GauSE: Gaussian Enhanced Self-Attention for Event Extraction

Anonymous ACL submission

Abstract

Event Extraction (EE) has benefited from pre-trained language models (PLMs), in which the self-attention mechanism could pay attention to the global relationship between triggers/arguments and context words to enhance performance. However, existing PLM-based methods are not good enough at capturing local trigger/argument-specific knowledge. To this end, we propose a **Gau**ssian enhanced **S**elf-attention **E**vent extraction framework (GauSE), which models the syntactic-related local information of trigger/argument as a Gaussian bias for the first time, to pay more attention to the syntactic scope of the local region. Furthermore, existing methods rarely consider multiple occurrences of the same triggers/arguments in EE. We explore the global interaction strategies among multiple localness of the same triggers/arguments to fuse the corresponding distributions and capture more latent information scopes. Compared to traditional GCN-based models, our methods could introduce syntactic relationships without over-smoothing problem in deep GCN layers. Experiments on EE datasets demonstrate the effectiveness and generalization of our proposed approach.

1 Introduction

Event extraction is an essential information extraction (IE) task, aims to extract event structures from unstructured event mentions. It consists of event detection (ED) and event argument extraction (EAE). For example, in the event mention "CNN's Kelly Wallace reports on today's attack in Netanya.", ED model should identify the event trigger "attack" and classify the event type "Conflict:Attack", EAE model should identify the event arguments "today" and "Netanya", then classify argument roles "Time-Within" and "Place". The extracted event structures could benefit numerous downstream tasks, such as biomedical science (Li et al., 2019; Wang et al., 2020), financial analysis (Deng et al., 2019; Liang et al., 2020), information retrieval (Glavas and Snajder, 2014), and so on.

Existing EE methods mainly focus on feature engineering. Inspired by the significant performance of PLMs for various NLP tasks, some prior work (Wang et al., 2019; Wadden et al., 2019) utilizes general PLMs, such as BERT (Devlin et al., 2019), to construct global dependencies among context words by self-attention. Although leveraging PLMs has improved EE performance, it still cannot capture the local information of specific triggers or arguments. As shown in Figure 1, in the sentence "Attack happened without declaration of war, the attack was judged in trials.", the first trigger "attack" is more important to the second trigger "attack" than other words when computing self-attention, trigger-specific information is not strengthened enough by the original self-attention.

To avoid the information insufficiency problem, dependency tree based Graph Convolution Network (GCN) (Nguyen and Grishman, 2018; Liu et al., 2018; Yan et al., 2019) was adopted to capture syntactic relations between triggers and related words. However, in the literature, the information introduced by GCN still exists some problems: (1) GCN mainly focuses on the nearest syntactic neighbors (Nguyen and Grishman, 2018; Liu et al., 2018), as shown in Figure 2, the second trigger "attack" just attaches 2-hop syntactic neighbor "happened", early stop of message passing limits the capture of neighbors; (2) in multi-hop message passing, due to over-smoothing (Zhou et al., 2020) in deep layers of GCN, the importance of "War" will gradually disappear to "happened", this phenomenon further decreases deeper information passing; (3) existing GCN-based methods have not modeled the relationship among the same triggers/arguments occur multiple times in event mentions.

Another general localness-enhanced self-attention method is modeling local regions as Gaussian priors (Yang et al., 2018; Guo et al.,

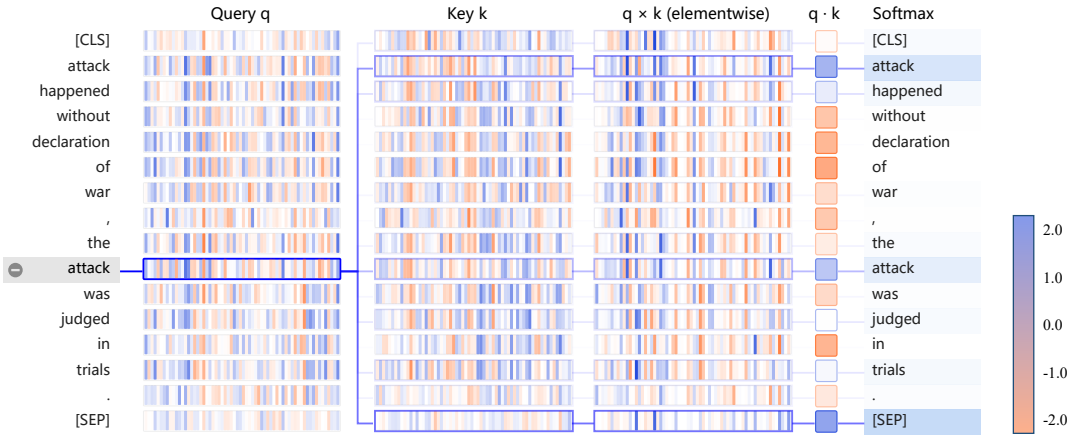


Figure 1: Attention score in original self-attention, the connecting line colors are weighted by the attention values.

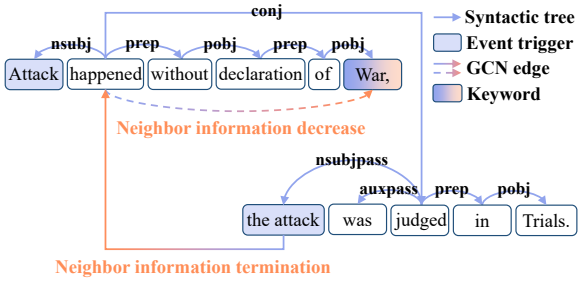


Figure 2: Syntactic dependency parsing.

2019), which could enhance the neighbor text spans of central words. Compared to GCN-based models, Gaussian-based methods could expand the neighbor scopes and alleviate the over-smoothing problem without changing the model structure.

In this paper, we combine the above methods in EE for the first time. Specifically, we enrich the trigger/argument specific information using Gaussian prior probability over the corresponding window (i.e. the deviation of syntactic dependency distribution) to the central word (i.e. the position of trigger/argument), and further enhance the original self-attention.

The above method is adapted to the situation that triggers/arguments appear once in event mentions. However, the same triggers/arguments may occur multiple times in the same mentions, according to our statistics on the ACE-2005 dataset, 13.18% (1959/14862) of mentions have the same triggers/arguments occurring multiple times. We hypothesize that the same triggers/arguments in the same mentions are identical, modeling the relationship among multiple occurrences of the same triggers/arguments is beneficial. We further explore several fusion methods among Gaussian priors to capture latent knowledge. Specifically, we adopt Gaussian multiplication and GMM among the identical trigger/argument distributions to pay attention to intermediate words. In addition, we regularize the output distributions of the same triggers/arguments to be consistent by minimizing the Wasserstein (WA) divergence among the outputs.

Our contributions are summarized as follows:

- We propose a novel Gaussian-enhanced self-attention framework, which aims to alleviate the trigger/argument specific syntactic information insufficiency problem in EE for the first time and better capture local dependencies of event mention sequences.
- We propose efficient Gaussian high-order interaction mechanisms to promote knowledge fusion. In addition, we adopt a novel distribution metric loss to strengthen the model's generalization.
- Experiments on several datasets indicate that GauSE achieves significant performances on both overall and few-sample settings.

2 Related Work

Event Extraction. Most of existing EE models rely on feature construction. Traditional methods (Ji and Grishman, 2008; Gupta and Ji, 2009; Li et al., 2013) mainly focus on manual features. Recently, neural network based models have been widely used to extract features automatically, such as convolutional neural networks (CNN) (Nguyen and Grishman, 2015; Chen et al., 2015), recurrent neural networks (RNN) (Nguyen et al., 2016),

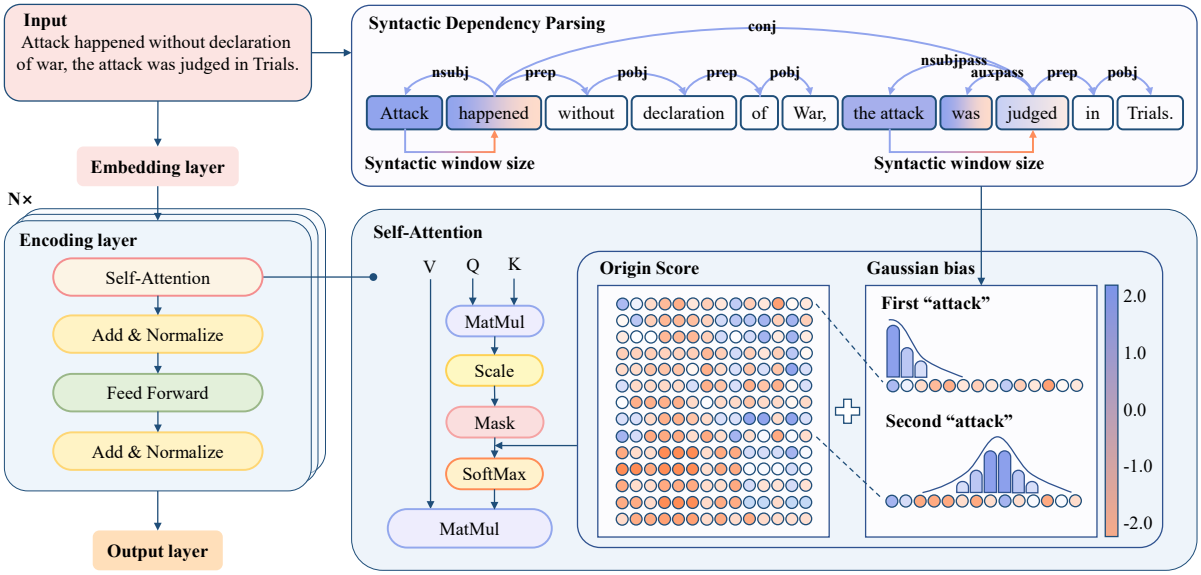


Figure 3: Illustration of the proposed single Gaussian-based localness modeling approach.

graph neural networks (GNN) (Nguyen and Grishman, 2018; Lai et al., 2020), and PLM-based models because of their significant performance (Wang et al., 2019; Yang et al., 2019; Wadden et al., 2019; Tong et al., 2020). Existing NN models regard triggers/arguments and other context words as the same, trigger/argument specific information cannot be captured. In this work, we enhance the weight of specific local knowledge to better utilize rich syntactic information in event mentions.

Localness Modeling of Self-Attention. To improve the quality of designed features, besides linguistic features, syntactic information has been explored in many works. Dependency-bridge based RNN (Sha et al., 2018) introduced the dependency tree into EE. Syntactic dependency tree based GCN (Nguyen and Grishman, 2018; Liu et al., 2018; Yan et al., 2019) promoted information propagation over graphs. Although these works could capture syntactic information, over-smoothing still limits message passing. Furthermore, the relations among the same triggers/arguments occurring multiple times in event mentions have not been explored.

Several works have demonstrated that explicitly modeling localness by Gaussian bias benefits more, such as restricting self-attention on neighbor information (Sperber et al., 2018), adopting relative neighbor position encoding between tokens (Shaw et al., 2018), dynamically adjusting attention distribution (Yang et al., 2018; Guo et al., 2019) to better model localness. Compared to GCN-based models, these methods could introduce local knowledge without complicated model architecture.

In this work, we adjust the syntactic information into a Gaussian-enhanced self-attention mechanism for the first time, which alleviates information loss and explores the interactions among several Gaussian priors, to model latent knowledge efficiently.

3 Methodology

The overall GauSE framework consists of three Gaussian-based enhancement strategies: (1) Single Localness Modeling, (2) Localness Fusion Learning, (3) Localness Regularization Learning.

3.1 Preprocessing

To build syntactic-aware neighbor signals for modeling trigger/argument specific information, we use the automatic syntactic dependency parser spaCy[1] to parse event mentions into syntactic dependency tree structures. Formally, given an event mention, the dependency tree structure consists of tokens as nodes and syntactic dependencies as edges.

3.2 Single Localness Modeling

The original self-attention mechanism computes attention scores as Figure 1. Specifically, given an input event mention $s = \{w_1, ..., w_n\}$ contains n tokens, the hidden state H of s is calculated by the transformed queries $Q \in \mathbb{R}^{n \times d}$, keys $K \in \mathbb{R}^{n \times d}$, and values $V \in \mathbb{R}^{n \times d}$ as follows:

$$H = Att(Q, K, V) \quad (1)$$

[1] https://spacy.io/api/dependencyparser

where $Att(\cdot)$ means the dot-product self-attention mechanism, which is defined as:

$$Att(Q, K, V) = softmax(Score_{Ori})V \quad (2)$$

$$Score_{Ori} = \frac{QK^T}{\sqrt{d}} \quad (3)$$

The original calculation regards all words as the same, even the syntactic information that dependency-based GCN introduced is limited to over-smoothing. To efficiently model trigger/argument specific syntactic information, we suppose that trigger t_i or argument a_i represents the central word of the event mention s; further, we strengthen the importance of tokens in the range of 1-hop syntactic neighbors. We hypothesize that 1-hop neighbors have the most critical syntactic information. Considering the distance factor, the syntactic contributions of different tokens to central words obey the normal distribution. We model this phenomenon via Gaussian prior in this paper.

Specifically, as shown in Figure 3, the Gaussian bias $G \in \mathbb{R}^{n \times n}$, where n is the length of event mention, is defined to strengthen the original self-attention score, i.e. $Score_{Ori}$ in Equation 2:

$$Att(Q, K, V) = softmax(Score_{Ori} + G)V \quad (4)$$

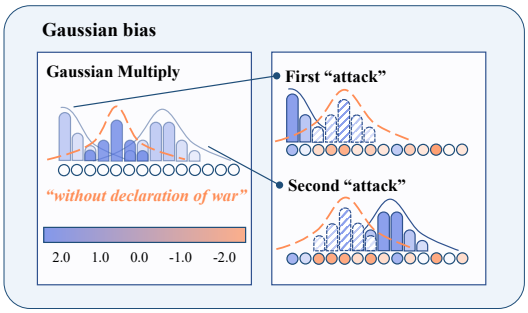
The element $G_{i,j} \in (-\infty, 0]$ measures the syntactic distance between context token w_j in the 1-hop syntactic neighbor scope and the central trigger/argument word w_i, defined as:

$$G_{i,j} = -\frac{(P_j - P_i)^2}{2{\sigma_i}^2} \quad (5)$$

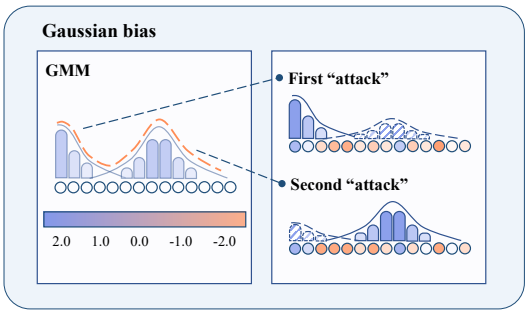
where P_j and P_i are the positions of neighbor and central tokens, σ_i represents the standard deviation, which is generally defined as $\frac{D_i}{2}$, D_i is the window size of the corresponding central word, i.e. the distance of 1-hop syntactic neighbor to specific trigger/argument. Furthermore, due to the exponential operation in $softmax$, the enhanced Gaussian bias equates to multiplying the original self-attention score by a Gaussian weight.

3.3 Localness Fusion Learning

Since the Gaussian bias was introduced independently for each central word, it may be beneficial to consider them simultaneously, when the same triggers/arguments appear multiple times in the event mentions. As shown in Figure 2, the trigger "attack" occurs twice in the same event mention, and they represent the same event type about "war", the interaction between them could provide beneficial syntactic neighbor knowledge to each other. Specifically, we adopt localness interaction strategies, i.e. Gaussian multiplication and GMM, to promote message passing among several Gaussian distributions of the same triggers/arguments.



(a) Gaussian Multiplication.



(b) GMM.

Figure 4: Localness fusion learning strategies.

Considering the tokens that simultaneously appear in the same trigger/argument neighbors should be enhanced, as shown in Figure 4(a), the first trigger word "attack" has syntactic neighbor context region "attack happened without", while the second "attack" has neighbor "of war, the attack was judged", the text "happened without declaration of war", which appears in the interaction of several syntactic neighbor texts of "attack", should be paid more attention to, we adopt Gaussian multiplication to obtain the corresponding latent distribution:

$$G_{i,j} = -\frac{(P_j - P_i)^2}{2\sigma_i^2} - \frac{(P_j - \mu_{mul})^2}{2\sigma_{mul}^2} \quad (6)$$

$$\mu_{mul} = \sigma_{mul}^2 \sum_{k=1}^{N} \frac{\mu_k}{{\sigma_k}^2} \quad (7)$$

$$\frac{1}{\sigma_{mul}^2} = \sum_{k=1}^{N} \frac{1}{{\sigma_k}^2} \quad (8)$$

where μ_k and σ_k denote the positions and syntactic window sizes of same triggers/arguments, μ_{mul} and σ_{mul} denote the latent Gaussian bias.

To combine several Gaussian distributions of the same trigger/argument words, as shown in Figure 4(b), we adopt GMM enhancement strategy to promote message passing of co-occurrence syntactic neighbor regions, which is defined as:

$$G_{i,j} = -\alpha \frac{(P_j - P_i)^2}{2{\sigma_i}^2} - \sum_{k=1}^{N} \beta_k \frac{(P_j - P_k)^2}{2{\sigma_k}^2} \quad (9)$$

where α and β_k denote the weights of original and other Gaussian biases of the same triggers/arguments, in this way, all of the Gaussian biases of triggers/arguments that exist multiple times in the same event mentions could be weighted to share knowledge between each other.

Considering the exponential operation in $softmax$, we further define GMM as follows:

$$\begin{aligned} G_{i,j} = &\log\Big(\frac{\alpha}{\sqrt{2\pi{\sigma_i}^2}} \exp - \frac{(P_j - P_i)^2}{2{\sigma_i}^2} \\ &+ \sum_{k=1}^{N} \frac{\beta_k}{\sqrt{2\pi{\sigma_k}^2}} \exp - \frac{(P_j - P_k)^2}{2{\sigma_k}^2}\Big) \end{aligned} \quad (10)$$

The above interaction strategies fuse multiple Gaussian distributions, so that all of the Gaussian enhanced knowledge of the same triggers/arguments could be paid more attention to simultaneously. In this way, message passing among multiple distributions could be realized more efficiently. Besides, our method is compatible with the original BERT in model parameters and could be applied conveniently.

3.4 Localness Regularization Learning

Considering most of the same triggers/arguments occurring multiple times in the same event mentions are identical in EE, we hope the information learned from the same triggers/arguments Gaussian distributions could be similar. Specifically, the corresponding distributions of predictions $P_k^{G_k}(y||x_i)$ should be more consistent, where G_k means the k_{th} Gaussian distribution of the same triggers/arguments.

As shown in Figure 5, since the output predictions $P_j^{G_j}(y||x_i)$ and $P_k^{G_k}(y||x_i)$ of the same triggers/arguments specific Gaussian enhanced knowledge are different, where G_j and G_k means the j_{th} and k_{th} Gaussian distribution of the same triggers/arguments, the predictions are different for the same input event mention x_i. To alleviate the inconsistency problem, we adopt a distribution divergence metric, i.e. WA Distance[2], to regularize the corresponding prediction distributions:

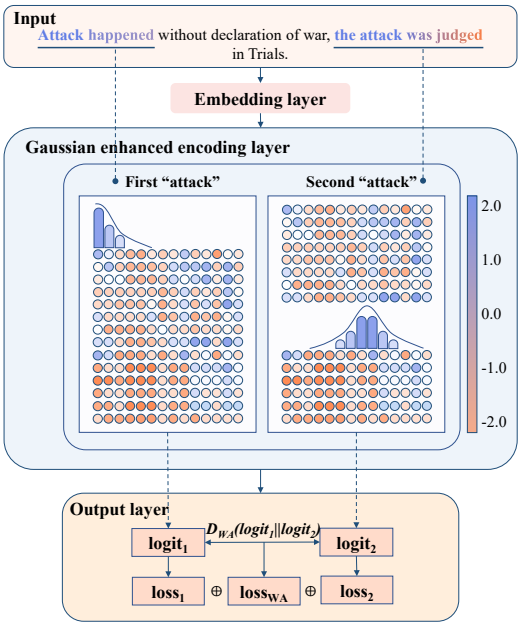


Figure 5: Localness regularization learning by Wassertein Distance metric.

$$L_{WA}^i = \frac{\sum_{j=1}^{N-1}\sum_{k=j+1}^{N} WA(P_j^{G_j}(y||x_i), P_k^{G_k}(y||x_i))}{N(N-1)/2} \quad (11)$$

With the negative log-likelihood learning objective L_{NLL}^i of Gaussian enhanced predictions for given training data (x_i, y_i):

$$L_{NLL}^i = -\frac{\sum_{j=1}^{N} log(P_j^{G_j}(y_i||x_i))}{N} \quad (12)$$

Based on these training objectives, we define the final training objective as to minimize L_{total}^i for the input event mention x_i:

$$\begin{aligned} L_{total}^i = &L_{NLL}^i + L_{WA}^i \\ = &-\frac{\sum_{j=1}^{N} log(P_j^{G_j}(y_i||x_i))}{N} \\ &+ \frac{\sum_{j=1}^{N-1}\sum_{k=j+1}^{N} WA(P_j^{G_j}(y||x_i), P_k^{G_k}(y||x_i))}{N(N-1)/2} \end{aligned} \quad (13)$$

In this way, our method further regularizes the model output distributions and enhances the generalization of the original model.

[2]where WA Distance could solve the asymmetry problem in Kullback-Leibler (KL) divergence, avoid the bidirectional calculation between two output distributions.

Dataset	Doc	Ins	EveT
ACE 2005 ED subset	599	4,090	33
ACE 2005 EAC subset	599	4,090	35
OntoEvent	4,115	60,546	100
FewEvent	-	70,852	100

Table 1: Statics of existing widely-used EE datasets. ACE 2005 dataset contains ED and EAC subsets. (Doc: document, Ins: instance, EveT: event type.)

4 Experiments

The experiments aim to demonstrate that GauSE with several Gaussian enhanced strategies could benefit EE tasks in both overall and low-resource settings.

4.1 Datasets

We evaluate our methods on three generally-used EE datasets, including ACE 2005 English subset (Walker et al., 2006), the recently-constructed large-scale OntoEvent dataset (Deng et al., 2021), and FewEvent dataset (Deng et al., 2020). The statics of these datasets are introduced in Table 1. EE performance is assessed with two subtasks: Event Detection (ED) and Event Argument Classification (EAC). In the ACE 2005 dataset, we evaluate both ED and EAC subtasks, while in OntoEvent and FewEvent, we only evaluate ED subtask due to the structures of the corresponding datasets. We further evaluate our model in low-resource scenarios of the FewEvent dataset.

4.2 Baselines

For evaluation, we adopt several official EE baselines, including: (1) vanilla CNN-based model DMCNN (Chen et al., 2015); (2) the model dependent on syntactic dependency knowledge, such as RNN-based model JRNN (Nguyen et al., 2016), GCN-based model JMEE (Liu et al., 2018), graph-based models DYGIE++ (Wadden et al., 2019), OneIE (Lin et al., 2020), PathLM (Li et al., 2020) and OntoED (Deng et al., 2021), joint-based model Joint3EE (Nguyen and Nguyen, 2019); (3) GAN-based model GAIL (Zhang et al., 2019); and (4) some new forms of EE models, such as QA-based model BERT_QA (Du and Cardie, 2020), generation-based paradigm Text2Event (Lu et al., 2021).

We adopt the official BERT-based model DMBERT (Wang et al., 2019) as our baseline to further continue our experiments.

Model	ED			EAC		
	P	R	F1	P	R	F1
DMCNN	75.60	63.60	69.10	**62.20**	46.90	53.50
JRNN	66.00	73.00	69.30	54.20	56.70	55.40
Joint3EE	68.00	71.80	69.80	52.10	52.10	52.10
DYGIE++	-	-	69.70	-	-	48.80
GAIL	74.80	69.40	72.00	61.60	45.70	52.40
OneIE	-	-	74.70	-	-	56.80
PathLM	-	-	73.40	-	-	56.60
BERT_QA	71.12	73.70	72.39	56.77	50.24	53.31
Text2Event	69.60	74.40	71.90	52.50	55.20	53.80
DMBERT	71.60	72.30	70.87	53.14	54.24	52.76
+Gau	73.43	74.23	72.98	53.82	54.45	53.22
+Fusion	74.22	76.24	74.70	56.35	55.12	54.27
+Regularization	**77.16**	**77.96**	**76.80**	57.43	**58.37**	**57.03**

Table 2: Evaluation of EE with various models on ACE 2005. $P(\%)$, $R(\%)$ and $F1(\%)$ represent precision, recall and F1-score respectively.

4.3 Experiment Settings

Gaussian enhanced model settings. Before the detailed experiments, we adopt pre-process method to determine the syntactic neighbors of triggers and arguments. Specifically, we obtain the syntactic dependency tree by spaCy and select 1-hop neighbors to calculate Gaussian enhanced regions.

General settings. We adopt the same model structure as BERT, which is with 12 layers, 768 hidden sizes and 12 attention heads. AdamW (Loshchilov and Hutter, 2017) optimizer is used with the learning rate of 1×10^{-5}, a dropout rate of 0.1 is adopted to avoid over-fitting. The dimension of token embedding is 768, while the maximum length of input event mention is 128. The hyperparameters of α and β are set to 0.5 and 0.5, respectively. We evaluate the performance of EE with Precision (P), Recall (R) and F1 Score (F1). We follow the evaluation protocol of previous EE models, event instances are split into training, validating and testing sets with the ratio of 0.8, 0.1 and 0.1, respectively. We run each method 5 times on all datasets and report the average performances to get stable results.

4.4 Overall Evaluation

The evaluation results are shown in Tables 2 to 4. We can see that:

(1) *By modeling trigger/argument syntactic related regions, GauSE could efficiently utilize latent knowledge which benefits to EE task.* GauSE achieves significant improvements compared to the basic model DMBERT on all datasets and outperforms all baselines, especially models based on syntactic dependency tree, such as JRNN, JMEE and Joint3EE. The results demonstrate the effec-

Model	ED		
	P	R	F1
DMCNN	62.51	62.35	63.72
JRNN	63.73	63.54	66.95
JMEE	52.02	53.80	68.07
AD-DMBERT	67.35	73.46	71.89
OneIE	71.94	68.52	71.77
PathLM	73.51	68.74	72.83
OntoED	75.46	70.38	74.92
DMBERT	80.00	78.30	78.40
+Gau	80.45	79.80	79.70
+Fusion	81.18	80.32	80.30
+Regularization	**82.30**	**80.57**	**80.90**

Table 3: Evaluation of ED with various models on OntoEvent. $P(\%)$, $R(\%)$ and $F1(\%)$ represent precision, recall and F1-score respectively.

Model	ED		
	P	R	F1
DMBERT	81.75	81.83	80.52
+Gau	82.27	82.58	81.30
+Fusion	82.53	83.00	81.65
+Regularization	**83.17**	**83.80**	**82.43**

Table 4: Evaluation of ED with various models on FewEvent. $P(\%)$, $R(\%)$ and $F1(\%)$ represent precision, recall and F1-score respectively.

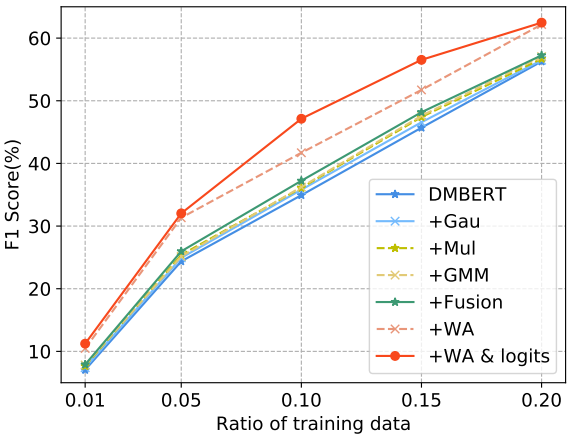


Figure 6: ED performance (F1-score) on FewEvent with different ratios of training data.

Model	ED				
	20%	40%	60%	80%	100%
DMBERT	56.27	72.47	78.17	79.10	80.52
+Gau	56.48	72.77	78.30	79.75	81.30
+Mul	56.70	73.10	78.53	79.90	81.55
+GMM	57.05	73.07	78.50	79.87	81.46
+Fusion	57.25	73.67	79.03	80.23	81.65
+WA	62.17	75.27	79.50	80.73	81.83
+WA & logits	**62.47**	**75.53**	**80.13**	**81.17**	**82.43**

Table 5: Few-sample evaluation with F1-score $(\%)$ performance of event classification with various models on FewEvent.

tiveness of GauSE, and our method could introduce syntactic information more efficiently. Furthermore, using single self-attention enhancement strategies, GauSE surpasses most of the methods based on complicated architectures without additional parameters.

(2) *By combining different Gaussian enhancement strategies, GauSE could achieve different extents of improvements.* The general ablation study indicates that different Gaussian enhancement strategies benefit the model differently. The interaction among Gaussian distributions could promote message passing, so that the improvements of Gau Fusion, which combines Gau Mul and Gau GMM, are more significant than Gau. Gau Regularization could further improve the performance by strengthening the basic model's generalization.

(3) *By enhancing the syntactic neighbor information, our model could extend to other keyword-based tasks.* The experiments on both ED and EAC tasks explicitly excel baselines, implies that our model could leverage and propagate trigger/argument specific syntactic knowledge. Furthermore, we could select keywords in texts in other tasks, then adopt Gaussian enhanced keyword-specific syntactic knowledge to benefit models.

4.5 Few-sample Evaluation

In this section, considering that auxiliary information for the basic model will be more urgent in low-resource EE scenarios, we also study how to influence the performance of our Gaussian enhancement strategies by changing the available training data size. We compare the ED performance, i.e., F1 score results, of all our proposed methods on the FewEvent dataset when trained with different ratios of randomly-sampled training data. We can observe from the experiment results that:

(1) *GauSE is especially beneficial for extremely low-resource EE scenarios.* As shown in Figure 6, the improvements of our Gaussian enhanced strategies compared to the basic model DMBERT are generally more significant when less training data is available. Specifically, in extremely low-resource EE scenarios (training model with less than 20% data), Gau Fusion based strategy obtains 37.23%

Model	ACE ED			ACE AC			ONTO ED			FEW ED		
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
DMBERT	71.60	72.30	70.87	53.14	54.24	52.76	80.00	78.30	78.40	81.75	81.83	80.52
+Gau	73.43	74.23	72.98	53.82	54.45	53.22	80.45	79.80	79.70	82.27	82.58	81.30
+Mul	73.39	75.24	73.56	55.35	54.72	53.72	81.16	80.29	80.13	82.48	82.85	81.55
+GMM	73.18	74.78	73.28	55.15	55.20	54.13	81.00	79.98	79.85	82.78	82.62	81.46
+Fusion	74.22	76.24	74.70	56.35	55.12	54.27	81.18	80.32	80.30	82.53	83.00	81.65
+WA	76.52	**78.00**	76.30	**57.67**	58.03	56.83	81.90	80.10	80.23	**83.57**	82.83	81.83
+WA & logits	**77.16**	77.96	**76.80**	57.43	**58.37**	**57.03**	**82.30**	**80.57**	**80.90**	83.17	**83.80**	**82.43**

Table 6: Ablation study of EE with various models on three datasets. $P(\%)$, $R(\%)$ and $F1(\%)$ represent precision, recall and F1-score respectively.

F1 score with 10% training data, while Gau WA & logits based strategy, which enhances the generalization of the basic model, obtains 47.13% F1 score, in comparison to 34.93% in DMBERT.

(2) *GauSE could achieve better performance with less data constantly.* As shown in Table 5, GauSE obtains more advanced performances with less training data than baseline continuously. Especially, DMBERT requires 100% training data to almost achieve the best performance, while Gau Fusion based strategy only needs 80%. Gau Fusion based strategy could even obtain 81.65% F1 score with overall data, while Gau WA & logits based strategy obtains 82.43%, 1.91% higher than 80.52% in DMBERT. These results demonstrate that GauSE is especially beneficial for low-resource EE tasks, which is essential since the annotation of EE is quite expensive and laborious.

5 Detailed Ablation Analysis

To evaluate the effect of different Gaussian-based enhancement strategies, we study the performances of the corresponding modules, and the ablation results are shown in Table 6. We can observe that:

(1) *Gaussian Fusion strategy could generally guide the generation of more syntactic knowledge.* The introduction of Gaussian enhanced self-attention syntactic region could achieve general improvements based on DMBERT. The F1-score performance of Gau is 1.16% higher than the basic model on the average of all datasets. The Gau Mul strategy surpasses the Gau GMM strategy slightly since Gau Mul generates more unseen information. The fusion of Gau Mul and Gau GMM indicates that promoting message passing among several distributions of the same triggers or arguments could further benefit the model to some extent. The F1-score performance of Gau Fusion is 0.93% higher than Gau on average.

(2) *Gaussian regularization strategy is essential for the generalization of the model.* The comparison between Gau WA and Gau WA & logits shows the importance of adjusted Gaussian logit guidance. Since Gau WA only relies on the regularization of different Gaussian prediction distributions, the adjusted predictions cannot guide the further classification, so that the improvements are relatively implicit; Gau WA even results in a 0.07% performance drop by comparison with Gau Fusion on the OntoEvent dataset. Gau WA & logits utilizes all of the Gaussian enhanced logits to further improve regularization. The F1-score performance of Gau WA & logits is 0.49% higher than Gau WA on the average of all datasets, while 3.65% higher than basic model DMBERT on average.

6 Conclusion and Future Work

In this paper, we propose GauSE, a Gaussian enhanced self-attention EE mechanism for the first time to better utilize the syntactic related trigger/argument specific knowledge. To explore the latent information among Gaussian distributions, we design several interaction strategies based on existing ones. Concretely, we construct Gaussian fusion methods and regularization methods based on distribution divergence metric. Experiments on several datasets demonstrate that our model achieves significant improvements above the previous state-of-the-art models, especially those based on syntactic knowledge. Further experiments in few-sample scenarios indicate that our model benefits low-resource EE tasks.

We will explore more efficient interaction strategies among Gaussian enhanced information to introduce latent knowledge for future work. And expand our methods to other keyword-based IE tasks, such as relation extraction, to improve the generalization ability of our model.

References

Yubo Chen, Liheng Xu, Kang Liu, Daojian Zeng, and Jun Zhao. 2015. Event extraction via dynamic multi-pooling convolutional neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, pages 167–176. The Association for Computer Linguistics.

Shumin Deng, Ningyu Zhang, Jiaojian Kang, Yichi Zhang, Wei Zhang, and Huajun Chen. 2020. Meta-learning with dynamic-memory-based prototypical network for few-shot event detection. In *WSDM '20: The Thirteenth ACM International Conference on Web Search and Data Mining, Houston, TX, USA, February 3-7, 2020*, pages 151–159. ACM.

Shumin Deng, Ningyu Zhang, Luoqiu Li, Chen Hui, Huaixiao Tou, 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, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 2828–2839. Association for Computational Linguistics.

Shumin Deng, Ningyu Zhang, Wen Zhang, Jiaoyan Chen, Jeff Z. Pan, and Huajun Chen. 2019. Knowledge-driven stock trend prediction and explanation via temporal convolutional network. In *Companion of The 2019 World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019*, pages 678–685. ACM.

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

Xinya Du and Claire Cardie. 2020. Event extraction by answering (almost) natural questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 671–683. Association for Computational Linguistics.

Goran Glavas and Jan Snajder. 2014. Event graphs for information retrieval and multi-document summarization. *Expert Syst. Appl.*, 41(15):6904–6916.

Maosheng Guo, Yu Zhang, and Ting Liu. 2019. Gaussian transformer: A lightweight approach for natural language inference. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 6489–6496. AAAI Press.

Prashant Gupta and Heng Ji. 2009. Predicting unknown time arguments based on cross-event propagation. In *ACL 2009, Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the AFNLP, 2-7 August 2009, Singapore, Short Papers*, pages 369–372. The Association for Computer Linguistics.

Heng Ji and Ralph Grishman. 2008. Refining event extraction through cross-document inference. In *ACL 2008, Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics, June 15-20, 2008, Columbus, Ohio, USA*, pages 254–262. The Association for Computer Linguistics.

Viet Dac Lai, Tuan Ngo Nguyen, and Thien Huu Nguyen. 2020. Event detection: Gate diversity and syntactic importance scores for graph convolution neural networks. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 5405–5411. Association for Computational Linguistics.

Diya Li, Lifu Huang, Heng Ji, and Jiawei Han. 2019. Biomedical event extraction based on knowledge-driven tree-lstm. 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 1421–1430.

Manling Li, Qi Zeng, Ying Lin, Kyunghyun Cho, Heng Ji, Jonathan May, Nathanael Chambers, and Clare R. Voss. 2020. Connecting the dots: Event graph schema induction with path language modeling. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 684–695. Association for Computational Linguistics.

Qi Li, Heng Ji, and Liang Huang. 2013. Joint event extraction via structured prediction with global features. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, 4-9 August 2013, Sofia, Bulgaria, Volume 1: Long Papers*, pages 73–82. The Association for Computer Linguistics.

Xin Liang, Dawei Cheng, Fangzhou Yang, Yifeng Luo, Weining Qian, and Aoying Zhou. 2020. F-hmtc: Detecting financial events for investment decisions based on neural hierarchical multi-label text classification. In *IJCAI*, pages 4490–4496.

Ying Lin, Heng Ji, Fei Huang, and Lingfei Wu. 2020. A joint neural model for information extraction with

global features. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7999–8009. Association for Computational Linguistics.

Xiao Liu, Zhunchen Luo, and Heyan Huang. 2018. Jointly multiple events extraction via attention-based graph information aggregation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 1247–1256. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2017. Fixing weight decay regularization in adam. *CoRR*, abs/1711.05101.

Yaojie Lu, Hongyu Lin, Jin Xu, Xianpei Han, Jialong Tang, Annan Li, Le Sun, Meng Liao, and Shaoyi Chen. 2021. Text2event: Controllable sequence-to-structure generation for end-to-end event extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 2795–2806. Association for Computational Linguistics.

Thien Huu Nguyen, Kyunghyun Cho, and Ralph Grishman. 2016. Joint event extraction via recurrent neural networks. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 300–309. The Association for Computational Linguistics.

Thien Huu Nguyen and Ralph Grishman. 2015. Event detection and domain adaptation with convolutional neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 2: Short Papers*, pages 365–371. The Association for Computer Linguistics.

Thien Huu Nguyen and Ralph Grishman. 2018. Graph convolutional networks with argument-aware pooling for event detection. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 5900–5907. AAAI Press.

Trung Minh Nguyen and Thien Huu Nguyen. 2019. One for all: Neural joint modeling of entities and events. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 6851–6858. AAAI Press.

Lei Sha, Feng Qian, Baobao Chang, and Zhifang Sui. 2018. Jointly extracting event triggers and arguments by dependency-bridge RNN and tensor-based argument interaction. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 5916–5923. AAAI Press.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. 2018. Self-attention with relative position representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 2 (Short Papers)*, pages 464–468. Association for Computational Linguistics.

Matthias Sperber, Jan Niehues, Graham Neubig, Sebastian Stüker, and Alex Waibel. 2018. Self-attentional acoustic models. In *Interspeech 2018, 19th Annual Conference of the International Speech Communication Association, Hyderabad, India, 2-6 September 2018*, pages 3723–3727. ISCA.

Meihan Tong, Bin Xu, Shuai Wang, Yixin Cao, Lei Hou, Juanzi Li, and Jun Xie. 2020. Improving event detection via open-domain trigger knowledge. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 5887–5897. Association for Computational Linguistics.

David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. 2019. Entity, relation, and event extraction with contextualized span representations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 5783–5788. Association for Computational Linguistics.

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

Xiaozhi Wang, Xu Han, Zhiyuan Liu, Maosong Sun, and Peng Li. 2019. Adversarial training for weakly supervised event detection. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 998–1008. Association for Computational Linguistics.

Xing David Wang, Leon Weber, and Ulf Leser. 2020. Biomedical event extraction as multi-turn question answering. In *Proceedings of the 11th International Workshop on Health Text Mining and Information Analysis*, pages 88–96.

Haoran Yan, Xiaolong Jin, Xiangbin Meng, Jiafeng Guo, and Xueqi Cheng. 2019. Event detection with multi-order graph convolution and aggregated attention. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 5765–5769. Association for Computational Linguistics.

Baosong Yang, Zhaopeng Tu, Derek F. Wong, Fandong Meng, Lidia S. Chao, and Tong Zhang. 2018. Modeling localness for self-attention networks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 4449–4458. Association for Computational Linguistics.

Sen Yang, Dawei Feng, Linbo Qiao, Zhigang Kan, and Dongsheng Li. 2019. Exploring pre-trained language models for event extraction and generation. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 5284–5294. Association for Computational Linguistics.

Tongtao Zhang, Heng Ji, and Avirup Sil. 2019. Joint entity and event extraction with generative adversarial imitation learning. *Data Intell.*, 1(2):99–120.

Jie Zhou, Ganqu Cui, Shengding Hu, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. 2020. Graph neural networks: A review of methods and applications. *AI Open*, 1:57–81.