Biomedical Knowledge Derivation from Scientific Publications via Dual-Graph Resonance

Anonymous ACL submission

Abstract

Scientific Information Extraction (SciIE) is a vital task and is increasingly being adopted in biomedical data mining to conceptualize and epitomize knowledge triplets from scientific literature. Existing relation extraction methods aim to extract explicit triplet knowledge from documents, however, they can hardly perceive unobserved factual relations. Recent generative methods have more flexibility, but their generated relations will encounter trustworthiness problems. In this paper, we first propose a novel **E**xtraction-**C**ontextualization-**D**erivation (**ECD**) strategy to generate a document-specific and entity-expanded dynamic graph from a shared static knowledge graph. Then, we propose a novel **D**ual-**G**raph **R**esonance **N**etwork (**DGRN**) which can generate richer explicit and implicit relations under the guidance of static and dynamic knowledge topologies. Experiments conducted on a public PubMed corpus validate the superiority of our method against several state-of-the-art baselines[1].

1 Introduction

As biomedical literature grows at an expeditious pace, it becomes increasingly labor-intensive for scholars to curate the massive information and consume their interested knowledge. For example, PubMed, as one of the most commonly used searching biomedical publication databases, contains more than 34 million publications[2]. Even with a focused research interest, such as oncology, it's still very laborious to locate useful information from noisy retrieval results in terms of domain knowledge. To address this challenge, SciIE can be employed to extract structural information (e.g., triplet) from scientific articles, which has drawn great attention from Natural Language Processing (NLP) community (Gupta and Manning, 2011; Viswanathan et al., 2021).

[1]All the resources will be publicly available.

[2]https://pubmed.ncbi.nlm.nih.gov/about/

Biomedical document: The skin is … and causes cancerous process, … that aggravates … skin cancer … .

Method: Extractive method
Triples: <skin, cancerous, skin cancer> ✓

Method: Generative method
Triples: <skin, cancerous, skin cancer> ✓
<skin, cancerous, throat cancer> ✗

Method: Knowledge Graph (KG) method
Triples: <melanoma, belongs to, skin cancer> ✓

Method: Generative method + KG
Triples: <skin, cancerous, skin cancer> ✓
<skin, cancerous, melanoma> ✓

Figure 1: Comparison among different methods. Entities and relations observed in the document are denoted by red color, and unobserved ones are denoted by blue. Correct/wrong triplets are labeled as check/cross marks.

Recently, efforts have been made regarding SciIE tasks and obtained substantial achievements. Lee et al. (2020) and Beltagy et al. (2019), for instance, trained pre-trained language models with a domain-specific corpus for representation learning and downstream task fine-tuning. Ye et al. (2020) proposed a Contrastive triplet extraction with Generative Transformer (CGT) model which adopted transformers as encoder and decoder within a contrastive learning framework. Despite some success, they ignore that entities may not appear in the same sentence and have long-distance dependencies. To explore relation reasoning, Zeng et al. (2020b) proposed a Graph Aggregation and Inference Network (GAIN), which resorted to a mention-level graph constructed from each document to leverage latent logical reasoning paths and predict the relations among entities. However, with the increasing demands for biomedical knowledge acquisition, existing extractive methodologies can hardly derivate latent and factual knowledge unobserved in the input document.

To enable latent relation extraction, generative models, rather than extractive approaches, need to be adopted. Unfortunately, without pragmatical re-

strictions, existing generative methods can produce unrelated/incorrect triples and result in trustworthiness concerns (Zhang et al., 2020). Take Figure 1 as an example, the generative method produces an incorrect triplet <*skin, cancerous, throat cancer*> because there exists no direct relation between entities *skin* and *throat cancer*, while *throat cancer* is generated based on the trigger word "*cancerous*". To address this problem, biomedical knowledge graphs emerge, which can provide prior domain knowledge guidance (e.g., <*melanoma, belongs to, skin cancer*> in this context) to enhance generative methods to produce reliable and often unobservable triplets (e.g., <*skin, cancerous, melanoma*>). While high-quality and large-scale biomedical knowledge graphs (BKG) have been studied extensively (Wang et al., 2021), one may hesitate to adopt BKG directly due to the potential of massive noise. In this study, we propose a novel *dual-graph resonance network* to address this problem - using large-scale BKG as the *Static Graph* to encapsulate the biomedical domain knowledge and the derived knowledge sub-BKG as *Dynamic Graph* to characterize document-related knowledge. The interactions between static and dynamic graphs ensure the comprehensiveness and trustworthiness of the knowledge generation.

Prior SciIE studies rarely equip generative methods with domain knowledge in an end-to-end fashion, and such investigation can be especially critical for biomedical content mining which poses several challenges for us. Firstly, the input document and knowledge graph are information complementary and should be fully interactively modeled. Secondly, multi-hop relations on graphs should be used for deriving unobserved triplets. To this end, we propose a novel Dual-Graph Resonance Network (DGRN) with a threefold contribution:

- We propose an "Extraction-Contextualization-Derivation (ECD)" strategy, which can derive a document-related dynamic graph from a shared static graph via a dynamic graph generator.
- An innovative end-to-end "Dual-Graph Resonance Network (DGRN)" model is further proposed to generate both observed and unobserved knowledge triplets by jointly modeling the input document and the dual graphs.
- Extensive experiments conducted on a PubMed corpus validate the effectiveness of our method. All the resources will be publicly available, which can facilitate an in-depth study of the SciIE task.

2 Related Work

Existing SciIE works extract or generate knowledge triplets from different parts of scientific publications, such as content (Luan et al., 2019; Augenstein et al., 2017), abstract, introduction and citation sentences (Nakov et al., 2004). The mainstream approaches of the SciIE task generally include extractive, graph and generative models.

Extractive models have been extensively studied. Wei et al. (2020); Beltagy et al. (2019) integrated pretrained language models, like BERT (Devlin et al., 2018), into an encoder-decoder framework for performance improvements. Nayak and Ng (2020); Zhao et al. (2021) introduced a joint learning framework to model connections between relations and their corresponding entity pairs. Furthermore, Takanobu et al. (2019); Bai and Zhao (2018) utilized a hierarchical structure which featured connections among different content layers to perform the SciIE task. However, these methods ignore fine-grained entity-level information integration and interaction, and they can not find unobserved knowledge triplets and observed triplets with long-range relations in the document.

Graph-based models can provide both dependencies among entities and path reasoning potential for inference. Zeng et al. (2020b) proposed a mention-to-entity graph aggregation model which can capture the relation of entities across sentences. Instead of integrating graph structure into neural network models, Peng et al. (2017); Guo et al. (2019); Xu et al. (2021); Huang et al. (2021b) enhanced the mention-to-entity graph paradigm by introducing multi-hop path reasoning and reconstructing the graph based on the obtained path information. Unfortunately, existing graph-based models can not synthesize new domain knowledge.

Generative models are recently proposed to generate triplets flexibly from input documents. Zeng et al. (2018) proposed a CopyRE model to select entities or relations via copy mechanism. Zeng et al. (2020a); Ye et al. (2020) further improved this paradigm by introducing multi-task learning and contrastive learning frameworks. Other generative models utilized additional information. For example, Zhang et al. (2021) proposed a Knowledge-Graph (KG)-enriched Abstract Meaning Representation (AMR) framework which uses external information to enrich the AMR graph extracted from scientific papers. Garg et al. (2021) leveraged transformers to refine semantic embedding of a given

text for better generation. However, Zhang et al. (2020) claimed the factual correctness and trustworthiness problems of these methods which ignore prior knowledge to ensure that generated knowledge triplets are more reliable.

Although several studies have solved part of the challenges, they can not compete with our method in two ways: (1) To utilize external prior knowledge, we propose an innovative dynamic graph generation strategy; (2) To generate rich knowledge triplets, we propose an end-to-end Dual-Graph Resonance Network. Both efforts are rarely studied together to our best knowledge.

3 Dual-Graph Construction

To equip the model with domain knowledge guidance, we introduce a static BKG based on public resources, and then generate a dynamic sub-BKG for each document. The construction process is depicted in Figure 2.

3.1 Static Biomedical Knowledge Graph

Large-scale biomedical knowledge graphs are recently constructed and most of them are made publicly available. Wang et al. (2021) constructed a knowledge graph with 1.47 million triplets and 96,397 entities from multiple sources, such as PubMed, DrugCental etc. The constructed public knowledge graph can provide massive biomedical prior knowledge and make the model more capable of deriving unobserved and factual triplets. It can be formulated as a shared static graph $G_S = \{(e_i, r_{i,j}, e_j) | e_i, e_j \in \mathcal{E}, r_{i,j} \in \mathcal{R}\}$, where $\mathcal{E}$ and $\mathcal{R}$ represent a entity set and a relation set respectively. G_S will participate in dynamic graph generation and play the teacher role in the decoding stage to mitigate the trustworthiness problems. However, simply direct application of G_S may introduce noisy information into generative models and pollute the SciIE results.

3.2 Dynamic Biomedical Knowledge Graph

Inspired by Viswanathan et al. (2021); Neumann et al. (2019), we let $D = \{S_{abs}, S_{int}, S_{cit}\}$ be a biomedical document, which consists of three important sections, i.e., *abstract*, *introduction* and *citance*. Each section $S_* = \{w_i\}_{i=1}^{N_*}$ refers to a sequence of words of length N_*. We aim to derive a sub-graph called *Dynamic Graph* automatically for each document under the prior knowledge guidance of the static graph G_S. *Note that we consider the three sections together when constructing the sub-graph*. Particularly, we propose a novel **E**xtraction-**C**ontextualization-**D**erivation (**ECD**) framework which consists of three key steps:

Extraction We first extract all the biomedical entities from the document D and obtain an entity (i.e., node) set $\mathcal{E}^\circ$ using SciSpacy[3](Neumann et al., 2019). Then, we retrieve pre-defined relations from the static graph G_S and produce an edge set $\mathcal{R}^\circ$. Finally, we construct an initial dynamic graph $G_D^\circ = \{(e_i, r_{ij}, e_j) | e_i, e_j \in \mathcal{E}^\circ, r_{ij} \in \mathcal{R}^\circ\}$, where e_i denotes a head entity, e_j is a tail entity, and r_{ij} denotes the relation. G_D° needs to be expanded to cover unobserved biomedical knowledge.

Contextualization Intuitively, we can choose k-hop expansion (we use one-hop in our experiments) for contextualization because the directly connected entities are always similar. However, such a simple expansion will bring ungovernable noisy entities and relations, which can pollute the dynamic graph generation quality. Instead, we propose a Dynamic Graph Generator (DGG) which produces triplets and expands G_D° as $G_D = \{(e_i, r_{i,j}, e_j) | e_i, e_j \in \mathcal{E}_D, r_{i,j} \in \mathcal{R}_D\}$, where $\mathcal{E}_D$ and $\mathcal{R}_D$ denote the expanded entity set and relation set respectively, $\mathcal{E}^\circ \subseteq \mathcal{E}_D$ and $\mathcal{R}^\circ \subseteq \mathcal{R}_D$.

Inspired by the masked language model (Devlin et al., 2018), we can randomly mask entities in the graph G_D° and train a DGG to recover the original topology. The trained generator, then, can detect the unobserved triplets while avoiding introducing noise. The training procedure are detailed as below:

We mask 10% entities randomly from G_D° and label the masked entities $\mathcal{E}_+^\circ$ as positive instances, and label the remaining unmasked entities $\mathcal{E}_-^\circ$ as negative instances, and $\mathcal{E}^\circ = \mathcal{E}_+^\circ \cup \mathcal{E}_-^\circ$.

Then, we let $\mathbf{e}_i$ be the representation of any entity $\mathbf{e}_i \in \mathcal{E}_-^\circ$ and $\mathbf{e}_j$ be the representation of any entity $\mathbf{e}_j \in \mathcal{E}_+^\circ$. The initial entity representations are obtained based on pre-trained language models, like BERT and its variants (Devlin et al., 2018; Lee et al., 2020). For each entity e_j to be recovered, we can calculate the selected probability based on an attention mechanism as below:

$$p(e_j|\mathcal{E}_-^\circ) = \delta\Big(\text{Dense}\Big(\sum_{e_i \in \mathcal{E}_-^\circ} \alpha_i \mathbf{e}_i\Big)\Big) \qquad (1)$$
$$\alpha_i = \sigma\big([\mathbf{e}_i; \mathbf{e}_j]\mathbf{W}_0\big)$$

where $\delta(\cdot)$ is a sigmoid function, Dense$(\cdot)$ is a fully-connected layer, $[\,;\,]$ denotes a vector concatenation

[3]https://allenai.github.io/scispacy/

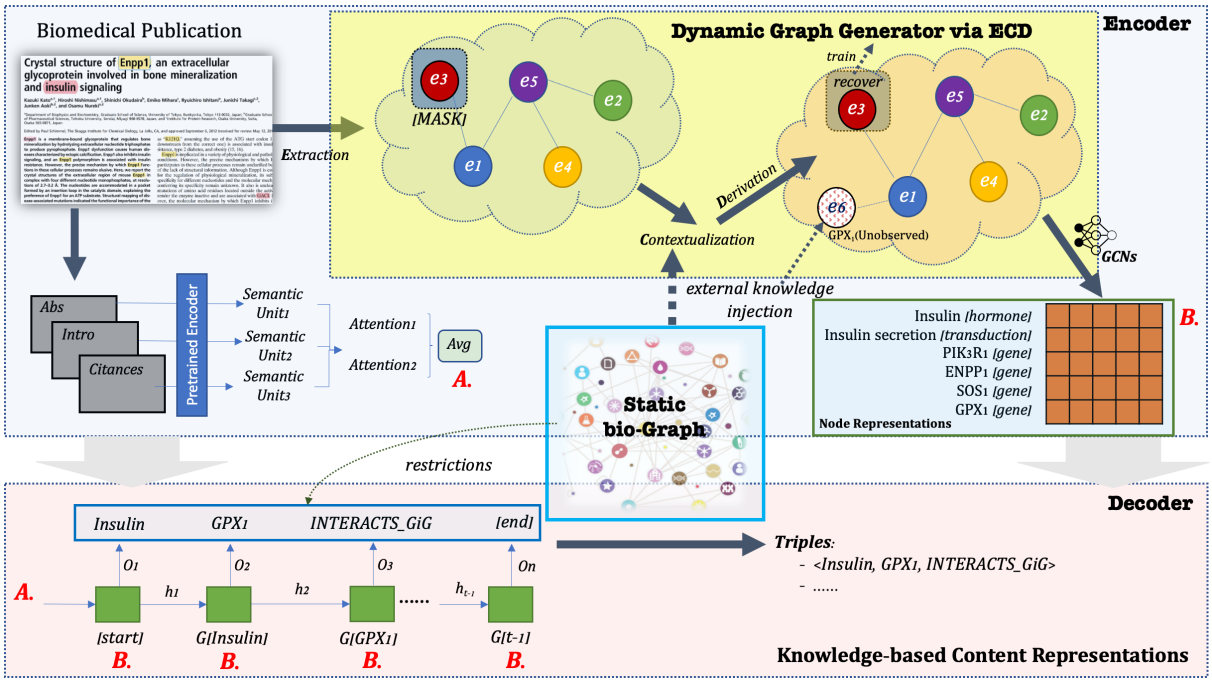

Figure 2: The DGRN architecture and Extraction-Contextualization-Derivation (ECD) strategy. Given any biomedical publication, the dynamic graph G_D can be generated from the static biomedical graph G_S via ECD strategy. Then, both observed and unobserved triplets can be generated by the DGRN model with prior knowledge guidance.

operation, $\sigma(\cdot)$ denotes the ReLU function and $\mathbf{W}_0$ represents a trainable weight matrix.

To train the DGG, we use the cross-entropy function to minimize the loss between the masked entities and their corresponding recovered probabilities:

$$\mathcal{L}_{DGG} = \frac{1}{|\mathcal{E}^{\circ}_{+}|} \sum_{e_j \in \mathcal{E}^{\circ}_{+}} -\log\Big(p\big(e_j|\mathcal{E}^{\circ}_{-}\big)\Big) \quad (2)$$

Derivation In the derivation step, we feed the initial dynamic graph G°_D to the trained DGG which considers one-hop entities as positive instances and the initial entities $\mathcal{E}^{\circ}$ as negative instances. Then, we choose those entities with high probabilities[4] and produce the expanded entity set $\mathcal{E}_D = \mathcal{E}^{\circ} \cup \{e_j\}_{\geq Threshold}$ and the expanded dynamic graph G_D.

4 Dual-Graph Resonance Network

In this section, we propose an innovative Dual-Graph Resonance Network (DGRN) which combines the dual graphs and generative method into a unified framework. The method is depicted in Figure 2, which consists of three modules: *Text Encoder*, *Graph Encoder* and *Triplet Decoder*.

[4]The hyper-parameter of $Threshold$ is set as 0.5.

4.1 Text Encoder Module

Given any document D, its consisted section S_* is fed into a Bio-BERT (Lee et al., 2020) to produce token-level representations ($\mathbf{H}^{abs} = \{\mathbf{h}^{abs}_i\}$, $\mathbf{H}^{int} = \{\mathbf{h}^{int}_i\}$ and $\mathbf{H}^{cit} = \{\mathbf{h}^{cit}_i\}$) and section-level representations ($\mathbf{h}^{abs}_{CLS}$, $\mathbf{h}^{int}_{CLS}$ and $\mathbf{h}^{cit}_{CLS}$). Viswanathan et al. (2021) found that citation sentences are more relevant to the document topics in comparison with abstract and introduction. Based on this, we opt for abstract-aware attention to measure the importance of each token representation $\mathbf{h}^{cit}_i$ through a scoring function using a feed-forward neural network:

$$\alpha^{abs}_i = \text{softmax}\big((\mathbf{h}^{abs}_{CLS})^{\mathrm{T}}\sigma(\mathbf{W}_1\mathbf{h}^{cit}_i + \mathbf{b}_1)\big)$$
$$\mathbf{p}^{abs-cit} = \sum_{i\in[1,N_{cit}]} \alpha^{abs}_i \times \mathbf{h}^{cit}_i \quad (3)$$

Similarly, we also use introduction-aware attention to measure the importance of each token representation $\mathbf{h}^{cit}_i$ through a scoring function as below:

$$\alpha^{int}_i = \text{softmax}\big((\mathbf{h}^{int}_{CLS})^{\mathrm{T}}\sigma(\mathbf{W}_2\mathbf{h}^{cit}_i + \mathbf{b}_2)\big)$$
$$\mathbf{p}^{int-cit} = \sum_{i\in[1,N_{cit}]} \alpha^{int}_i \times \mathbf{h}^{cit}_i \quad (4)$$

where $\mathbf{W}_*$ and $\mathbf{b}_*$ are trainable model parameters, $\mathbf{p}^{abs-cit}$ and $\mathbf{p}^{int-cit}$ represent the abstract-aware citance representation and introduction-aware citance representation, respectively. Finally, we take

the average of the sum of the two vectors and obtain the document representation $\mathbf{p}^{doc}$ as below:

$$\mathbf{p}^{doc} = \frac{1}{2}(\mathbf{p}^{abs-cit} + \mathbf{p}^{int-cit}) \tag{5}$$

4.2 Graph Encoder Module

We apply multi-layer Graph Convolutional Network (GCN) (Kipf and Welling, 2017) on the dynamic graph G_D to aggregate the features from neighbors to obtain node representation. For any entity e_i at the l-th layer, the graph convolutional operation can be applied by the formula as below:

$$\mathbf{e}_i^{(l+1)} = \sigma\Big(\sum_{e_j \in \mathcal{N}_{e_i}} \mathbf{W}_3^{(l)} \mathbf{e}_j^{(l)} + \mathbf{b}_3^{(l)}\Big) \tag{6}$$

where $\mathcal{N}_{e_i}$ denotes the neighbors for the node e_i, $\mathbf{W}_3^{(l)}$ and $\mathbf{b}_3^{(l)}$ are trainable model parameters. In the initial stage, the $\mathbf{e}_i^{(0)} = \frac{1}{t-s+1}\sum_{i\in[s,t]} \mathbf{h}_i^*$ indicates entity i ranges from t-th token to s-th token in any section S_*. Intuitively, the GCN layers can encapsulate rich topological information.

4.3 Triplet Decoder Module

A decoder is adopted to generate knowledge triplets. Given training data, the decoder can copy an entity from the graph G_D as the head entity of the triplet, and then generate a relation for the triplet. Then, it can copy the tail entity from G_D. Repeating this process, the decoder could generate multiple triplets. In time step t $(1 \leq t)$, we can calculate the decoder output $\mathbf{o}_t$ and hidden state $\mathbf{h}'_t$ as follows:

$$\mathbf{o}_t, \mathbf{h}'_t = f(\mathbf{x}_t, \mathbf{h}'_{t-1}) \tag{7}$$

where $f(\cdot)$ represents the RNN-based decoder function, $\mathbf{h}'_{t-1}$ indicates the hidden state of time step $t-1$, and $\mathbf{x}_t$ is the input representation of time step t and defined as below:

$$\mathbf{x}_t = [\mathbf{o}_{t-1}; \mathbf{c}_t]\mathbf{W}_4 \tag{8}$$

where $\mathbf{o}_{t-1}$ denotes the entity or relation representation copied from dynamic graph in time step $t-1$, $\mathbf{c}_t$ is the attention vector (Bahdanau et al., 2014) and $\mathbf{W}_4$ is a trainable weight matrix. In the initial step, input representation $\mathbf{x}_0 = \mathbf{p}^{doc}$ (see Eq. 5).

Attention Vector Entities and relations are generated and treated differently based on their different positions. In the time step t $(t\%3 = 0, 1)$ (generating head or tail), the attention vector $\mathbf{c}_t$ is calculated by copying entities from the entity set $\mathcal{E}_D$ by the following formula:

$$\begin{aligned} \mathbf{c}_t &= \sum_{e_i \in \mathcal{E}_D} \beta_i \times \mathbf{e}_i \\ \beta_i &= \text{softmax}\Big(\sigma\big([\mathbf{h}'_{t-1}; \mathbf{e}_i]\mathbf{W}_5\big)\Big) \end{aligned} \tag{9}$$

where $\mathbf{h}'_{t-1}$ is the hidden state of the decoder in the $t-1$ time step, and $\mathbf{W}_5$ are trainable parameters.

In the time step t $(t\%3 = 2)$ (generating relation), $\mathbf{c}_t$ can be calculated by copying relations from relation set $\mathcal{R}_D$ by the following formula:

$$\begin{aligned} \mathbf{c}_t &= \sum_{r_{ij} \in \mathcal{R}_D} \gamma_{i,j} \times \mathbf{r}_{i,j} \\ \gamma_{i,j} &= \text{softmax}\Big(\sigma\big([\mathbf{h}'_{t-1}; \mathbf{r}_{i,j}; \mathbf{p}_{i,j}]\mathbf{W}_6\big)\Big) \end{aligned} \tag{10}$$

where $\mathbf{W}_6$ are trainable parameters, $\mathbf{r}_{i,j}$ is the relation representation, $\mathbf{p}_{i,j}$ is the representation of edge between entity e_i and e_j via path reasoning.

Path Reasoning In the dynamic graph G_D, the head entity and tail entity in a triplet are not always directly connected. Thus, we introduce a path reasoning method that can model dependency among entities with multi-hop distances in the graph. Similar to (Zeng et al., 2020b), given the head entity and tail entity, we can define the representation of directed edge from entity e_i to entity e_j as below:

$$\mathbf{e}_{ij} = \sigma(\mathbf{W}_7[\mathbf{e}_i; \mathbf{e}_j] + \mathbf{b}_7) \tag{11}$$

where $\mathbf{W}_7$ and $\mathbf{b}_7$ are trainable parameters, $\mathbf{e}_i$ and $\mathbf{e}_j$ denote representations of entity e_i and entity e_j respectively (see Eq. 6).

Based on the vectorized edge representation, the path between head entity e_i and tail entity e_j passing through entity e_o is represented as follow:

$$\mathbf{p}_{i,j} = [\mathbf{e}_{i,o}; \mathbf{e}_{o,t}; \mathbf{e}_{t,o}; \mathbf{e}_{o,j}] \tag{12}$$

For computation efficiency, we choose one-hop path, while it can be extended to multi-hop paths.

Entity Prediction To copy a head/tail entity, we calculate the confidence vector $\mathbf{q} = [q_1, ..., q_{|\mathcal{E}_D|}]$ for all the entities in $\mathcal{E}_D$. We also apply a softmax on $\mathbf{q}$ to obtain the probability distribution $\mathbf{p}^{entity} = [p_1^{entity}, ..., p_{|\mathcal{E}_D|}^{entity}]$ by the formulas as below:

$$\begin{aligned} q_t &= \sigma(\mathbf{o}_t\mathbf{W}_8 + \mathbf{b}_8) \\ \mathbf{p}^{entity} &= \text{softmax}(\mathbf{q}) \end{aligned} \tag{13}$$

where $\mathbf{W}_8$ and $\mathbf{b}_8$ are trainable parameters. We select the entity with the highest probability as the

predicted entity and use its embedding to produce the next time step input $\mathbf{c}_{t+1}$. *Note that the tail entity can not be the same as the head entity*.

Relation Prediction We now use a fully connected layer to calculate a confidence vector $\mathbf{q}' = [q'_1, ..., q'_{|\mathcal{R}_D|}]$ of all the valid relations. Specifically, we apply a softmax on $\mathbf{q}'$ to obtain the probability distribution $\mathbf{p}^{relation} = [p_1^{relation}, ..., p_{|\mathcal{R}_D|}^{relation}]$ by the formulas as below:

$$\begin{aligned} \mathbf{q}' &= \sigma(\mathbf{o}_t \mathbf{W}_9 + \mathbf{b}_9) \\ \mathbf{p}^{relation} &= \text{softmax}(\mathbf{q}') \end{aligned} \tag{14}$$

where $\mathbf{W}_9$ and $\mathbf{b}_9$ are trainable parameters. We select the relation with the highest probability as the prediction relation and use its embedding to produce input $\mathbf{c}_{t+1}$ in the next time step.

Objective Function Our DGRN is trained with the negative log-likelihood loss function. Suppose $Y = \{y_1, y_2, ..., y_T\}$ is the target result for triplets generation, the loss function is defined as:

$$\mathcal{L}_{SEQ} = \frac{1}{T} \sum_{t=1}^{T} -log\big(p(y_t|y_{<t})\big) \tag{15}$$

where T is the maximum steps of the decoder, $p(y_t|y_{<t})$ denotes the conditional probability of target y_t given previous output sequence $y_{<t}$.

There are two negative log-likelihood loss functions used for training our DGRN; one is DGG loss (see Eq. 2) and one is the seq2seq loss (see Eq. 15). We optimize $\mathcal{L}_{DGG}$ and $\mathcal{L}_{SEQ}$ iteratively. We use backpropagation to calculate the gradients of all the trainable parameters and update them with Adam optimizer (Loshchilov and Hutter, 2019).

5 Experiments

5.1 Datasets

Following Wang et al. (2021), we use their public triplet dataset built from multiple public datasets as the bases to construct our static biomedical knowledge graph. This dataset consists of 1,426,025 triplets, 41,078 entities, and 27 relation types. Besides, we also construct a dataset with 32,330 biomedical publications collected from PubMed, and split the dataset into training, development and testing sets with a split of 22,330/5,000/5,000. To create knowledge triplets for each training/development document, we follow Xing et al. (2020) to retrieve biomedical concepts (Achakulvisut et al., 2020). For the testing set, we randomly remove 10% entities and the corresponding sentences from each document for evaluating the entity recovering ability.

Human Agreement Test To evaluate the quality of our dataset, we conduct human agreement tests on a randomly selected small-scale and human-annotated testing set. Specifically, we first randomly selected 50 documents and provide 8 triplets as label candidates for each document. Two annotators with professional linguistics knowledge participated in the annotation task and chose the most appropriate triplets. The Kappa value of manual annotation is 0.88. Besides, we conduct a paired t-test between the best performance baseline and our DGRN, the p-value (2.048) is less than the significance level of 0.05.

5.2 Comparative Study

In Table 1, we compare our DGRN with several state-of-the-art baselines:

HRL is a hierarchical extraction paradigm which approaches relation extraction via hierarchical reinforcement learning (Takanobu et al., 2019).

CasRel is a cascade binary tagging framework, which models relations as functions that map subjects to objects in a sentence (Wei et al., 2020).

CopyRE is a Seq2Seq model with copy mechanism, which can jointly extract relational facts from sentences of any class (*Normal*, *EntityPairOverlap* and *SingleEntiyOverlap*). (Zeng et al., 2018).

CopyMTL is a multi-task learning framework equipped with copy mechanism to allow the model to predict multi-token entities (Zeng et al., 2020a).

GAIN is a double graph resonaing network to better cope with document-level relation extraction, which features double graphs in different granularity (Zeng et al., 2020b).

AGGCN is a soft-pruning approach, that automatically learns how to selectively attend to the relevant sub-structures useful for the relation extraction task. (Guo et al., 2019).

For fair comparison, we also utilize some state-of-the-art works which use pre-trained model:

KECI is an end-to-end model which utilizes external domain knowledge graph for jointly entity and relation extraction(Lai et al., 2021).

TEMPGEN is a a cross-attention guided model for triplet template generation(Huang et al., 2021a).

Note that we replace their encoders with BioBert and retain their best performance settings reported in their paper for fair comparison.

Model	Model Type	P	R	F1
HRL	Extractive	61.17	21.81	32.22
CasRel	Extractive	71.12	32.94	44.71
CopyRE	Generative	54.73	25.72	35.01
CopyMTL	Generative	56.91	29.64	39.11
GAIN	Extractive+Graph	56.01	21.43	31.81
AGGCN	Extractive+Graph	61.43	33.91	44.43
KECI-BioBert	Extractive	61.93	**40.81**	43.66
TEMPGEN-BioBert	Generative	63.13	33.71	49.57
GAIN-BioBert	Extractive+Graph	56.61	21.87	31.55
DGRN	Generative+Graph	**73.77***	39.69*	**51.61***

Table 1: Comparison among different models. Models above the double line do not use pre-trained model. Superscript * indicates statistical significance at $p < 0.05$ level compared to the best performance of baselines.

GCN for Dual-Graph		DGRN Training	
Parameter	**Value**	**Parameter**	**Value**
Number of layers	1, **2**, 3	learning rate	$\mathbf{10^{-3}}$, 10^{-5}
emb size	100	dropout	0.2, **0.5**, 0.8
hidden size	808	batch_size	10, **20**, 50
		weight decay	$\mathbf{10^{-3}}$

Table 2: The experimental settings of our method. The best parameter settings are highlighted.

5.3 Experimental Settings

We choose BioBert as our text encoder and implement DGRN with Pytorch and DGL (Wang et al., 2019). The hyper-parameter settings are detailed in Table 2. We choose commonly used *precision*, *recall* and *F1-score* as the evaluation metrics. All the methods run on a server configured with 4 Tesla P100 GPUs, 16 CPUs, and 512G memory.

5.4 Comparative Results

Table 1 provides experiment results. One can witness the DGRN superiority compared with the best-performed baseline (CasRel) with 15.43% improvement in F1 (because CasRel fails to produce unobserved knowledge triplets). Interestingly, we find that generative methods can not compete with extractive approaches since generative methods require more searching space in the decoding stage than extractive models which only need to select target tokens among candidates. Meanwhile, the graph-based models achieve convincing performance, which indicates that graphs can provide essential topological information for knowledge extraction. We further investigate the performance of other state-of-the-art works using pre-trained model (BioBert). As can be observed in Table 1, DGRN achieves a 3.95% F1 improvement than TEMPGEN-BioBert and a 14.42% improvement in precision. However, we notice that KECI-BioBert gets the highest recall among all models. This can be explained by the fact that KECI-BioBert first constructs a span graph to filter the noisy tokens before encoding the input while our method first encodes the sentence and can only process 1024 tokens in each iteration. We may consider adding position embedding to enlarge the receptive field of DGRN in the future. Overall, DGRN outperforms all baselines because it utilizes topology information with dual-graph resonance to guide knowledge triplet generation while avoids disturbance from noisy information, which assures its effectiveness.

Model	ECD Step	P	R	F1
Extract	1	47.07	23.02	30.91
Extract+Text	1	59.03	38.21	48.01
Extract+Context+Text	1,2	41.17	20.81	27.78
DGRN (Full)	1,2,3	**73.77**	**39.69**	**51.61**

Table 3: Ablation study on dynamic graph generation.

Model	ECD Step	P	R	F1
Extract	1	45.61	19.97	27.78
Extract+Text	1	51.53	37.62	43.50
Extract+Context+Text	1,2	39.01	19.22	25.75
DGRN (Full)	1,2,3	**73.19**	**39.98**	**51.71**

Table 4: Robustness study on dynamic graph generation with training instances containing more than 10 triplets.

5.5 Ablation Study

We also explored how different configurations can impact our model performance. **Extract** only uses the initial graph. **Extract+Text** considers text modeling additionally. **Extract+Context+Text** considers one-hop path expansion by adding up to 20 entities. **DGRN** is our fully configured model. The experimental results are displayed in Table 3.

As can be seen from Table 3, **Extract+Text** outperforms **Extract** due to the semantic richness provided by text. However, interestingly, **Extract+Context+Text** performs the worst which we attribute to the fact that a simple expansion strategy will introduce noisy entities/relations from the static knowledge graph. Our fully configured **DGRN** considers all the necessary components and achieves the best result, which again validates the effectiveness of our *Extraction-Contextualization-Derivation* strategy.

Model	Mask	Precision	Recall	F1-score
DGRN (Full)	**10%**	**64.07**	**38.71**	**48.26**
	15%	61.87	37.22	46.48
	20%	61.21	37.01	46.12
	25%	57.41	31.63	40.79

Table 5: Results on different percent of masked entities.

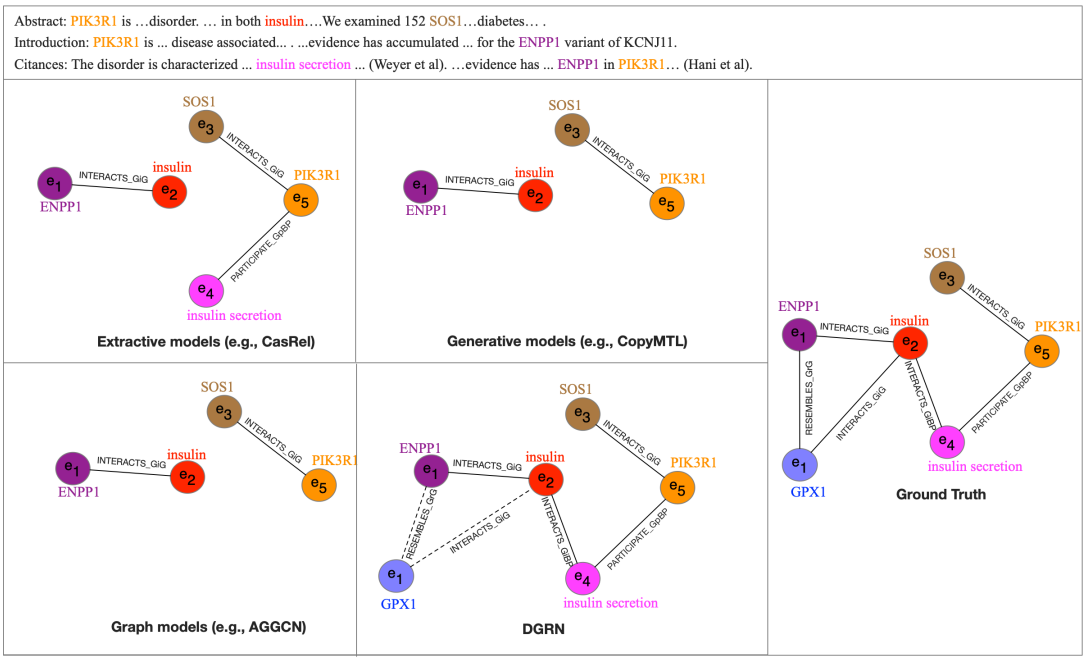

Figure 3: The exemplar generation results of baseline models (CASREL, CopyMTL, AGGCN) and DGRN.

5.6 Robustness Study

Compared with short document, complex long documents can be more challenging for SciIE. Besides, the number of triplets contained in each document can affect the model's performance appreciably. Thus, we build another training set with instances carrying more than 10 triplets. As Table 4 shows, we find that complex documents lead to the substantial performance degradation of the imperfect models. However, the fully configured DGRN is rarely affected because document content and the generated dynamic graph can complement each other. This validates the robustness of our model.

5.7 Masking Study

To validate the masking efficiency, we mask different percentages of nodes for DGG training. The experimental results are presented in Table 5.

From Table 5, we find that the model performance is optimal when 10% of entities are masked from the dynamic graph. This is because the dynamic graph is initialized from a document, and the scale of most constructed graphs can be small (10 to 20 nodes in most cases). Thus, masking small percentages of nodes can help the DGG to learn more ontological information.

5.8 Case Study

We further conduct case study to depict the model effectiveness. As displayed in Figure 3, we can find CASREL is the best performed baseline, however it fails to identify the relation between "*insulin*" and "*insulin secretion*". CopyMTL and AGGCN ignore essential relations and they can not detect the entity "*insulin secretion*". As comparison, DGRN successfully generates the unobserved entity "*GPX1* (gene)" and all the triplets with two vital unobserved triplets, such as "*<GPX1, RESEMBLES_GrG, ENPP1>*" and "*<GPX1, INTERACTS_GiG, insulin>*", which validates its better inference ability and superior performance.

6 Conclusions & Future Work

In this paper, we discuss the challenges of existing extractive and generative methods, and make two efforts to deal with them. First, we propose a novel Extraction-Contextualization-Derivation (ECD) strategy to generate a document-specific dynamic graph from a static knowledge graph. Then, we propose a novel Dual-Graph Resonance Network (DGRN) to generate richer triplets under the guidance of dual-graph. Extensive experiments validate the effectiveness of our proposed method.

In the future, we will study heterogeneous graph simplification (Wu et al., 2019) and sub-graph mining (Jiang et al., 2020) techniques to further improve the performance of our model. Also, we will incorporate the position factors into our model and investigate a proper method to combine graph embedding and text embedding.

Limitations

We identified two limitations during our experiments. First, as BKG can be very large, graph pre-training can be expensive and challenging (i.e., 336 hours on 4 Tesla P100 GPUs for this study). Meanwhile, the domain knowledge-bases and their quality can be different (e.g., biomedical KG quality can be higher than social science KG, and cancer KG quality can be better than parasitic disease KG(Abu-Salih, 2021)). Unfortunately, the static bio-graph used in this work didn't define the domain boundaries, which may bring noise disturbance. More experiments are needed when we generalize this work to other scientific domains. Second, considering limited computational resources, we chose one-hop expansion for the dynamic graph generator, which will ignore some useful long-distance relations.

Ethics Statement

After carefully reviewing the ACL Ethics Policy, we sincerely show our respect and obey to consent all. To reproduce the experiment outcomes, we will make our dataset and code publicly available (for review reasons, the project link is omitted).

References

Bilal Abu-Salih. 2021. Domain-specific knowledge graphs: A survey. *Journal of Network and Computer Applications*, 185:103076.

Titipat Achakulvisut, Daniel Acuna, and Konrad Kording. 2020. Pubmed parser: A python parser for pubmed open-access xml subset and medline xml dataset xml dataset. *Journal of Open Source Software*, 5(46):1979.

Isabelle Augenstein, Mrinal Das, Sebastian Riedel, Lakshmi Vikraman, and Andrew McCallum. 2017. SemEval 2017 task 10: ScienceIE - extracting keyphrases and relations from scientific publications. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 546–555, Vancouver, Canada. Association for Computational Linguistics.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Hongxiao Bai and Hai Zhao. 2018. Deep enhanced representation for implicit discourse relation recognition. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 571–583, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. Scibert: Pretrained language model for scientific text. In *EMNLP*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Krishna Garg, Jishnu Ray Chowdhury, and Cornelia Caragea. 2021. Keyphrase generation beyond the boundaries of title and abstract. *arXiv preprint arXiv:2112.06776*.

Zhijiang Guo, Yan Zhang, and Wei Lu. 2019. Attention guided graph convolutional networks for relation extraction. *arXiv preprint arXiv:1906.07510*.

Sonal Gupta and Christopher Manning. 2011. Analyzing the dynamics of research by extracting key aspects of scientific papers. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 1–9, Chiang Mai, Thailand. Asian Federation of Natural Language Processing.

Kung-Hsiang Huang, Sam Tang, and Nanyun Peng. 2021a. Document-level entity-based extraction as template generation. *arXiv preprint arXiv:2109.04901*.

Quzhe Huang, Shengqi Zhu, Yansong Feng, Yuan Ye, Yuxuan Lai, and Dongyan Zhao. 2021b. Three sentences are all you need: Local path enhanced document relation extraction. *arXiv preprint arXiv:2106.01793*.

Zhuoren Jiang, Zheng Gao, Jinjiong Lan, Hongxia Yang, Yao Lu, and Xiaozhong Liu. 2020. Task-oriented genetic activation for large-scale complex heterogeneous graph embedding. In *Proceedings of The Web Conference 2020*, pages 1581–1591.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*.

Tuan Lai, Heng Ji, ChengXiang Zhai, and Quan Hung Tran. 2021. Joint biomedical entity and relation extraction with knowledge-enhanced collective inference. *arXiv preprint arXiv:2105.13456*.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Yi Luan, Dave Wadden, Luheng He, Amy Shah, Mari Ostendorf, and Hannaneh Hajishirzi. 2019. A general framework for information extraction using dynamic span graphs. 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 3036–3046, Minneapolis, Minnesota. Association for Computational Linguistics.

Preslav I Nakov, Ariel S Schwartz, Marti Hearst, et al. 2004. Citances: Citation sentences for semantic analysis of bioscience text. In *Proceedings of the SIGIR*, volume 4, pages 81–88. Citeseer.

Tapas Nayak and Hwee Tou Ng. 2020. Effective modeling of encoder-decoder architecture for joint entity and relation extraction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 8528–8535.

Mark Neumann, Daniel King, Iz Beltagy, and Waleed Ammar. 2019. ScispaCy: Fast and Robust Models for Biomedical Natural Language Processing. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 319–327, Florence, Italy. Association for Computational Linguistics.

Nanyun Peng, Hoifung Poon, Chris Quirk, Kristina Toutanova, and Wen-tau Yih. 2017. Cross-sentence n-ary relation extraction with graph lstms. *Transactions of the Association for Computational Linguistics*, 5:101–115.

Ryuichi Takanobu, Tianyang Zhang, Jiexi Liu, and Minlie Huang. 2019. A hierarchical framework for relation extraction with reinforcement learning. In *AAAI*, pages 7072–7079.

Vijay Viswanathan, Graham Neubig, and Pengfei Liu. 2021. Citationie: Leveraging the citation graph for scientific information extraction. *arXiv preprint arXiv:2106.01560*.

Minjie Wang, Da Zheng, Zihao Ye, Quan Gan, Mufei Li, Xiang Song, Jinjing Zhou, Chao Ma, Lingfan Yu, Yu Gai, Tianjun Xiao, Tong He, George Karypis, Jinyang Li, and Zheng Zhang. 2019. Deep graph library: A graph-centric, highly-performant package for graph neural networks. *arXiv preprint arXiv:1909.01315*.

Shike Wang, Fan Xu, Yunyang Li, Jie Wang, Ke Zhang, Yong Liu, Min Wu, and Jie Zheng. 2021. Kg4sl: knowledge graph neural network for synthetic lethality prediction in human cancers. *Bioinformatics*, 37(Supplement_1):i418–i425.

Zhepei Wei, Jianlin Su, Yue Wang, Yuan Tian, and Yi Chang. 2020. A novel cascade binary tagging framework for relational triple extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1476–1488.

Felix Wu, Amauri Souza, Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Weinberger. 2019. Simplifying graph convolutional networks. In *International conference on machine learning*, pages 6861–6871. PMLR.

Rui Xing, Jie Luo, and Tengwei Song. 2020. Biorel: towards large-scale biomedical relation extraction. *BMC bioinformatics*, 21(16):1–13.

Wang Xu, Kehai Chen, and Tiejun Zhao. 2021. Document-level relation extraction with reconstruction. In *The 35th AAAI Conference on Artificial Intelligence (AAAI-21)*.

Hongbin Ye, Ningyu Zhang, Shumin Deng, Mosha Chen, Chuanqi Tan, Fei Huang, and Huajun Chen. 2020. Contrastive triple extraction with generative transformer. *arXiv preprint arXiv:2009.06207*.

Daojian Zeng, Haoran Zhang, and Qianying Liu. 2020a. Copymtl: Copy mechanism for joint extraction of entities and relations with multi-task learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 9507–9514.

Shuang Zeng, Runxin Xu, Baobao Chang, and Lei Li. 2020b. Double graph based reasoning for document-level relation extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1630–1640, Online. Association for Computational Linguistics.

Xiangrong Zeng, Daojian Zeng, Shizhu He, Kang Liu, and Jun Zhao. 2018. Extracting relational facts by an end-to-end neural model with copy mechanism. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 506–514, Melbourne, Australia. Association for Computational Linguistics.

Yuhao Zhang, Derek Merck, Emily Tsai, Christopher D. Manning, and Curtis Langlotz. 2020. Optimizing the factual correctness of a summary: A study of summarizing radiology reports. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5108–5120, Online. Association for Computational Linguistics.

Zixuan Zhang, Nikolaus Parulian, Heng Ji, Ahmed Elsayed, Skatje Myers, and Martha Palmer. 2021. Fine-grained information extraction from biomedical literature based on knowledge-enriched Abstract Meaning Representation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6261–6270, Online. Association for Computational Linguistics.

Tianyang Zhao, Zhao Yan, Yunbo Cao, and Zhoujun Li. 2021. Asking effective and diverse questions: a machine reading comprehension based framework for joint entity-relation extraction. In *Proceedings of*

the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence, pages 3948–3954.