

# Event Detection with Dual Relational Graph Attention Networks

Jiaxin Mi<sup>1,2,3</sup>, Po Hu<sup>1,2,3\*</sup>, Peng Li<sup>4</sup>

<sup>1</sup>Hubei Provincial Key Laboratory of Artificial Intelligence and Smart Learning,  
Central China Normal University, Wuhan, Hubei, China

<sup>2</sup>School of Computer Science, Central China Normal University, Wuhan, Hubei, China

<sup>3</sup>National Language Resources Monitoring & Research Center for Network Media,  
Central China Normal University, Wuhan, Hubei, China

<sup>4</sup>Institute for AI Industry Research (AIR), Tsinghua University, China

{mjx, phu}@mail.ccnu.edu.cn, lipeng@air.tsinghua.edu.cn

## Abstract

Event detection, which aims to identify instances of specific event types from pieces of text, is a fundamental task in information extraction. Most existing approaches leverage syntactic knowledge with a set of syntactic relations to enhance event detection. However, a side effect of these syntactic-based approaches is that they may confuse different syntactic relations and tend to introduce redundant or noisy information, which may lead to performance degradation. To this end, we propose a simple yet effective model named DualGAT (Dual Relational Graph Attention Networks), which exploits the complementary nature of syntactic and semantic relations to alleviate the problem. Specifically, we first construct a dual relational graph that both aggregates syntactic and semantic relations to the key nodes in the graph, so that event-relevant information can be comprehensively captured from multiple perspectives (i.e., syntactic and semantic views). We then adopt augmented relational graph attention networks to encode the graph and optimize its attention weights by introducing contextual information, which further improves the performance of event detection. Extensive experiments conducted on the standard ACE2005 benchmark dataset indicate that our method significantly outperforms the state-of-the-art methods and verifies the superiority of DualGAT over existing syntactic-based methods.

## 1 Introduction

Event Detection (ED) aims to identify event triggers from a given text and classify them into pre-defined event types (Chen et al., 2015), playing an important role in information extraction. As shown in Figure 1, an event detection method is expected to identify the event trigger “thrust” from the example sentence and classify it into the event type *Transport*. Event detection can be used to facilitate various natural language processing

(NLP) applications, such as adverse drug event discovery (Wei et al., 2020; Liu et al., 2018a), court decision event identification (Filtz et al., 2020), financial event extraction (Zheng et al., 2019; Liang et al., 2020) and so on.

Various event detection methods have been explored, including traditional feature-based (Hong et al., 2011; Li et al., 2013) and deep learning methods (Chen et al., 2015; Nguyen et al., 2016). Due to the powerful feature representation extraction capability of deep neural networks, deep learning methods have outperformed traditional feature-based methods in most cases and have become popular. Most deep learning methods have recently paid more attention to exploiting syntactic relations in event detection. These methods adopt Graph Neural Networks (GNNs) such as Graph Convolutional Networks (GCNs) and Graph Attention Networks (GATs) to encode dependency trees to learn effective representations for the words. Since dependency trees convey rich linguistic information for ED, syntactic-based methods usually achieve better performance (Xie et al., 2021).

However, existing syntactic-based methods still have two shortcomings to be solved. First, dependency trees cannot always capture trigger-related salient information concisely, which may contain noisy dependency relations close to the root node and mislead event detection (Lai et al., 2020; Liu et al., 2021). As shown in Figure 1, the dependencies marked in red are incorrectly identified by the syntactic-based event detection methods as trigger-related hints for the *Attack* event. Since “troops” and the trigger candidate “striking” have a direct connection with the root node “distance”, GNN-based methods are prone to pay more attention to them and predict incorrect triggers (Liu et al., 2021). It is worth noting that in this case, the marked blue dependencies that are closely related to the real trigger “thrust” should be exploited with more emphasis. Second, relying solely on syntactic

\* Corresponding author

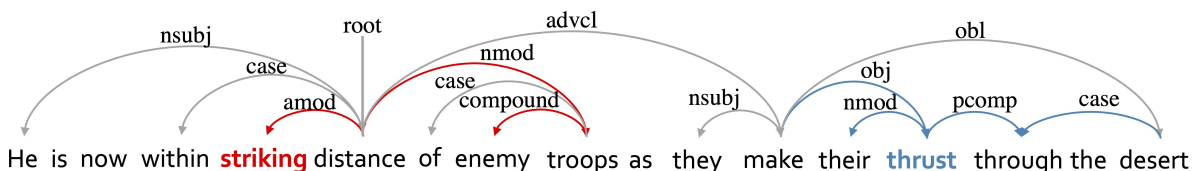


Figure 1: An event detection example with syntactic dependency relations. The “thrust” is the true trigger of Transport event. The “striking” is incorrectly identified as the trigger for an Attack event by some existing methods. The red arrow indicates a noisy dependency relation and blue arrow indicates a trigger-relevant relation.

dependency trees may not be sufficient for event detection. The parsing results of the existing dependency parser may contain incorrect or weakly correlated information, which will inevitably affect the performance of syntactic-based event detection methods due to possible error propagation. Moreover, syntactic dependency trees cannot provide all the linguistic knowledge needed for event detection.

To address the above problems, we propose a novel model named DualGAT, which makes full use of syntactic and semantic information to improve event detection performance. Inspired by aspect-based dependency parsing (Wang et al., 2020a), we construct a dual relational graph that contains both syntactic and semantic relations to capture trigger-relevant information. Empirically, only a small part of dependency relations in a sentence is task-aware (Zhang et al., 2018; He et al., 2018). Therefore, to reduce the influence of noisy relations, we prune the original syntactic dependencies that are not directly connected to the trigger candidates and reconstruct other connections between the remaining words of the sentence and the trigger candidates. In addition to syntactic information, we also introduce semantic relation information and make them rooted in the predicate of the sentence. Next, we adopt augmented relational graph attention networks to encode the graph to learn the root node’s representations from syntactic and semantic views. Specially, we introduced contextual information to adjust the attention weights to mitigate the possible loss of information due to the introduction of the dependency parser. The experimental results on the standard ACE2005 benchmark dataset indicate that DualGAT outperforms the state-of-the-art methods by a large margin.

The main contributions of this work can be summarized as follows:

- We construct a dual relational graph that converges syntactic and semantic relations to the critical nodes in the graph, which can capture

important event information from different perspectives and reduce information loss or noise caused by the syntactic parser.

- We adopt an augmented relational graph attention network to encode the graph and optimize its attention weights by introducing contextual information.
- Experimental results further verify the superiority of DualGAT over existing approaches. DualGAT achieves the 5% improvements in F1 score with state-of-the-art syntactic-based methods.

## 2 Related Work

Event detection has attracted extensive attention in recent years. Traditional event detection methods use hand-crafted linguistic features for event detection, such as lexical features, syntactic features, or entity features (Hong et al., 2011; Ahn, 2006). However, it is time-consuming to design these features and is not easy to adapt to other tasks or new domains.

With the rapid development of neural networks, a series of neural event detection methods have been proposed. Many researchers applied new learning strategies to event detection, such as leveraging the weakly-supervised learning strategy to generate more labeled data to improve the performance of event detection (Zeng et al., 2018; Yang et al., 2018). Wang et al. (2019) applied an adversarial training mechanism to obtain diverse and accurate training data. Lu et al. (2019) proposed a method based on knowledge distillation to achieve better performance on sparsely labeled triggers. Some recently proposed methods introduced extra knowledge to improve event detection via open-domain trigger knowledge (Tong et al., 2020), entity knowledge (Liu et al., 2019) and syntactic dependency relations (Yan et al., 2019).

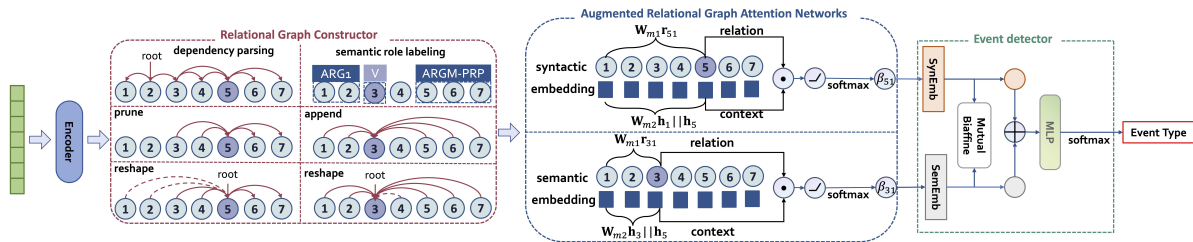


Figure 2: The architecture of our proposed dual relational graph attention networks for event detection. The 5-th node is the assumed trigger candidate and the 3-th is the predicate verb.  $r_{ij}$  denotes the relation embedding of  $e_{ij}$ .

The effectiveness of dependency relations has been verified in many natural language processing (NLP) tasks, such as sentiment analysis (Wang et al., 2020a) and relation extraction (He et al., 2018). Due to the rich syntactic and structure information, syntactic dependency relations also play an important role in event detection (Liu et al., 2018b). For example, Yan et al. (2019) exploited multi-order syntactic relations in sentences to obtain better representations of trigger words. Lv et al. (2021) integrated syntax and document information for better event detection. Cui et al. (2020) proposed a model to explore further the type information of dependency relations to capture task-aware knowledge. Since existing graph-based models introduced many trigger-agnostic representations, Lai et al. (2020) proposed to filter noisy information via a gating mechanism. Due to the effective combination of GNNs and syntactic dependency trees, these syntactic-based event detection methods achieved overall better performance than ordinary deep learning methods.

Although these works use similar syntactic information, few of them take a new perspective to reshape the original graph to facilitate the capture of event-relevant salient information. The original dependency tree contains rich structural and linguistic knowledge, but it may be inaccurate and contains event-irrelevant information. Besides, as far as we know, there are no syntactic-based approaches that explicitly use semantic information to complement syntactic information for event detection. To this end, we propose DualGAT, which takes into account both syntactic and semantic relations as well as contextual information for event detection.

### 3 Methodology

#### 3.1 Model Overview

The overall architecture of DualGAT is shown in Figure 2, which consists of three major compo-

nents: (1) *Relational Graph Constructor* (§3.2), which constructs the dual relational graph by dependency parsing and semantic role labeling; (2) *Augmented Relational Graph Attention Networks* (§3.3), which introduces additional contextual information into the adaptation of attention weight and encodes the dual relational graph to get the root node’s representations from syntactic and semantic views; (3) *Event detector* (§3.4), which leverages the Biaffine module to exchange relevant features between syntactic and semantic representations and performs event detection.

#### 3.2 Relational Graph Constructor

##### 3.2.1 Dual Relational Graph

Existing methods are interfered with by noisy dependency relations and tend to learn trigger-agnostic representations (Lai et al., 2020). Since the trigger candidates are the focus of the event detection task, this noisy information irrelevant to the trigger words inevitably hurts the performance. Besides, the parsing results of the existing dependency parser may contain incorrect or weakly correlated information, which limits the performance of event detection.

Many works have proven that only a small part of the syntactic relations is task-aware (Zhang et al., 2018; He et al., 2018) and identifying trigger words is the core of event detection tasks. Thus, we believe that paying more attention to syntactic relations directly linked to trigger candidates can reduce the effect of erroneous syntactic relations. Besides, the complementary nature of semantic and syntactic knowledge has been exploited and validated in related NLP tasks, such as relations extraction (Bovi et al., 2015) and entity extraction (Chan and Roth, 2011). We believe that semantic relations can make up for the inadequacy of syntactic relations and reduce reliance on dependency parsers.

Motivated by the above idea, we first construct a dual relational graph structure that aggregates

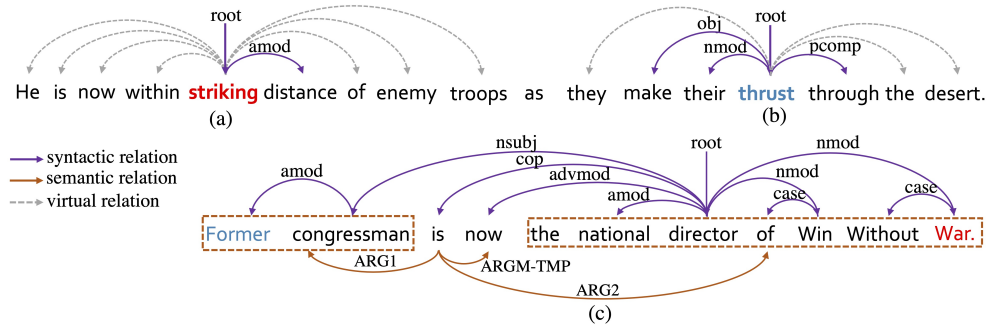


Figure 3: Three event detection examples to illustrate the advantages of the proposed dual relational graph. The correct trigger is marked blue and the incorrect one is marked red. Example (a) and example (b) are two dual relational graphs of one sentence that omit semantic relations. In example (c), the purple arrow is the original syntactic relation, and the orange arrow is the additional semantic relation of the dual relational graph.

---

**Algorithm 1** The Construction Process of Dual Relational Graph.

---

**Input:**

The sentence  $X = \{x_i \mid i \in [1, L]\}$  with a trigger candidate  $x_t$  and a predicate verb  $x_v$ .

**Output:**

Dual relational graph  $G$ ;

- 1: Getting syntactic, semantic edge set  $D, A$ .
  - 2: Construct initial dual relational graph  $G$  with  $L$  nodes.
  - 3: **for**  $i = 1$  to  $L$  **do**
  - 4:   **if**  $e_{it}$  or  $e_{ti}$  in  $D$  **then**
  - 5:     add  $e_{ti}$  in  $G$ .
  - 6:   **else**
  - 7:      $d = t - i + 1, e_{ti} = vir : d$ .
  - 8:     add  $e_{ti}$  in  $G$
  - 9:   **end if**
  - 10:   **if**  $e_{iv}$  or  $e_{vi}$  in  $A$  **then**
  - 11:     add  $e_{vi}$  in  $G$ .
  - 12:   **else**
  - 13:      $d = v - i + 1, e_{vi} = vir : d$ .
  - 14:     add  $e_{vi}$  in  $G$
  - 15:   **end if**
  - 16: **end for**
  - 17: **return**  $G$ ;
- 

syntactic relations to trigger candidates and additionally aggregates semantic relations to predicate verbs to improve the robustness of the graph. Algorithm 1 describes the construction process of the dual relational graph. We first leverage a dependency parser to obtain the original syntactic relations of a given sentence. Then we retain the syntactic relations directly connected to the trigger candidates and prune the remaining relations. To improve the robustness of the graph, we further

replace the pruned relations with a virtual one classified by distance with trigger candidates. Specifically, the type of virtual relations is defined as  $vir : d$ , where  $d$  represents the distance between a word and the trigger candidate. Finally, we perform semantic role labeling to append semantic relations between other words and predicate verbs and do the same process as above.

The graph obtained after the above processing is the dual relational graph. It has two same levels of subgraphs: the syntactic relational subgraph, which converges syntactic information to the trigger candidates, and the semantic relational subgraph, which converges semantic information to the predicate verbs. Formally, we define the dual relational graph as  $G = (V, E)$  associated with a edge type mapping function  $\tau : E \rightarrow T_E$ .  $V$  represents the set of word nodes in a sentence, and  $E$  represents the set of relational edges.  $T_E$  represents all types of relations including syntactic relations, semantic relations and virtual relations. The relational edge between word node  $i$  and word node  $j$  is defined as  $e_{ij}$ .

**3.2.2 The Advantages of Dual Relational Graph**

The dual relational graph has proprietary advantages in event detection. First, the dual relational graph is customized for every trigger candidate, which can reduce the introduction of noisy trigger-agnostic information. Second, the semantic relations rooted in predicate verbs can improve the robustness of the dual relational graph. Third, the dual relational graph structure facilitates the event detection model to capture task-aware information. We use three simple examples to illustrate the above advantages, as shown in Figure 3.



First, example (a) and example (b) are two customized dual relational graphs for different trigger candidates in the same sentence. The existing methods tend to identify “striking” as a trigger evoking an `Attack` event, since the strong bonding between “striking”, “enemy”, and “troops” in the original dependency tree. However, the dual relational graph of “striking” prunes the trigger-agnostic relations and clearly illustrates that “striking” is only an adjectival modifier (amod) of “distance”. Moreover, the dual relational graph of “thrust” can facilitate ED models to capture its connections with “through” and “make” without the interference of noisy dependency relations in the original dependency tree. Thus, ED methods tend to identify “thrust” as a trigger evoking a `Transport` event rather than “striking”.

Besides, example (c) can illustrate the importance of semantic relations. In example (c), the purple arrow is the original syntactic relation, and the orange arrow is the additional semantic relation of the dual relational graph. Based on original syntactic relations, ED methods tend to identify “War” as a trigger of an `Attack` event, since the strong bonding between “War” and “Win”. Even pruning the trigger-agnostic syntactic relations, the correct trigger “Former” still cannot obtain much attention. However, the semantic relations can make the deep semantic information flow from “now” to “Former”. The “ARGM-TMP” relation introduces timing information to help identify “Former” as a trigger for an `End-position` event.

Finally, as shown in examples (a) and (b), the dual relational graph does not have complex mutual interactions. The structure of the dual relational graph is clear and aggregated. It can reduce the interference of noisy interactions and the ED model’s difficulty in capturing trigger-relevant information.

### 3.3 Augmented Relational Graph Attention Networks

To more effectively encode the dual relational graph for event detection, we propose an augmented relational graph attention network (AR-GAT) by introducing additional contextual information to encode graphs constructed from words in a sentence.

#### 3.3.1 Graph Attention Network

Graph neural networks (Scarselli et al., 2009) have been widely used to encode dependency trees for event detection, as they can effectively capture

relevant information based on an information aggregation scheme (Cao et al., 2021). In addition, much work has shown that graph convolutional networks (Schlichtkrull et al., 2018) cannot effectively leverage multi-hop relational information (Yan et al., 2019). Intuitively, the heart of the event detection task is to capture the relevant words with the trigger candidates. Thus, we apply graph attention networks (Velickovic et al., 2018) which can more efficiently leverage the relations between words to encode the dual relational graph.

Formally, given a dual relational graph  $G$  with  $L$  word nodes, and the set of neighborhood nodes of node  $i$  is defined as  $\mathcal{N}_i$ . The feature vector of node  $i$  at layer  $l$  is denoted as  $\mathbf{h}_i^l$  ( $\mathbf{h}_i^l \in \mathbb{R}^F$ ),  $F$  is the dim of node features. For the node  $i$  at the layer  $l+1$ , the computation of multi-head graph attention networks can be defined as follows:

$$\mathbf{h}_{att_i}^{l+1} = \parallel_{k=1}^K \sigma \left( \sum_{j \in \mathcal{N}_i} \alpha_{ij}^{lk} \mathbf{W}_k^l \mathbf{h}_j^l \right) \quad (1)$$

$$\alpha_{ij}^{lk} = \frac{\exp(f_1(\mathbf{a}^T [\mathbf{W}_k^l \mathbf{h}_i^l \parallel \mathbf{W}_k^l \mathbf{h}_j^l]))}{\sum_{t \in \mathcal{N}_i} \exp(f_1(\mathbf{a}^T [\mathbf{W}_k^l \mathbf{h}_i^l \parallel \mathbf{W}_k^l \mathbf{h}_t^l]))} \quad (2)$$

where  $\mathbf{h}_{att_i}^{l+1}$  is the attention head of node  $i$  at the layer  $l+1$ ,  $\mathbf{W}_k^l$  is a transformation matrix,  $f_1(\cdot)$  is the function of LeakyReLU,  $\mathbf{a}$  is a weight vector,  $K$  is the number of heads and  $\parallel_{k=1}^K \mathbf{h}_k$  represents concatenation of vectors from  $\mathbf{h}_1$  to  $\mathbf{h}_K$ .

#### 3.3.2 AR-GAT

The relational graph attention networks (Wang et al., 2020a) extended original graph attention networks with additional heads to leverage the type information of edges. However, relational graph attention networks are not sufficiently compatible with encoding the dual relational graph. On the one hand, original syntactic and semantic relations that are initially generated may be wrong. On the other hand, in the construction process of the dual relational graph, reshaping and pruning may further lead to the propagation of errors originating from the parser. Thus, the relation heads are not sufficient to accurately control information flow from neighborhood nodes.

To overcome the above problems, we propose introducing additional contextual information from word nodes to control information flow more accurately. The performance of factorization machines has been proven in many tasks (Guo et al., 2017). Inspired by factorization machines, we employ an

inner product unit to combine contextual and type information.

Formally, the initial relation embedding matrix is defined as  $\mathbf{W}_t \in \mathbb{R}^{N_t \times F}$ ,  $N_t$  is the number of relation type. For the node  $i$  at the layer  $l+1$ , the computation of multi-head augmented relational graph attention networks can be defined as follows:

$$\mathbf{h}_{rel_i}^{l+1} = \parallel_{m=1}^M \sigma \left( \sum_{j \in \mathcal{N}_i} \beta_{ij}^{lm} \mathbf{W}_m^l \mathbf{h}_j^l \right) \quad (3)$$

$$R_{ij}^m = (\mathbf{W}_{m1} f_2(\tau(e_{ij}), \mathbf{W}_t) + b_{m1}) \odot (\mathbf{W}_{m2}(\mathbf{h}_i \parallel \mathbf{h}_j) + b_{m2}) \quad (4)$$

$$g_{ij}^{lm} = \sigma(\text{relu}(R_{ij}^m \mathbf{W}_{m3} + b_{m3}) \mathbf{W}_{m4} + b_{m4}) \quad (5)$$

$$\beta_{ij}^{lm} = \frac{\exp(g_{ij}^{lm})}{\sum_{j \in \mathcal{N}_i} \exp(g_{ij}^{lm})} \quad (6)$$

where  $\mathbf{h}_{rel_i}^{l+1}$  represents the augmented relational attention head of node  $i$  at layer  $l+1$ ,  $M$  is the number of heads,  $f_2(\tau(e_{ij}), \mathbf{W}_t)$  is a mapping function mapping edge  $e_{ij}$  into the corresponding relation embedding according to relation embedding matrix  $\mathbf{W}_t$ . The final representation of node  $i$  is computed by:

$$o_i^{l+1} = \mathbf{h}_{att_i}^{l+1} \parallel \mathbf{h}_{rel_i}^{l+1} \quad (7)$$

$$\mathbf{h}_i^{l+1} = \text{relu}(\mathbf{W}_{l+1} o_i^{l+1} + b_{l+1}) \quad (8)$$

### 3.4 Event Detector

We use BERT (Devlin et al., 2019) to obtain the word embedding of graph nodes, and the embedding of word  $x_i$  is defined as  $\mathbf{h}_i^0$ . We use  $\mathbf{h}_t^0$  and  $\mathbf{h}_v^0$  to denote the initial embedding of the trigger candidate node and the predicate verb node, respectively. Then we apply AR-GAT to encode two subgraphs of dual relational graph respectively and obtain their root representation  $\mathbf{h}_t^l$  and  $\mathbf{h}_v^l$ . The  $\mathbf{h}_t^l$  and  $\mathbf{h}_v^l$  aggregate the syntactic and semantic information respectively.

To effectively exchange relevant features between these two types of information, we employ a mutual Biaffine transformation (Li et al., 2021b). The interaction process is:

$$\mathbf{h}_t^l = \text{softmax}(\mathbf{h}_t^l \mathbf{W}_1 (\mathbf{h}_v^l)^T) \mathbf{h}_v^l \quad (9)$$

$$\mathbf{h}_v^l = \text{softmax}(\mathbf{h}_v^l \mathbf{W}_2 (\mathbf{h}_t^l)^T) \mathbf{h}_t^l \quad (10)$$

where  $\mathbf{W}_1$  and  $\mathbf{W}_2$  are parameters.

Finally, we concatenate syntactic root  $\mathbf{h}_t^l$  and semantic root  $\mathbf{h}_v^l$  to obtain final feature representation:

$$\mathbf{x} = \mathbf{h}_t^l \parallel \mathbf{h}_v^l \quad (11)$$

Then the final feature representation  $\mathbf{x}$  is fed into a fully connected layer and adopt softmax layer to get a final event type probability distribution:

$$p(t) = \text{softmax}(\mathbf{W}_p \mathbf{x} + b_p) \quad (12)$$

## 4 Experiments

### 4.1 Datasets and Evaluation Metrics

We evaluate our proposed model on the widely used standard benchmark dataset ACE2005. ACE2005 consists of 33 event types and contains 599 documents with 4090 event instances. Following the previous works (Chen et al., 2015; Wang et al., 2020b), we use the same data split for train, dev and test set. We adopt Precision (P), Recall (R) and micro F1 score (F1), which are the standard metrics for event detection, as the evaluation criteria for method performance. The three metrics are defined as follows:

$$P = \frac{N_C}{N_P} \quad (13)$$

$$R = \frac{N_C}{N_D} \quad (14)$$

$$F1 = \frac{2PR}{P+R} \quad (15)$$

where  $N_C$ ,  $N_P$ , and  $N_D$  are the number of correctly predicted events, all predicted events, and all events in the dataset, respectively.

### 4.2 Parameter Settings

Our implementation <sup>1</sup> is based on the bert-base-uncased model, whose layer number is 12, hidden size is 768, and attention head number is 12. For the ‘‘Our w/o BERT’’ setting, we use Glove (Pennington et al., 2014) and BiLSTM to obtain word embedding. The dimension of both word embedding and relational embedding is 300. the hidden state size of AR-GAT and BiLSTM is set to 200. The number of attention heads is 6. The max sentence length is set to 128 by cutting longer sentences and padding shorter ones. We employ the Biaffine Parser (Dozat and Manning, 2017) and SRL BERT (Shi and Lin, 2019) for dependency

<sup>1</sup>Code is available at <https://github.com/Macvh/DualGAT>, which leverage the HuggingFace’s Transformers library for loading pre-trained models.

parsing and semantic role labeling. The Adam algorithm (Kingma and Ba, 2015) is used to optimize model parameters and the dropout rate is set to 0.3 to avoid over-fitting. The max epoch is 30. In addition, we report the average performance of 5 trials with different random seeds for each experiment.

### 4.3 Baselines

We select various representative methods as baselines, including:

**Syntactic based Methods:** (1) **GCN-ED** uses GCN with entity mention based pooling method for event detection (Nguyen and Grishman, 2018). (2) **SA-GRCN** introduces a self-attention mechanism for better modeling word dependencies (Liu et al., 2021). (3) **EE-GCN** introduces additional typed dependency label information into GCNs (Cui et al., 2020). (4) **GatedGCN** proposes to filter noisy information via a gating mechanism (Lai et al., 2020).

**External Knowledge based Methods:** (1) **PLMEE** proposes a method to enlarge the scale of labeled data by editing prototypes and an evaluation method to screen out generated data (Yang et al., 2019). (2) **DNR** uses additional external knowledge to link predefined event types to each sentence to improve performance (Liao et al., 2021). (3) **SS-JDN** introduces statistical features to cooperate with the contextual features for event detection (Li et al., 2021a).

### 4.4 Overall Results

Experimental results are shown in Table 1, where “Our w/o BERT” denotes replacing BERT with BiLSTM in our model. The highest values are shown in bold. We can observe that our proposed DualGAT outperforms all the baselines on the ACE2005 dataset in terms of three metrics from Table 1. Particularly, DualGAT outperforms the syntactic-based methods by a large margin. It is worth mentioning that the performance of Our w/o BERT is even better than that of GatedGCN which used BERT. Compared with the next best method among all compared methods, DualGAT improves the F1, Precision, and Recall by 0.9%, 0.5%, and 1.3%, respectively. It proves the effectiveness of our proposed model for ED tasks. DualGAT fully considers the complementary of syntactic and semantic information and effectively captures the trigger-relevant information. Compared with existing syntactic-based methods, the advantages of DualGAT are more apparent. DualGAT is able to more effectively leverage the internal critical

Type	Method	P	R	F1	BERT
Syn.	GCN-ED	77.9	68.8	73.1	-
	EE-GCN	76.7	78.6	77.6	-
	SA-GRCN	78.6	77.4	78.0	-
	GatedGCN	78.8	76.3	77.6	✓
Exter.	SS-JDN	80.3	78.8	79.5	✓
	PLMEE	81.0	80.4	80.7	✓
	DNR	81.2	82.4	81.8	✓
Ours	Our w/o BERT	79.1	80.8	80.0	-
	DualGAT	<b>81.7</b>	<b>83.7</b>	<b>82.7</b>	✓

Table 1: Overall performance on ACE2005 dataset (%). “Syn.” indicates syntactic dependency relations are used, “Exter.” indicates external knowledge and resources are used.

Method	P	R	F1
DualGAT w/o reshape	78.1	75.3	76.7
DualGAT w/o syntactic	76.2	79.3	77.7
DualGAT w/o semantic	78.9	81.3	80.1
DualGAT w/o AR-GAT	80.3	82.1	81.2
DualGAT	<b>81.7</b>	<b>83.7</b>	<b>82.7</b>

Table 2: Experimental results of ablation study on ACE2005 dataset (%).

information of sentences for ED tasks. Besides, DualGAT is even better than three methods that use external knowledge. It indicates that existing ED methods do not fully exploit the internal information of sentences of original data that are worth further exploiting.

### 4.5 Ablation Study

To assess the effect of the dual relational graph and augmented relational graph attention networks, we further conduct several ablation studies. We design four variants of the proposed model:

**DualGAT w/o reshape:** to study whether the dual relational graph contributes to the performance improvement, we use the ordinary syntactic dependency tree to replace the dual relational graph.

**DualGAT w/o syntactic:** to prove the effectiveness of the syntactic relational subgraph, we remove the syntactic relational subgraph and only use the semantic relational subgraph.

**DualGAT w/o semantic:** to prove the effectiveness of the semantic relational subgraph, we remove the semantic relational subgraph and only use the syntactic relational subgraph.

**DualGAT w/o AR-GAT:** to prove the effectiveness of the augmented relational graph attention

networks, we use relational graph attention networks to encode the dual relational graph.

The results of the ablation study are shown in Table 2. From the results, we can observe that:

(1) DualGAT w/o semantic outperforms DualGAT w/o reshape by a large margin. It indicates that pruning and reshaping original syntactic relations are useful for event detection. The syntactic relational subgraph converges all trigger-relevant relations to the trigger candidates, which reduces the interference of noisy relations. It also indicates that a small part of dependency is task-aware, and encoding the entire dependency tree is unnecessary for event detection.

(2) In terms of performance degradation compared to DualGAT, the F1 score of DualGAT without syntactic relational subgraph drops more seriously than that without semantic relational subgraph. It indicates that syntactic relations converged on trigger candidates are more necessary in DualGAT. The reason may be that the syntactic relational subgraph establishes more effective correlations among trigger candidates and other words in a sentence.

(3) The DualGAT is improved by using the augmented relational graph attention networks and achieves 1.5% improvements in the F1 score. It indicates that the introduction of contextual information effectively captures key information between words. The dependency parsers do not always parse sentences correctly, which causes much loss of information. Thus, we use additional contextual information to get more accurate attention weights.

#### 4.6 The Effect of Multiple Event Recognition

To verify the effectiveness of the dual relational graph customized for every trigger candidate, we study the performance of the proposed DualGAT for multiple event recognition. Following (Xie et al., 2021), we divide the test set into the “1/1” and “1/N” subsets and perform evaluation on the two subsets separately, where one sentence has only one event in the 1/1 subset but has multiple events in the 1/N subset. Moreover, one sentence may contain multiple event types in the 1/N subset. Figure 4 illustrates the performance (F1 score) of DualGAT w/o reshape and DualGAT.

As shown in Figure 4, our proposed DualGAT significantly outperforms DualGAT w/o reshape in three situations. DualGAT improves upon the DualGAT w/o reshape by 6.1% and 9.9% in 1/1 data

split and 1/N data split, respectively. The multi-event scenario in a sentence confuses the event detection methods, resulting in poor performance. However, DualGAT improves the performance by using the dual relational graph. Since each trigger candidate has its particular dual relational graph, the dual relational graph can reduce the interference of irrelevant nodes and relations. The additional semantic relations further capture sentence-level information. Thus, our proposed DualGAT can alleviate the multi-event problem to a certain extent. The experimental results demonstrate that the dual relational graph is effective for the task of multiple event recognition.

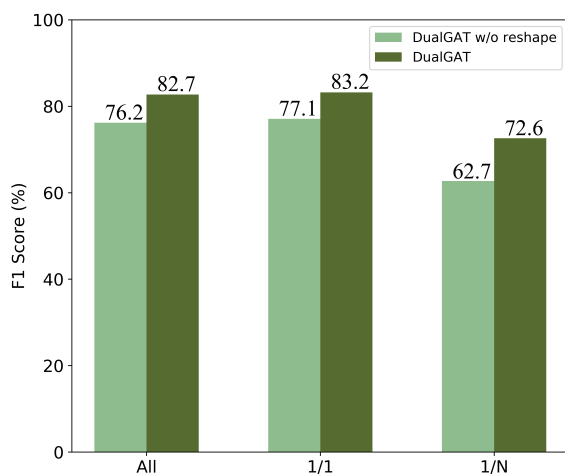


Figure 4: Performance of our method with ordinary syntactic dependency tree (“DualGAT w/o reshape”) and the proposed dual relational graph (“DualGAT”).

#### 4.7 Semantic Relations Improve Robustness

We further study whether semantic relations can make the dual relational graphs less vulnerable to dependency parsing errors. Therefore, we conduct experiments based on two widely used dependency parsers: Stanford Parser (Chen and Manning, 2014) and Biaffine Parser (Dozat and Manning, 2017). The performance of the two parsers is shown in Table 3, measured by two widely used metrics UAS and LAS, of which higher is better. We use each of these two dependency parsers to construct the dual relational graph and evaluate the final performance of the proposed method.

The experimental results of DualGAT with different dependency parsers are shown in Table 4. From Table 4, we can find that the DualGAT with Biaffine parser achieves better performance in event detection since the dependency parsing performance of the Biaffine parser is better than Stanford Parser.



Parser	UAS	LAS
Stanford	94.10	91.49
Biaffine	<b>95.74</b>	<b>94.08</b>

Table 3: The performance of two dependency parsers. The results are from (Chen and Manning, 2014) and (Dozat and Manning, 2017) respectively.

Method	P	R	F1
DualGAT (Sta) w/o sem	76.1	78.6	77.3
DualGAT (Bia) w/o sem	78.9	81.3	80.1
DualGAT (Sta)	81.5	82.3	81.9
DualGAT (Bia)	<b>81.7</b>	<b>83.7</b>	<b>82.7</b>

Table 4: The performance of DualGAT using different parsers on ACE2005 dataset (%).

In the case of only syntactic relations, DualGAT with Biaffine parser improves upon the DualGAT with Stanford parser by 2.8%, 2.8% and 2.7% in terms of F1 score, Precision and Recall. However, compared with DualGAT (Bia), DualGAT (Sta) declines by only 0.8%, 0.2% and 1.4% in F1 score, Precision and Recall. It illustrates that semantic relations can resist interference with syntactic parser performance and improve the robustness of the dual relational graph. Semantic relations have complementarity with syntactic relations and sustain the performance of DualGAT in the case of syntactic relationship failure. Besides, it also implies that our proposed DualGAT can benefit from the advances in syntactic parsing and semantic parsing techniques.

## 5 Conclusion and Future Work

In this paper, we propose a simple yet effective model named DualGAT (Dual Relational Graph Attention Networks) to address the disadvantages of syntactic-based methods for event detection tasks. To facilitate the accurate capture of key information from different perspectives in a sentence, we devise a dual relational graph that aggregates syntactic and semantic relations to key nodes in the graph. To efficiently encode the dual relational graph, we propose augmented relational graph attention networks that introduce contextual information to compute more robust attention weights. Experimental results show that our proposed method achieves state-of-the-art performance.

We intend to explore several aspects of our work further in the future. First, we would improve the way semantic information is introduced. Second,

we would develop a more effective method for fusing syntactic and semantic information. Third, we would explore the effect of augmented relational graph attention networks in other tasks.

## Acknowledgements

We want to express gratitude to the anonymous reviewers for their hard work and kind comments. This work was supported by the National Social Science Fund of China under Grant No. 20BTQ068.

## References

- David Ahn. 2006. The stages of event extraction. In *Proceedings of the Workshop on Annotating and Reasoning about Time and Events*, pages 1–8.
- Claudio Delli Bovi, Luca Telesca, and Roberto Navigli. 2015. Large-scale information extraction from textual definitions through deep syntactic and semantic analysis. *Transactions of the Association for Computational Linguistics*, 3:529–543.
- Pengfei Cao, Xinyu Zuo, Yubo Chen, Kang Liu, Jun Zhao, Yuguang Chen, and Weihua Peng. 2021. Knowledge-enriched event causality identification via latent structure induction networks. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pages 4862–4872.
- Yee Seng Chan and Dan Roth. 2011. Exploiting syntactico-semantic structures for relation extraction. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 551–560.
- Danqi Chen and Christopher D. Manning. 2014. A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 740–750.
- 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*, pages 167–176.
- Shiyao Cui, Bowen Yu, Tingwen Liu, Zhenyu Zhang, Xuebin Wang, and Jinqiao Shi. 2020. Edge-enhanced graph convolution networks for event detection with syntactic relation. In *Findings of the Association for Computational Linguistics*, pages 2329–2339.

- 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*, pages 4171–4186.
- Timothy Dozat and Christopher D. Manning. 2017. Deep biaffine attention for neural dependency parsing. In *Proceedings of the 5th International Conference on Learning Representations*, pages 1–8.
- Erwin Filtz, María Navas-Loro, Cristiana Santos, Axel Polleres, and Sabrina Kirrane. 2020. Events matter: Extraction of events from court decisions. In *Proceedings of Legal Knowledge and Information Systems*, pages 33–42.
- Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. Deepfm: A factorization-machine based neural network for CTR prediction. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*, pages 1725–1731.
- Shexia He, Zuchao Li, Hai Zhao, and Hongxiao Bai. 2018. Syntax for semantic role labeling, to be, or not to be. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 2061–2071.
- Yu Hong, Jianfeng Zhang, Bin Ma, Jian-Min Yao, Guodong Zhou, and Qiaoming Zhu. 2011. Using cross-entity inference to improve event extraction. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1127–1136.
- Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations*, pages 1–15.
- 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*, pages 5405–5411.
- 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*, pages 73–82.
- Rui Li, Wenlin Zhao, Cheng Yang, and Sen Su. 2021a. Treasures outside contexts: Improving event detection via global statistics. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2625–2635.
- Ruifan Li, Hao Chen, Fangxiang Feng, Zhanyu Ma, Xiaojie Wang, and Eduard H. Hovy. 2021b. Dual graph convolutional networks for aspect-based sentiment analysis. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pages 6319–6329.
- 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 *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, pages 4490–4496.
- Jinshi Liao, Xiang Zhao, Xinyi Li, Lingling Zhang, and Jiuyang Tang. 2021. Learning discriminative neural representations for event detection. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 644–653.
- Anan Liu, Ning Xu, and Haozhe Liu. 2021. Self-attention graph residual convolutional networks for event detection with dependency relations. In *Findings of the Association for Computational Linguistics*, pages 302–311.
- Jian Liu, Yubo Chen, and Kang Liu. 2019. Exploiting the ground-truth: An adversarial imitation based knowledge distillation approach for event detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 6754–6761.
- Jing Liu, Songzheng Zhao, and Gang Wang. 2018a. Ssel-ade: a semi-supervised ensemble learning framework for extracting adverse drug events from social media. *Artificial intelligence in medicine*, 84:34–49.
- Xiao Liu, Zhunchen Luo, and Heyan Huang. 2018b. Jointly multiple events extraction via attention-based graph information aggregation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1247–1256.
- Yaojie Lu, Hongyu Lin, Xianpei Han, and Le Sun. 2019. Distilling discrimination and generalization knowledge for event detection via delta-representation learning. In *Proceedings of the 57th Conference of the Association for Computational Linguistics*, pages 4366–4376.
- Jianwei Lv, Zequn Zhang, Li Jin, Shuchao Li, Xiaoyu Li, Guangluan Xu, and Xian Sun. 2021. HGEED: hierarchical graph enhanced event detection. *Neurocomputing*, 453:141–150.
- Thien Huu Nguyen, Kyunghyun Cho, and Ralph Grishman. 2016. Joint event extraction via recurrent neural networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 300–309.
- 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*, pages 5900–5907.

- Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1532–1543.
- Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. 2009. The graph neural network model. *IEEE transactions on neural networks*, 20(1):61–80.
- Michael Sejr Schlichtkrull, Thomas N. Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *Proceedings of the 15th European Semantic Web Conference*, pages 593–607.
- Peng Shi and Jimmy Lin. 2019. Simple bert models for relation extraction and semantic role labeling. *ArXiv*, abs/1904.05255.
- 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*, pages 5887–5897.
- Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph attention networks. In *Proceedings of the 6th International Conference on Learning Representations*.
- Kai Wang, Weizhou Shen, Yunyi Yang, Xiaojun Quan, and Rui Wang. 2020a. Relational graph attention network for aspect-based sentiment analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3229–3238.
- 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*, pages 998–1008.
- Xiaozhi Wang, Ziqi Wang, Xu Han, Wangyi Jiang, Rong Han, Zhiyuan Liu, Juanzi Li, Peng Li, Yankai Lin, and Jie Zhou. 2020b. MAVEN: A massive general domain event detection dataset. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 1652–1671.
- Qiang Wei, Zongcheng Ji, Zhiheng Li, Jingcheng Du, Jingqi Wang, Jun Xu, Yang Xiang, Firat Tiryaki, Stephen Wu, Yaoyun Zhang, Cui Tao, and Hua Xu. 2020. A study of deep learning approaches for medication and adverse drug event extraction from clinical text. *Journal of the American Medical Informatics Association*, 27(1):13–21.
- Jianye Xie, Haotong Sun, Junsheng Zhou, Weiguang Qu, and Xinyu Dai. 2021. Event detection as graph parsing. In *Findings of the Association for Computational Linguistics*, pages 1630–1640.
- 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*, pages 5765–5769.
- Hang Yang, Yubo Chen, Kang Liu, Yang Xiao, and Jun Zhao. 2018. DCFEE: A document-level chinese financial event extraction system based on automatically labeled training data. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics-System Demonstrations*, pages 50–55.
- 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*, pages 5284–5294.
- Ying Zeng, Yansong Feng, Rong Ma, Zheng Wang, Rui Yan, Chongde Shi, and Dongyan Zhao. 2018. Scale up event extraction learning via automatic training data generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 6045–6052.
- Yuhao Zhang, Peng Qi, and Christopher D. Manning. 2018. Graph convolution over pruned dependency trees improves relation extraction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2205–2215.
- Shun Zheng, Wei Cao, Wei Xu, and Jiang Bian. 2019. Doc2edag: An end-to-end document-level framework for chinese financial event extraction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 337–346.