Do Syntactic Features Help Biomedical Relation Extraction? An Empirical Study of Verb Token and Dependency Graph Augmentation

Mustafa Kamal Sikder and Ernest Kwegyir-Afful
Hazard Assessment and Analytics Branch
Human Foods Program
U.S. Food and Drug Administration
{Abu.Sikder, Ernest.Kwegyir-Afful}@fda.hhs.gov

Abstract

We investigate whether explicit syntactic features improve transformer-based biomedical relation extraction when added to typed entity marker pooling. We evaluate two augmentation strategies on top of BiomedBERT: (1) verb token augmentation, which concatenates the hidden state of the dependency root verb to the entity representations, and (2) a two-layer graph convolutional network (GCN) that refines encoder hidden states over the dependency parse before entity pooling. We experimented on three biomedical datasets: ChemProt, DDI, and AIMed with three random seeds. We found neither strategy consistently outperformed the entity-only baseline. The GCN yielded modest gains on AIMed ($+0.007$ F1) and ChemProt ($+0.003$ F1) but decreased performance on DDI (-0.013 F1). Verb token augmentation helps only on AIMed ($+0.004$ F1) and underperforms on the other two datasets. A syntactic characterization of the datasets reveals that DDI has substantially higher passive voice usage (50.7% of relation-bearing sentences) than AIMed (27.0%) or ChemProt (30.9%), suggesting that syntactic augmentation is more effective when sentences exhibit active verbal structure with semantically informative predicates. These results suggest that corpus-level syntactic characteristics, particularly passive voice usage, may moderate the utility of explicit syntactic augmentation, though the small magnitude of observed differences warrants caution in interpretation.

1 Introduction

Relation extraction (RE) is a core task in biomedical natural language processing that involves identifying semantic relationships between entities in text (Madan et al., 2019; Huang et al., 2024). Automatically detecting chemical-protein, drug-drug, and protein-protein interactions from scientific literature supports downstream applications in pharmacovigilance, knowledge base construction, and public health and food safety surveillance, including hazard signal detection, adverse event monitoring, and contamination source identification. (Zhang et al., 2025; Bawankar and Kumar, 2026; Munkhdalai et al., 2018; Zuo et al., 2022).

The dominant approach to RE with pretrained language models involves inserting typed entity markers around entity spans in the input sequence, and the model is trained to classify relations from the pooled representations at marker positions (Zhong and Chen, 2021; Ye et al., 2022). This approach has been widely adopted for biomedical RE due to its simplicity and strong empirical performance (Gu et al., 2021).

In most biomedical sentences, the relational predicate or the verb connecting two entities carries essential information about the nature of the relationship. For example, in "aspirin inhibits COX-2", the verb inhibits is the primary signal distinguishing an inhibitor relation from an activator or substrate relation. Majewska et al. (2021) demonstrated that biomedical verbs exhibit domain-specific semantic-syntactic properties distinct from general English, and that structured verb knowledge can improve biomedical NLP models, motivating the hypothesis that the predicate verb carries informative relational signal in biomedical sentences. Additionally, prior work on syntax-augmented RE has demonstrated that incorporating dependency structure can improve performance, particularly for long-range relations (Zhang et al., 2018). However, these approaches were not applied on top of the token representations of the language model encoders.

In this paper, we investigate two strategies for incorporating syntactic information into entity marker-based RE: (1) verb token augmentation, which concatenates the hidden state of the main predicate verb to the entity representations; and (2) dependency graph convolutional network (GCN), which refines entity representations through graph

Proceedings of the 25th Workshop on Biomedical Language Processing (BioNLP 2026), pages 128–134

convolution over the dependency parse before classification. We evaluated both strategies against an entity-only baseline across three biomedical datasets: ChemProt (Krallinger, 2017), the DDI corpus (Herrero-Zazo et al., 2013), and AIMed (Bunescu et al., 2005), using BiomedBERT (Gu et al., 2021) as the base encoder.

2 Related Works

The entity marker approach for relation extraction was introduced by Soares et al., 2019, who inserted special boundary tokens around entities and used their hidden states for classification. Zhong and Chen (2021) showed that typed markers encoding entity categories (e.g., <S:CHEMICAL>) significantly outperform generic ones, and Ye et al. (2022) proposed packed levitated markers that avoid disrupting the original token sequence. Zhong and Chen (2021) demonstrated that concatenating entity start-marker representations with the sentence-level [CLS] embedding yields strong performance with minimal architectural overhead — an approach we adopt as our baseline. In the biomedical domain, BioBERT (Lee et al., 2019) and BiomedBERT (Gu et al., 2021) established that domain-specific pretraining combined with entity markers achieves competitive results on benchmarks such as ChemProt and DDI.

Our work builds on two lines of syntactic augmentation. Zhang et al. (2018) showed that graph convolution over pruned dependency trees improves relation extraction, particularly for long-range entity pairs, though their method operated on static embeddings rather than transformer hidden states. Majewska et al. (2021) demonstrated that biomedical verbs exhibit domain-specific knowledge and retrofitting that into the model can improve classification tasks. We extend both ideas to the transformer entity-marker setting: verb token augmentation directly concatenates the predicate's contextual representation, while our dependency GCN applies graph convolution over the full parse tree on top of BiomedBERT's hidden states.

3 Methods

We investigate three relation representation strategies built on top of a shared transformer encoder. All three share the same input format, training procedure, and evaluation protocol, differing only in how the final relation representation is constructed before classification.

3.1 Base Encoder and Input Representation

We use BiomedBERT (Gu et al., 2021) as our encoder across all conditions. For each candidate entity pair in a sentence, we insert typed markers around the subject and object spans following the PURE framework (Zhong and Chen, 2021).

$$[CLS] < S_S E_1 > aspirin < S_E E_1 > ... \\ < O_S E_2 > COX - 2 < O_E E_2 > [SEP]$$

Where $[CLS]$ and $[SEP]$ are encoder's special tokens, S_S and S_E indicate the start and end of the subject entity markers and O_S and and O_E indicate the object entity marker, and E_1, E_2 were the entity categories (e.g., CHEMICAL, GENE-Y, DRUG). Markers are typed with the entity category so the encoder can distinguish entities semantically. These marker tokens were added to the tokenizer vocabulary and their embeddings are initialized to the vocabulary mean to avoid out-of-distribution activations at the start of fine-tuning. One training example is generated per ordered entity pair per sentence; pairs with no gold relation are assigned the $no\_relation$ label.

The encoder processes the marked-up sequence and produces contextual hidden states $h \in \mathbb{R}_{seq\_len \times 768}$. All three classification models described below operate on these hidden states.

3.2 Entity Marker Pooling (Baseline)

The updated entity representations were obtained from the encoder last hidden state at the position of each entity start marker (e.g. $S_S E_1$ and $O_S E_2$). The vectors were concatenated and passed through layer normalization, dropout, and a linear classifier to produce logits over relation types. This served as the reference condition for all comparisons.

3.3 Verb Token Augmentation

For this condition, together with the entity marker embeddings, we obtained the representation of the main predicate verb and concatenated that with the entity markers. We used spaCy's dependency parser (Honnibal et al., 2020) with $en\_core\_web\_sm$ model. We selected the ROOT token of the dependency tree if its part-of-speech tag was VERB or AUX. If a sentence contained multiple ROOT candidates, we selected the one closest to the subject entity span center. When no ROOT verb was found, a special NO_VERB_TOKEN embedding was appended, initialized to the vocabulary mean.

Unlike the entity markers, which point to dedicated special tokens, the verb index points to the first subword token of the verb word itself, preserving the lexical identity of the predicate alongside its contextual representation. Thus, the hidden state encoded the verb's semantic and syntactic context.

The verb index is extracted from the original (un-marked) sentence token sequence and then adjusted to account for the marker tokens inserted during feature creation. Verb extraction is performed once per sentence during preprocessing and cached for all entity pairs sharing the same sentence. The NO_VERB_TOKEN fallback was triggered for 10.3%, 11.7%, and 7.8% of relations in ChemProt, AIMed, and DDI respectively, indicating that a meaningful minority of instances received no relational verb signal regardless of dataset.

3.4 Dependency Graph Neural Network

In the entity + graph convolutional network (GCN) condition, we replace the direct pooling step with a two-layer graph convolutional network (Kipf and Welling, 2016) that refines the model's hidden states according to the syntactic dependency structure of the sentence before extracting entity representations.

Word-level pooling: Because spaCy operates at the word level while BiomedBERT uses WordPiece subtokenisation, we first mean-pool the BiomedBERT hidden states of all subtokens belonging to the same original word. Marker tokens ($< S_S E_1 >$, etc.) are excluded from this alignment — they have no corresponding spaCy word. This produces a word-level embedding matrix $X \in \mathbb{R}_W \times 768$ where W is the number of original words (capped at 128).

Dependency graph construction: We ran dependency parser on the original unmarked sentence to obtain a dependency parse. We built an undirected adjacency matrix $A \in \mathbb{R}_W \times W$ where $A[i, j] = 1$ if word i and word j share a head-child dependency relation, and $A[i, i] = 1$ (self loops). The matrix is symmetrically normalised as $D^{-1/2}AD^{-1/2}$ where D is the diagonal degree matrix. Adjacency matrices are computed once per unique sentence and cached across all entity pairs from the same sentence.

GCN message passing: The word embeddings are refined through two GCN layers:

$$X^1 = \text{ReLU}(A\_normX^0W^1)... \\ [W \times 768 \rightarrow W \times 256]$$

$$X^2 = \text{ReLU}(A\_normX^1W^2)... \\ [W \times 256 \rightarrow W \times 768]$$

Each layer aggregates information from syntactically connected words. After two layers, each word's representation reflects information from words up to two dependency hops away. Because the dependency graph is undirected, information flows both from head to child and from child to head. The verb node aggregates from its subject and object arguments, and the entity nodes aggregate from the verbs governing them.

Classification: The GCN-refined embeddings at the subject and object word positions are extracted and passed to the classifier:

$$r = LayerNorm([X^2[s\_word]; X^2[o\_word]])$$

Note that s_word and o_word are the word-level indices of the first token of each entity span in the original sentence. The classifier was the same linear layer used in the other conditions, with input dimension 2×768. The GCN weights are randomly initialized and trained jointly with the BiomedBERT encoder from scratch during fine-tuning.

3.5 Training

All three conditions share identical training hyperparameters. We fine-tuned with AdamW (weight decay 0.01 on non-bias/LayerNorm parameters), a linear learning rate schedule with 10% warmup, batch size 16, and learning rate 2×10^{-5}. The maximum sequence length is 256 for ChemProt and AIMed and 384 for DDI (which contains longer sentences). We train for 10 epochs on AIMed and 5 epochs on ChemProt and DDI, selecting checkpoints by the development set F1. All experiments are repeated with three random seeds (0, 1, 42). Our primary evaluation metric was micro F1 over positive relation classes. We reported the mean F1 for all seeds with standard deviation. The loss of Cross-entropy is calculated over all pairs of candidate entities, with $no\_relation$ as a negative class. Gradient clipping is applied at norm 1.0. Marker token embeddings and GCN weights are initialized to the vocabulary mean and random normal respectively.

3.6 Datasets

We evaluate on three biomedical relation extraction datasets spanning chemical-protein, drug-drug, and protein-protein interaction tasks. All datasets used the entity markers described in 3.1 with entity type CHEMICAL, DRUG, or PROTEIN as appropriate.

ChemProt ChemProt (Krallinger, 2017) is a sentence-level relation extraction dataset from the BioCreative VI challenge covering interactions between chemicals and gene/protein targets in biomedical abstracts. Each candidate pair is classified into one of five relation groups (CPR:3 upregulator, CPR:4 downregulator, CPR:5 agonist, CPR:6 antagonist, CPR:9 substrate) or assigned no relation. We use the official BioCreative VI train, development, and test partitions and evaluate using micro F1 over the five positive CPR groups.

DDI Corpus The DDI corpus (Herrero-Zazo et al., 2013) contains pharmacological sentences annotated for drug-drug interactions, drawn from DrugBank database entries and MedLine abstracts. Relations are classified into four types: MECHANISM (pharmacokinetic mechanism of interaction), EFFECT (clinical effect of the interaction), ADVISE (recommendation against co-administration), and INT (unspecified interaction). We use the SemEval 2013 Task 9 train and test partitions. We hold out 20% of the training data for development and use the remaining 80% for training. We evaluate using micro F1 over the four positive relation types.

AIMed AIMed (Bunescu et al., 2005) is a binary protein-protein interaction dataset drawn from MEDLINE abstracts. Each candidate entity pair is labelled as interacting (INTERACT) or not. We use fold 1 of the original data and hold out 10% of the training set as development set using a fixed random seed. While results may vary across folds, the fold assignment only affects the train/dev/test split rather than the underlying data

4 Results

We present results across three conditions: entity marker pooling (Entity), verb token augmentation (Entity+Verb), and dependency GCN (GCN) on three datasets: ChemProt, DDI, and AIMed. All are sentence-level datasets.

4.1 Syntactic Characterization of Datasets

To contextualize the model results, we characterize the syntactic structure of each dataset's test set using automatic dependency parsing (spaCy $en\_core\_web\_sm$). We analyzed all sentences containing at least one gold relation. Table 1 presents sentence-level and relation-level syntactic metrics across the three datasets.

Dependency tree depth: The average tree depth is 7.5, 7.5, and 7.1 for AIMed, ChemProt, and DDI respectively. This indicates that the three datasets have similar structural complexity as measured by tree depth. The datasets also have similar average sentence lengths (26–31 tokens).

Passive voice usage: Passive constructions appear in 50.7% of DDI sentences containing relations, compared to 27.0% in AIMed and 30.9% in ChemProt. At the relation level, 33.6% of DDI gold relations occur in sentences where the ROOT verb is a passive construction which is higher than ChemProt (20.2%) and AIMed (18.9%). The passive ROOT verb is an unreliable predicate signal for relation extraction: in a passive sentence such as "Synergism was observed between methylglyoxal and piperacillin", the ROOT verb observed describes the research act rather than the pharmacological interaction between the entities.

4.2 Model Performance

Table 2 shows test F1 ($mean \pm std$) across three seeds) for all three conditions on all three datasets. Full per-seed results are provided in the Appendix.

Verb token augmentation helped AIMed dataset and underperformed the entity-only baseline on ChemProt and DDI datasets. The GCN condition helped AIMed and ChemProt datasets and performed better than entity+verb condition in the DDI dataset. On DDI, neither augmentation strategy improves over the entity-only baseline. On ChemProt, GCN marginally outperforms entity-only ($+0.003$) method.

Seed-level data suggested additional patterns. On DDI, both augmentation strategies underperformed the baseline across all three seeds, indicating systematic incompatibility. Verb augmentation also substantially increased training instability on DDI (F1 range: $0.755 - 0.770$) compared to the baseline ($0.778 - 0.784$). In contrast, the GCN outperformed the baseline in all three seeds on AIMed. On ChemProt, the GCN matched or exceeded the baseline in all three seeds despite small mean differences, suggesting a consistent modest advantage. To assess whether the observed differences are reliable, we ran paired approximate randomization tests comparing each augmentation strategy against

Metric	AIMed	ChemProt	DDI
Avg sentence length (tokens)	26.2 ± 12.7	30.5 ± 14.6	26.4 ± 13.5
Avg dependency tree depth	7.5 ± 2.1	7.5 ± 2.2	7.1 ± 2.3
Avg dep path between entities (hops)	4.8 ± 2.8	5.6 ± 2.6	6.6 ± 3.3
% sentences with passive voice	27.0%	30.9%	50.7%
% sentences with active ROOT verb	58.6%	55.3%	45.0%
% relations: active verb between entities	11.7%	25.5%	28.4%
% relations: passive ROOT verb	18.9%	20.2%	33.6%
% relations: copular ROOT verb	10.8%	13.3%	9.1%

Table 1: Syntactic characterization of test sets. Sentence-level metrics computed over sentences containing at least one gold relation. Dependency path lengths computed between subject and object entity head tokens. Parser: spaCy *en_core_web_sm*.

Condition	AIMed	ChemProt	DDI
N	222	3463	964
Entity (baseline)	0.818 ± 0.002	0.760 ± 0.006	$\mathbf{0.780 \pm 0.003}$
Entity + Verb	0.822 ± 0.002	0.755 ± 0.006	0.762 ± 0.006
Entity + GCN	$\mathbf{0.825 \pm 0.004}$	$\mathbf{0.763 \pm 0.003}$	0.767 ± 0.006

Table 2: Mean test $F1 \pm std$ across three seeds $(0, 1, 42)$. Bold marks best per dataset. Micro F1 over positive relation classes only. $N =$ number of positive test instances

the baseline separately per dataset (full results in Appendix). No comparison reached statistical significance (all $p > 0.05$), consistent with the small effect sizes and the limited statistical power of three seeds.

5 Discussions

5.1 GCN Outperforms Baseline Where Active Verbal Structure is More Prevalent

The dependency GCN improved over the entity-only baseline on two of three datasets: AIMed ($+0.007$) and ChemProt ($+0.003$). On DDI it underperformed (-0.013). This pattern aligns with a key difference in sentence structure across datasets. AIMed has the lowest passive construction rate (27.0%) and a relatively high proportion of sentences with an active ROOT verb (58.6%). ChemProt is similar (30.9% passive, 55.3% active ROOT). DDI, in contrast, has a substantially higher passive rate (50.7%) and fewer sentences with an active ROOT verb (45.0%). A likely explanation of these could be: when sentences followed an active structure, the dependency graph provided paths between entities that passed through semantically informative verbs. GCN message passing propagated this predicate information into the entity representations before classification. When sentences were predominantly passive, the ROOT verb was a reporting predicate rather than the relational one, and the dependency paths between entities carried less useful signal. This interpretation is consistent with the direction of results across all three datasets, though we caution that three datasets is insufficient to establish this relationship quantitatively.

5.2 Verb Token Augmentation is Inconsistent

Verb token augmentation improves over the baseline only on AIMed ($+0.004$) and underperforms on ChemProt (-0.005) and DDI (-0.018). The inconsistency is likely attributable to the noise floor of automatic ROOT verb extraction. For only 11.7% of AIMed relations, 25.5% of ChemProt relations, and 28.4% of DDI relations (Table 1), an active verb was directly between the two entity spans. Indicating, the smaller proportions of sentences where the ROOT verb reliably signaled the relation. In the remaining cases, the concatenated verb token added a representation that is structurally present but semantically peripheral, competing with rather than complementing the entity marker representations. That verb token helps on AIMed despite its low verb-between-entity rate (11.7%) is counterintuitive to this hypothesis. One possible explanation is that AIMed's binary task structure may be easier to benefit from any additional context, even imperfectly located verbal signal. A multi-class setting like ChemProt, where fine-grained relation types must be distinguished,

may penalize noisy verb tokens more heavily. This remains speculative without further controlled experiments.

5.3 Limitations and Future Directions

Our experiments have several limitations. All syntactic features are extracted using a general-domain dependency parser (spaCy $en\_core\_web\_sm$) not adapted to biomedical text. Parser errors such as, correctly detecting noun phrases and passive construction on complex biomedical sentences can propagate into both the verb token and GCN conditions. A biomedical dependency parser trained on domain-specific treebanks may yield more reliable syntactic features and alter the results for both augmentation strategies. We evaluate on a single encoder (BiomedBERT) and three seeds. Different encoders or a larger seed set may produce different conclusions. For AIMed we use only fold 1 of the 10-fold cross-validation structure; results may not generalize across all folds. Additionally, the GCN improvements on AIMed and ChemProt are small (0.003–0.007) and within or near one standard deviation of the baseline, so they should not be overstated. In future work we will investigate biomedical-domain dependency parsers, pruned dependency paths (Zhang et al., 2018) rather than the full graph.

6 Conclusions

We presented a controlled empirical study of two syntactic augmentation strategies: verb token concatenation and dependency GCN, and compared to entity marker pooling for biomedical relation extraction. Experiments on ChemProt, DDI, and AIMed using BiomedBERT with three random seeds showed that neither strategy consistently outperformed the entity-only baseline. The dependency GCN improved on AIMed ($+0.007$) and ChemProt ($+0.003$) but underperformed on DDI (-0.013). Verb token augmentation helped only on AIMed ($+0.004$).

Our syntactic analysis showed that datasets with lower passive construction rates (AIMed 27.0%, ChemProt 30.9%) benefit from GCN augmentation, while the passive-heavy DDI corpus (50.7%) did not. This suggests that the utility of explicit syntactic features in transformer-based RE depends on the linguistic characteristics of the underlying corpus rather than being universally beneficial. These findings offer practical guidance for biomedical RE systems supporting public health surveillance. When working with literature exhibiting predominantly active sentence structures, dependency GCN augmentation may provide modest improvements. For corpora with high passive voice usage, such as DDI, entity marker pooling alone could be more reliable compared to either augmentation strategy. This is particularly relevant for public health and food safety applications that rely on extracting chemical interactions, adverse events, and contamination signals from biomedical literature, where understanding which augmentation strategies generalize across different writing styles can improve robustness and reliability. For real-world deployment in applications such as chemical hazard surveillance, practitioners can characterize the syntactic profile of their target corpus and validate augmentation strategies on representative data before deployment. We release all code and converted datasets to support reproducibility and encourage future work to explicitly report syntactic characteristics of biomedical RE benchmarks.[1]

References

Sapna Bawankar and Sanjay Kumar. 2026. Scalable knowledge extraction from biomedical literature for drug discovery applications. *Procedia Computer Science*, 275:226–233. 7th International Conference on AI in Computational Linguistics.

Razvan Bunescu, Ruifang Ge, Rohit J. Kate, Edward M. Marcotte, Raymond J. Mooney, Arun K. Ramani, and Yuk Wah Wong. 2005. Comparative experiments on learning information extractors for proteins and their interactions. *Artificial Intelligence in Medicine*, 33(2):139–155. Information Extraction and Summarization from Medical Documents.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. Domain-specific language model pretraining for biomedical natural language processing. *ACM Trans. Comput. Healthcare*, 3(1).

María Herrero-Zazo, Isabel Segura-Bedmar, Paloma Martínez, and Thierry Declerck. 2013. The ddi corpus: An annotated corpus with pharmacological substances and drug–drug interactions. *Journal of Biomedical Informatics*, 46(5):914–920.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.

[1] https://github.com/mustafasikder/Biomedical-Relation-Extraction

Ming-Siang Huang, Jen-Chieh Han, Pei-Yen Lin, Yu-Ting You, Richard Tzong-Han Tsai, and Wen-Lian Hsu. 2024. Surveying biomedical relation extraction: a critical examination of current datasets and the proposal of a new resource. *Briefings in Bioinformatics*, 25(3):bbae132.

Thomas N. Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *CoRR*, abs/1609.02907.

Rabal-O. Lourenço A. Krallinger, M. 2017. Overview of the biocreative vi chemical-protein interaction track. *Proceedings of the BioCreative VI Workshop,*, 141-146.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Sumit Madan, Justyna Szostak, Ravikumar Komandur Elayavilli, Richard Tzong-Han Tsai, Mehdi Ali, Longhua Qian, Majid Rastegar-Mojarad, Julia Hoeng, and Juliane Fluck. 2019. The extraction of complex relationships and their conversion to biological expression language (bel) overview of the biocreative vi (2017) bel track. *Database*, 2019.

Olga Majewska, Charlotte Collins, Simon Baker, Jari Björne, Susan Windisch Brown, Anna Korhonen, and Martha Palmer. 2021. Bioverbnet: a large semantic-syntactic classification of verbs in biomedicine. *Journal of Biomedical Semantics*, 12(1).

Tsendsuren Munkhdalai, Feifan Liu, and Hong Yu. 2018. Clinical relation extraction toward drug safety surveillance using electronic health record narratives: Classical learning versus deep learning. *JMIR Public Health and Surveillance*, 4(2):e29.

Livio Baldini Soares, Nicholas FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. 2019. Matching the blanks: Distributional similarity for relation learning. *CoRR*, abs/1906.03158.

Deming Ye, Yankai Lin, Peng Li, and Maosong Sun. 2022. Packed levitated marker for entity and relation extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4904–4917, Dublin, Ireland. Association for Computational Linguistics.

Yuan Zhang, Xin Sui, Feng Pan, Kaixian Yu, Keqiao Li, Shubo Tian, Arslan Erdengasileng, Qing Han, Wanjing Wang, Jianan Wang, Jian Wang, Donghu Sun, Henry Chung, Jun Zhou, Eric Zhou, Ben Lee, Peili Zhang, Xing Qiu, Tingting Zhao, and Jinfeng Zhang. 2025. A comprehensive large-scale biomedical knowledge graph for AI-powered data-driven biomedical research. *Nature Machine Intelligence*, 7(4):602–614.

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, Brussels, Belgium. Association for Computational Linguistics.

Zexuan Zhong and Danqi Chen. 2021. A frustratingly easy approach for entity and relation extraction. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 50–61, Online. Association for Computational Linguistics.

Min Zuo, Baoyu Zhang, Qingchuan Zhang, Wenjing Yan, and Dongmei Ai. 2022. An entity relation extraction method for few-shot learning on the food health and safety domain. *Computational Intelligence and Neuroscience*, 2022:1–11.