrov et al. (2016) by replacing the point estimations with density estimations.
- **MERL**, the likelihood between image embedding and triple density embedding is employed as measurement of similarity.

Results As shown in Table 3, MERL outperforms other two baselines on image retrieval by a large margin of over 13%, but performs worse compared to either Order Embedding or Hierarchical Order Embedding. One possible reason is that it is more natural to rank the relatedness of points given a distribution, but not vice versa.

Conclusion

In this paper, we have proposed a multimodal event representation learning framework (MERL) which maps event triples and event-associated images onto heterogeneous embedding spaces. More concretely, MERL projects event triples onto a Gaussian embedding space with a dual-path Gaussian triple encoder, and event images onto a point embedding space by the visual event components-aware image encoder. To measure between these two heterogeneous spaces, a novel score function inspired by hypothesis testing has been proposed. Our experiments demonstrate that learning event embeddings from two modalities generate more informative embeddings compared with learning from text only, leading to generally better results in downstream tasks such as multimodal event similarity measurement, script event prediction, and cross-modal event retrieval. Future work may contain extend the proposed method in a Bayesian learning framework.

Acknowledgments

This work is funded by the National Key Research and Development Program of China (2016YFC1306704), the National Natural Science Foundation of China (61772132), the EPSRC (grant no. EP/T017112/1, EP/V048597/1) and a Turing AI Fellowship funded by the UK Research and Innovation (UKRI) (grant no. EP/V020579/1). The authors would like to thank the anonymous reviewers for the insightful comments.

References

Ashutosh, M.; and Ivan, T. 2014. Learning Semantic Script Knowledge with Event Embeddings. In *ICLR 2014 workshop*.

Ben, A.; and Andrew, W. 2018. Hierarchical Density Order Embeddings. In *6th International Conference on Learning Representations (ICLR)*.

Casella, G.; and Berger, R. L. 2002. *Statistical inference*, volume 2. Duxbury Pacific Grove, CA.

Chambers, N.; and Jurafsky, D. 2008. Unsupervised learning of narrative event chains. In *Proceedings of ACL-08: HLT*, 789–797.

Ding, X.; Liao, K.; Liu, T.; Li, Z.; and Duan, J. 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)*, 4894–4903.

Ding, X.; Zhang, Y.; Liu, T.; and Duan, J. 2015. Deep learning for event-driven stock prediction. In *Twenty-fourth international joint conference on artificial intelligence (IJCAI)*, 2327–2333.

Ding, X.; Zhang, Y.; Liu, T.; and Duan, J. 2016. Knowledge-driven event embedding for stock prediction. In *Proceedings of coling 2016, the 26th international conference on computational linguistics: Technical papers*, 2133–2142.

Frome, A.; Corrado, G. S.; Shlens, J.; Bengio, S.; Dean, J.; Ranzato, M.; and Mikolov, T. 2013. Devise: A deep visual-semantic embedding model. In *Advances in neural information processing systems*, 2121–2129.

Granroth-Wilding, M.; and Clark, S. 2016. What happens next? event prediction using a compositional neural network model. In *Association for the Advancement of Artificial Intelligence (AAAI)*, 2727–2733.

Kartsaklis, D.; and Sadrzadeh, M. 2014. A Study of Entanglement in a Categorical Framework of Natural Language. In *QPL*.

Kendall, A.; and Gal, Y. 2017. What uncertainties do we need in bayesian deep learning for computer vision? In *Advances in neural information processing systems*, 5574–5584.

Lee, I.-T.; and Goldwasser, D. 2018. FEEL: Featured Event Embedding Learning. In *AAAI*, 4840–4847.

Li, M.; Zareian, A.; Zeng, Q.; Whitehead, S.; Lu, D.; Ji, H.; and Chang, S.-F. 2020. Cross-media Structured Common Space for Multimedia Event Extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2557–2568.

Li, Z.; Ding, X.; and Liu, T. 2018. Constructing Narrative Event Evolutionary Graph for Script Event Prediction. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, 4201–4207.

Modi, A. 2016. Event embeddings for semantic script modeling. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, 75–83.

Oh, S. J.; Murphy, K.; Pan, J.; Roth, J.; Schroff, F.; and Gallagher, A. 2018. Modeling uncertainty with hedged instance embedding. *Proceedings of the International Conference on Learning Representations (ICLR)* .

Pichotta, K.; and Mooney, R. J. 2016. Learning Statistical Scripts with LSTM Recurrent Neural Networks. In *Association for the Advancement of Artificial Intelligence (AAAI)*, 2800–2806.

Rajagopalan, S. S.; Morency, L.-P.; Baltrusaitis, T.; and Goecke, R. 2016. Extending long short-term memory for multi-view structured learning. In *European Conference on Computer Vision*, 338–353. Springer.

Silberer, C.; and Lapata, M. 2014. Learning grounded meaning representations with autoencoders. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 721–732.

Simonyan, K.; and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* .

Vendrov, I.; Kiros, R.; Fidler, S.; and Urtasun, R. 2016. Order-Embeddings of Images and Language. In *4th International Conference on Learning Representations, ICLR 2016*.

Wang, Z.; Zhang, Y.; and Chang, C.-Y. 2017. Integrating Order Information and Event Relation for Script Event Prediction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 57–67.

Weber, N.; Balasubramanian, N.; and Chambers, N. 2018. Event Representations with Tensor-based Compositions. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18)*, 4946–4953.

Xiao, Y.; and Wang, W. Y. 2019. Quantifying uncertainties in natural language processing tasks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 7322–7329.

Yatskar, M.; Zettlemoyer, L.; and Farhadi, A. 2016. Situation recognition: Visual semantic role labeling for image understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5534–5542.