Open Information Extraction with Meta-pattern Discovery in Biomedical Literature

Xuan Wang
University of Illinois at Urbana-Champaign
xwang174@illinois.edu

Yu Zhang
University of Illinois at Urbana-Champaign
yuz9@illinois.edu

Qi Li
University of Illinois at Urbana-Champaign
qili5@illinois.edu

Yinyin Chen
University of Illinois at Urbana-Champaign
ychen409@illinois.edu

Jiawei Han
University of Illinois at Urbana-Champaign
hanj@illinois.edu

ABSTRACT

Biomedical open information extraction (BioOpenIE) is a novel paradigm to automatically extract structured information from unstructured text with no or little supervision. It does not require any pre-specified relation types but aims to extract all the relation tuples from the corpus. A major challenge for open information extraction (OpenIE) is that it produces massive surface-name formed relation tuples that cannot be directly used for downstream applications. We propose a novel framework CPIE (*Clause+Pattern-guided Information Extraction*) that incorporates clause extraction and meta-pattern discovery to extract structured relation tuples with little supervision. Compared with previous OpenIE methods, CPIE produces massive but more structured output that can be directly used for downstream applications. We first detect short clauses from input sentences. Then we extract quality textual patterns and perform synonymous pattern grouping to identify relation types. Last, we obtain the corresponding relation tuples by matching each quality pattern in the text. Experiments show that CPIE achieves the highest precision in comparison with state-of-the-art OpenIE baselines, and also keeps the distinctiveness and simplicity of the extracted relation tuples. CPIE shows great potential in effectively dealing with real-world biomedical literature with complicated sentence structures and rich information.

CCS CONCEPTS

• **Information systems → Data mining**; • **Applied computing → Life and medical sciences**;

KEYWORDS

open information extraction; pattern mining; biomedical information extraction; text mining

ACM Reference Format:
Xuan Wang, Yu Zhang, Qi Li, Yinyin Chen, and Jiawei Han. 2018. Open Information Extraction with Meta-pattern Discovery in Biomedical Literature. In *ACM-BCB'18: 9th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics, August 29-September 1, 2018, Washington, DC, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3233547.3233594

1 INTRODUCTION

Biomedical information extraction (BioIE) aims to automatically extract structured semantic information from unstructured text. It includes tasks such as named entity recognition, relation extraction, and event detection. BioIE has been successfully applied to many downstream applications such as clinical decision support [12] and question answering [4], biological pathway and network analysis [32], integrative biology [35], biocuration [36, 40, 42] and pharmacovigilance [17]. Machine learning (ML) models are widely adopted in current BioIE systems. For example, in a chemical-induced disease (CID) relation extraction task in the recent BioCreative V challenge [41], systems are proposed using models such as support vector machine [33], feature kernels [24] and convolutional neural network [15]. However, ML models rely on human annotation for training data generation, which is time and labor consuming. Moreover, the reliance on human annotation further limits the systems to certain pre-specified relation types and makes it unable to further extend the systems to new relation types.

Open information extraction (OpenIE), a novel information extraction paradigm, begins to attract great attention in the BioIE domain. OpenIE does not require any pre-specified relation types but aims to extract all the relation tuples from a corpus with no or little human supervision. Two major types of method are proposed for OpenIE: clause-based methods and pattern-based methods. Clause-based methods use linguistic features or sentence structures to induct long-distance relationships from sentences [2, 10, 13, 37]. For example, ClausIE [10] performs clause type analysis based on dependency parsing, chunking, and POS tagging to extract all the possible clauses from the input sentences [10]. In the "Clause extraction" step in Figure 1, eight different clauses can be extracted from the input sentence by ClausIE. The clause-based method extracts massive subject-verb-object relation tuples from the corpus in the surface-name form. Pattern-based methods exploit entity type

ACM-BCB'18, August 29-September 1, 2018, Washington, DC, USA

ACM ISBN 978-1-4503-5794-4/18/08…$15.00
https://doi.org/10.1145/3233547.3233594

information and frequent pattern mining to extract entity-attribute-value (EAV) tuples from input sentences [20, 23, 29]. For example, MetaPAD [20] uses context-aware segmentation and synonymous pattern grouping to extract texture meta-patterns and the value tuples from the input sentence [20]. A meta-pattern is a relationship between entity types. In the "Meta-pattern extraction" step in Figure 1, a meta-pattern "CHEMICAL decreased susceptibility to DISEASE in young SPECIES" can be extracted, which is a relation between three entity types. These meta-patterns can further be used to extract corresponding relation tuples from the corpus.

Both types of OpenIE methods have their pros and cons. Clause-based methods are good at resolving long and complicated sentence structures. However, the massive subject-verb-object relation tuples they extract are in purely surface-name form, which cannot be directly used for downstream applications. Pattern-based methods are easy to be extended to any n-ary relations and produce more structured output with entity type information. However, when the whole sentence has a complicated structure, the tokens between the entities can be lengthy, resulting in the sparsity of relation patterns extracted by the pattern-based method. OpenIE methods have been successfully applied in general domain such as extracting information from newspapers. However, they achieved limited success when applying to the biomedical domain due to some specific challenges with the biomedical text:

(1) Sentences are usually long with complicated structures. The entities within one relation may be far apart in a sentence, and one sentence may contain more than one relation type and more than one relation tuple. This greatly limits the performance of current pattern-based methods.
(2) The entity type information is important for both relation type selection/consolidation and relation instance interpretation. This is not considered by clause-based methods.
(3) High-ary relationships exist in sentences, which can provide more complete and accurate information than binary ones. This is also not considered by clause-based methods.

To address the above challenges, we propose a novel framework *CPIE: Clause+Pattern-guided Information Extraction* as shown in Figure 1. CPIE combines the merits of clause-based and pattern-based methods, extracting both relation types (meta-pattern synonymous groups) and relation tuples. The framework requires no supervision for clause extraction, meta-pattern extraction and relation tuple extraction, only a few positive examples for quality meta-pattern selection. Compared with previous OpenIE methods, CPIE produces massive but more structured output that can be directly used for downstream applications. We first resolve the long and complicated sentence structures by extracting short clauses from the input sentences. Then we perform meta-pattern extraction on the short clauses. Quality meta-patterns are selected and synonymous meta-patterns are grouped together as a single relation type. Then quality meta-patterns are used to extract relation tuples from the input sentences. Experiments show that our method achieves the highest precision in comparison with state-of-the-art OpenIE baselines, and also keeps the distinctiveness and simplicity of the extracted relation tuples. Case studies also show the power of CPIE in effectively dealing with real-world biomedical literature with complicated sentence structures and rich information.

2 RELATED WORK

Open-Domain Information Extraction. OpenIE aims to find new extraction paradigms and extract large sets of relational tuples from a corpus with no or little human supervision. OpenIE has been extensively studied in the NLP area. The task of OpenIE was first introduced by the seminal work of Banko et al. [3]. After their TextRunner system, most of the existing work follows two lines: clause-based methods and pattern-based methods.

For clause-based analysis, linguistic features, like dependency parsing results, are used to induct long-distance relationships. For example, ReVerb [11] identifies relational phrases via part-of-speech-based regular expressions. Ollie [37] further expands the syntactic scope of relation phrases and allows additional context information such as attribution and clausal modifiers. Similarly, ClausIE [10] inducts short but coherent pieces of information along dependency paths, which is typically subject, predicate and optional object with complement. Stanford OpenIE [2] adopts a clause splitter using distant training and mapped predicates to a known relation schema statistically. MinIE [13] further improves the clearness of relation tuples by introducing different statistical measures like polarity, modality, attribution, and quantities. ReMine [44] integrates local context and global cohesiveness to reduce the amount of uninformative or incoherent tuples.

Pattern-based information extraction can be traced back to Hearst patterns, such as "NP_0 such as $\{NP_1, NP_2, ...\}$", which are used for hyponymy relation extraction [18]. Mitchell et al. [27] introduce Never-Ending Language Learning (NELL) based on free-text predicate patterns. Patty [29] aims to extract a set of typed lexical patterns along the shortest dependency path. MetaPAD [20] generates quality meta-patterns (i.e., relational patterns between entity types) by context-aware segmentation, groups synonymous meta-patterns, and adjusts entity-type levels for appropriate granularity in the pattern groups. TruePIE [23] further adopts truth discovery ideas to extract reliable meta-patterns. Using these patterns to match the corpus, tuples with entities and relation phrases can be extracted.

Biomedical Open Information Extraction. BioIE aims to automatically extract structured semantic information from unstructured biomedical corpus. It includes tasks such as named entity recognition, relation extraction, and event extraction. The methods for BioIE include dictionary-based methods, rule-based methods, statistical methods, classification-based methods and hybrid methods [39, 43]. Several survey papers are published containing thorough and systematic summaries of previous work on BioIE [1, 5–7, 14, 16, 19, 22, 26, 34, 39, 43, 45]. We focus on introducing some related work about BioOpenIE, which is a novel paradigm of BioIE that is most related to our work.

OpenIE, as a novel information extraction paradigm, begins to attract great attention in BioIE domain. It does not require any pre-specified relation types but aims to extract all the relation tuples from a corpus with no or little human supervision. Liu et al. [25] introduce some applications of OpenIE to BioIE in their survey paper. For example, Attias et al. [28] present an OpenIE system for biomedical named entity recognition based on NELL [27], and propose a method for assessing seed qualities to prevent semantic drift. Nebot et al. [30, 31] propose a scalable method to extract surface-form biomedical relationships not specific to any relation type and

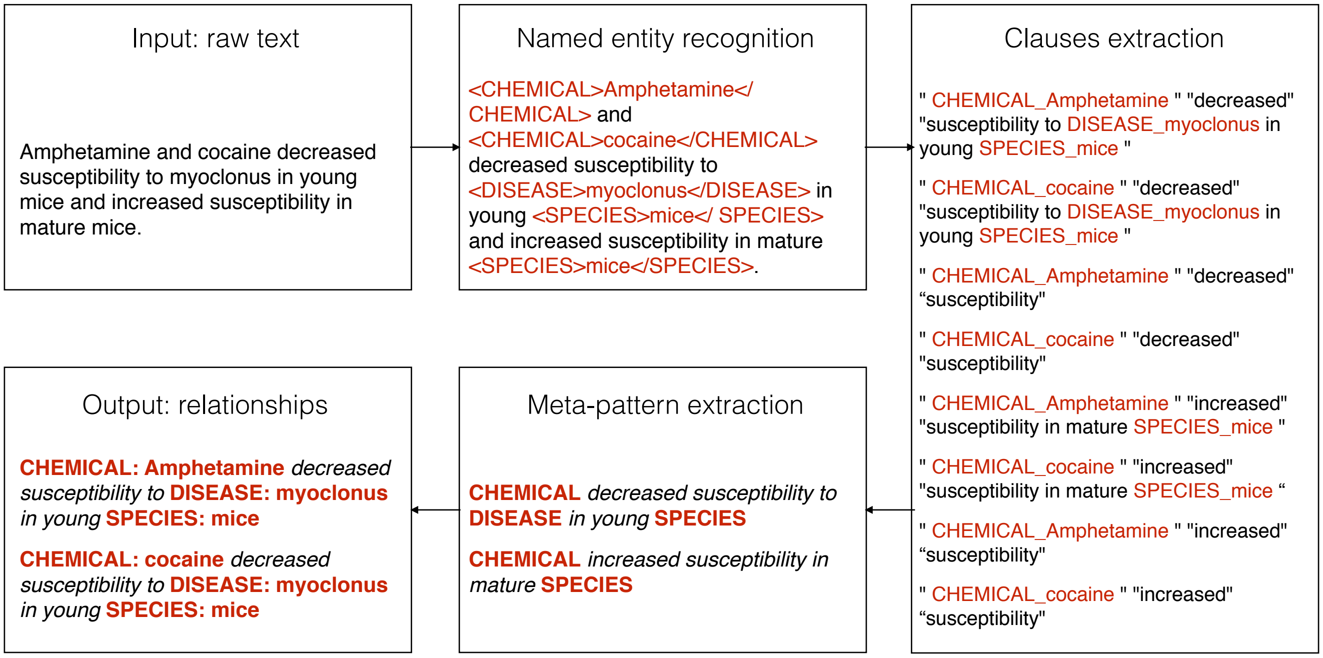
Input: raw text
Amphetamine and cocaine decreased susceptibility to myoclonus in young mice and increased susceptibility in mature mice.
Named entity recognition
<CHEMICAL>Amphetamine</ CHEMICAL> and <CHEMICAL>cocaine</CHEMICAL> decreased susceptibility to <DISEASE>myoclonus</DISEASE> in young <SPECIES>mice</ SPECIES> and increased susceptibility in mature <SPECIES>mice</SPECIES>.
Clauses extraction
" CHEMICAL_Amphetamine " "decreased" "susceptibility to DISEASE_myoclonus in young SPECIES_mice "
" CHEMICAL_cocaine " "decreased" "susceptibility to DISEASE_myoclonus in young SPECIES_mice "
" CHEMICAL_Amphetamine " "decreased" "susceptibility"
" CHEMICAL_cocaine " "decreased" "susceptibility"
" CHEMICAL_Amphetamine " "increased" "susceptibility in mature SPECIES_mice "
" CHEMICAL_cocaine " "increased" "susceptibility in mature SPECIES_mice "
" CHEMICAL_Amphetamine " "increased" "susceptibility"
" CHEMICAL_cocaine " "increased" "susceptibility"
Meta-pattern extraction
CHEMICAL decreased susceptibility to DISEASE in young SPECIES
CHEMICAL increased susceptibility in mature SPECIES
Output: relationships
CHEMICAL: Amphetamine decreased susceptibility to DISEASE: myoclonus in young SPECIES: mice
CHEMICAL: cocaine decreased susceptibility to DISEASE: myoclonus in young SPECIES: mice

Figure 1: The overall framework of CPIE: Clause+Pattern-guided Information Extraction.

further infer the semantic types of the extracted relationships by clustering. De Silva et al. [9] discover inconsistencies in PubMed abstracts through ontology-based information extraction, in which Ollie [37] is a key step.

3 THE BIOOPENIE FRAMEWORK

The overall framework of CPIE is shown in Figure 1. The inputs are sentences from a biomedical corpus. The first step is biomedical named entity recognition (BioNER), which recognizes and classifies the biomedical entities with their proper entity types, e.g., gene, chemical, and disease. Each entity mention is treated as an integrated unit in the process. Then we select the sentences that contain at least one typed entity and perform clause extraction to extract short clauses from the typed sentences. These short clauses are further selected as those that contain at least one typed entity. In each selected clause, we replace the entity mention with its type and extract all the quality meta-patterns as fine-grained relationship types. These quality meta-patterns are further grouped into synonymous groups. Last, we use the quality meta-patterns to extract their corresponding relationship instances from the original input sentences. The extracted meta-patterns and relationship instances are the output of our framework.

Biomedical named entity recognition. We perform BioNER as a preprocessing step of our pipeline to generate a typed corpus. We use Pubtator, a state-of-the-art BioNER tool, to recognize and type the biomedical entities from the corpus [40]. It includes five biomedical entity types: gene/protein, chemical, disease, species, and SNP. Pubtator provides a fully annotated version of the PubMed abstracts that can be directly downloaded from https://www.ncbi.nlm.nih.gov/CBBresearch/Lu/Demo/PubTator/.

4 CLAUSE EXTRACTION

As mentioned above, if the whole sentence has a complicated structure, the tokens between the entities can be lengthy, resulting in the sparsity of relation patterns. To alleviate this problem, we first extract clauses, which are simplified and express some coherent piece of information, from the sentence.

Definition 4.1 (Clause Extraction). A clause is a part of a sentence that consists of one subject (S), one verb (V), and optionally of an indirect object (O), a direct object (O), a complement (C), and one or more adverbials (A). The goal of clause extraction is to segment the whole sentence into clauses and represent them in the format of (subject, relation, optional components).

We use ClausIE [10] for clause extraction. We first concatenate all the words in each entity mention with underlines to make them an integrated unit. Then we select those sentences with at least one typed entity as the input for ClausIE.

For each input sentence, ClausIE conducts the following steps:

Dependency Parsing. We adopt Stanford dependency parser [21] to discover the syntactical structure of a sentence. The output parsing tree has a set of directed syntactic relations between the words in the sentence. Figure 2 shows an example. The root of the tree is the verb "appear". It is connected to "doses" via a subject relation (`nsubj`) and to "useful" via a complement relation (`xcomp`). For a complete list of relations, please refer to https://nlp-ml.io/jg/software/pac/standep.html.

Clause Identification. According to the dependency parsing tree, ClausIE first constructs a clause for each subject relation (e.g., `nsubj`) connecting the subject (S) and the governor verb (V). Besides, objects (O) and complements (C) linked with V through `dobj`, `iobj`,

Table 1: Patterns and clause types we use [10]. S**: Subject,** V**: Verb,** C**: Complement,** O**: Direct object,** O_i**: Indirect object,** A**: Adverbial,** V_i**: Intransitive verb,** V_c**: Copular verb,** V_c**: Extended-copular verb,** V_{mt}**:Monotransitive verb,** V_{dt}**: Ditransitive verb,** V_{ct}**: Complex-transitive verb.**

Pattern	Type	Example	Derived Clauses
		Basic Patterns	
SV_i	SV	Colectomy works.	(Colectomy, works)
SV_eA	SVA	Pulmonary toxicity is in lungs.	(Pulmonary toxicity, is, in lungs)
SV_cC	SVC	Pulmonary toxicity is vital.	(Pulmonary toxicity, is, vital)
$SV_{mt}O$	SVO	Nitrofurantoin causes pulmonary toxicity.	(Nitrofurantoin, causes, pulmonary toxicity)
$SV_{dt}O_iO$	SVOO	Colectomy gives the patient a chance.	(Colectomy, gives, the patient, a chance)
$SV_{ct}OA$	SVOA	Colectomy removes the colon away.	(Colectomy, removes, the colon, away)
$SV_{ct}OC$	SVOC	Pulmonary toxicity causes rats to die.	(Pulmonary toxicity, causes, rats, to die)
		Some Extended Patterns	
SV_iAA	SV	Pulmonary toxicity often appears in lungs in rats.	(Pulmonary toxicity, often appears) (Pulmonary toxicity, often appears, in lungs) (Pulmonary toxicity, often appears, in rats) (Pulmonary toxicity, often appears, in lungs, in rats)
SV_eAA	SVA	Pulmonary toxicity is often in lungs in rats.	(Pulmonary toxicity, is often, in lungs) (Pulmonary toxicity, is often, in lungs, in rats)
SV_cCA	SVC	Pulmonary toxicity is vital in the last century.	(Pulmonary toxicity, is, vital) (Pulmonary toxicity, is, vital, in the last century)
$SV_{mt}OA$	SVO	Nitrofurantoin causes pulmonary toxicity in rats.	(Nitrofurantoin, causes, pulmonary toxicity) (Nitrofurantoin, causes, pulmonary toxicity, in rats)
$ASV_{mt}O$	SVO	In rats, Nitrofurantoin causes pulmonary toxicity.	(Nitrofurantoin, causes, pulmonary toxicity) (Nitrofurantoin, causes, pulmonary toxicity, in rats)

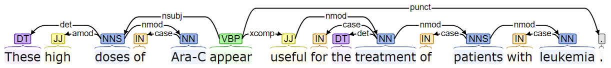

Figure 2: An example sentence with dependency parse.

`xcomp` or `ccomp`, as well as adverbials (A) related with V through `advmod`, `advcl` or `prep_in` will also be included.

Clause Type Analysis. Once clauses have been identified, ClausIE tries to analyze the type of each clause. A complete list of all clause types used in [10] is given in Table 1. Based on the lexical and syntactical structure, the clause type classification is essentially a decision tree. The complete decision flow can be found in [10].

Proposition Generation. Given the clause type, we are able to decide which tokens to place into the subject, the relation, and the optional components. ClausIE maps the subject of the clause to the subject of the proposition. For the optional components, an argument is created and includes the tokens following the verb and then the tokens preceding the verb, in the order in which they appear.

If there is no clause extracted for an input sentence, we will keep the original input sentence as the output clause. The output clauses are further selected to contain at least one typed entity for meta-pattern extraction.

5 META-PATTERN EXTRACTION

Definition 5.1 (Meta-pattern Extraction). Given a corpus C as a list of typed sentences (clauses) S, i.e., $C = [S_1, S_2, ..., S_n]$, each sentence is a sequence of word tokens t, i.e., $S = t_1 t_2 ... t_m$, in which $t_j \in \mathcal{T} \cup \mathcal{W}$ ($\mathcal{T}$ is the set of entity types and $\mathcal{W}$ is the set of non-type words). The goal of meta-pattern extraction is to extract meta-patterns mp, which are sub-sequences of the typed sentences (clauses) S that contain at least one token in the set of entity types $\mathcal{T}$. Quality meta-patterns are selected by some criteria and further grouped into synonymous groups $MPG = [mp_1, mp_2, ..., mp_k]$, in which mp_i and mp_j are synonymous meta-patterns.

Candidate meta-pattern generation. We assume the meta-patterns for relationship extraction should contain at least two typed entities. Given the typed clauses from the previous step, we take the segment between the first and last typed entity in each input clause as a candidate meta-pattern. This step generates a large number of noisy meta-patterns. These candidate meta-patterns are then used for quality selection.

5.1 Quality meta-pattern selection

Given a large number of candidate meta-patterns, we use a set of contextual features to train a classifier that estimates the quality of each candidate meta-pattern. These contextual features have also been adopted by MetaPAD [20].

(1) **Frequency**: A quality meta-pattern should occur frequently in the corpus. We use the normalized count of each meta-pattern $c(mp)/N$, where N is the total number of word tokens in the corpus, to measure the frequency of each meta-pattern. We ignore all the meta-patterns that appear less than 10 times in the corpus, i.e., $c(mp) < 10$.
(2) **Concordance**: A quality meta-pattern should have a frequency significantly higher than that is expected due to chance, which

is a higher concordance. The null hypothesis is: word tokens in the corpus is generated from a series of independent Bernoulli trials. Suppose the total number of word tokens N in the corpus is very large. The expected frequency of a pair of sub-patterns $\langle mp_l, mp_r \rangle$ under the null hypothesis is:

$$\mu_0(c(\langle mp_l, mp_r \rangle)) = N * p(mp_l) * p(mp_r), \tag{1}$$

where $p(mp) = \frac{c(mp)}{N}$ is the empirical probability of the meta-pattern mp. For each meta-pattern mp, we examine all the divisions of dividing mp into $\langle mp_l, mp_r \rangle$ without overlapping. We use Z score to measure the concordance of each meta-pattern mp by finding the division $\langle mp_l, mp_r \rangle$ with the maximum Z score:

$$Z(mp) = \underset{\langle mp_l, mp_r \rangle = mp}{max} \frac{c(mp) - \mu_0(c(\langle mp_l, mp_r \rangle))}{\sigma_{\langle mp_l, mp_r \rangle}}, \tag{2}$$

where $\sigma_{\langle mp_l, mp_r \rangle}$ is the standard deviation of the counts of mp_l and mp_r. For example, the meta-pattern “CHEMICAL against CHEMICAL induced DISEASE” may have a higher concordance than the meta-pattern “GENE, GENE and GENE”, thus is more likely to be a quality meta-pattern.

(3) **Informativeness**: A quality meta-pattern should have more informative context words. We use the average inverse document frequency (IDF) score of all the non-type context words $cw(mp)$ in each meta-pattern mp as its informativeness score:

$$Informativeness(mp) = \frac{\sum_{w \in cw(mp)} log \frac{N}{M(w)}}{|cw(mp)|}, \tag{3}$$

where $M(w)$ is the total number of meta-patterns containing the context word w. For example, the meta-pattern “CHEMICAL induced GENE” may be more informative than the meta-pattern “CHEMICAL and GENE”, thus is more likely to be a quality meta-pattern.

(4) **Completeness**: A quality meta-pattern should have a higher ratio between the frequencies of the meta-pattern and its sub-patterns. We use the average frequency ratio between each meta-pattern mp and all its sub-patterns $subp(mp)$ as its completeness score:

$$Completeness(mp) = \frac{\sum_{s \in subp(mp)} \frac{c(mp)}{c(s)}}{|subp(mp)|}. \tag{4}$$

For example, the meta-pattern “CHEMICAL against CHEMICAL induced DISEASE” may be more complete than its sub-pattern “CHEMICAL against”, thus is more likely to be a quality meta-pattern.

(5) **Coverage**: A quality meta-pattern should extract more distinct instances from the corpus. We used the normalized count of the number of distinct instances $ic(mp)/N$ extracted by each meta-pattern mp to measure the coverage of each meta-pattern.

After extracting the above features for each meta-pattern, we build a random forest based classifier to learn the quality function $Q(mp)$ that maps each meta-pattern to a quality score. We manually selected less than 20 positive and negative meta-patterns, respectively, as the training examples for the model training. After obtaining the quality score of each meta-pattern, we select the meta-pattern with a quality score above a threshold δ as quality meta-patterns. Here we use $\delta = 0.9$. We also assign the same quality score to the extracted instances of each meta-pattern as the instance score for comparison with other OpenIE models.

5.2 Synonymous meta-pattern grouping

Synonymous meta-patterns expressing the same kind of relationship should be grouped together to reduce the redundancy of extracted relationship types and enrich the relationship instances under each relationship type. We group synonymous meta-patterns under the following two assumptions:

(1) Synonymous meta-patterns should have the same entity types. For example, ”CHEMICAL induces DISEASE” and ”CHEMICAL induces GENE” cannot be synonymous meta-patterns.
(2) Synonymous meta-patterns should share similar extracted relationship instances. For example, in Figure 3, the four meta-patterns share some extracted instances and all expressing the meaning of ”CHEMICAL induces DISEASE”, thus should be grouped as a single relationship type.

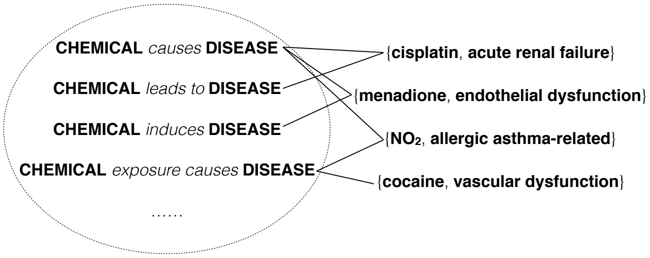
CHEMICAL *causes* DISEASE
CHEMICAL *leads to* DISEASE
CHEMICAL *induces* DISEASE
CHEMICAL *exposure causes* DISEASE
......
{cisplatin, acute renal failure}
{menadione, endothelial dysfunction}
{NO2, allergic asthma-related}
{cocaine, vascular dysfunction}

Figure 3: Grouping synonymous patterns by their shared instance sets.

We use a different method for pattern grouping compared with MetaPAD. For each meta-pattern, we generate the term frequency-inverse document frequency (TF-IDF) vector of its instance sets as its feature vector. The instances also contain the type information. Then we perform k-means clustering to group synonymous meta-patterns in groups. Since the number of groups cannot be predecided, we set k to be the total number of quality meta-patterns divided by 5. We assume that on average, each group will contain 5 synonymous meta-patterns. Examples of synonymous meta-pattern groups are discussed in Section 6.3.

6 EXPERIMENTS

6.1 Dataset

CPIE only requires biomedical text corpus as input. To test the effectiveness of CPIE, we collect a subset of PubMed paper abstracts. The PubMed id (PMID) of each selected paper can be found in the Comparative Toxicogenomics Database (CTD) [8], in which all the selected PMIDs are shown to contain some biomedical relationships. CTD is a human-curated database containing biological entities and their relationships. The entity and relationship statistics of our collected papers in CTD are shown in Table 2.

We focus on three entity types in CTD: gene, chemical, and disease. Among these three entity types, there are three relation types in CTD: chemical-gene, chemical-disease, and disease-gene relationships. We first randomly select 248,064 relationships from

Table 2: Statistics of the CTD dataset subset used in our experiments.

	# of Entity	# of Relation	# of PMID
Chemical-Gene	Chemical: 7,187 Gene: 667	163,126	26,786
Chemical-Disease	Chemical: 7,187 Disease: 598	84,104	1,619
Disease-Gene	Gene: 669 Disease: 599	834	1,622
Total	**Chemical: 7,187** **Gene: 669** **Disease: 599**	**248,064**	**28,007**

the above three relation types that are associated with experimental evidence in the CTD database. Among these relationships, there are 7,187 chemical, 669 gene and 599 disease entities. CTD also provides the PMIDs of PubMed papers related to these selected relationships. We collect all the 28,007 PubMed abstracts that are shown to be associated with the above relationships in CTD as our input corpus. All the following experiments are performed on this PubMed subset corpus.

6.2 Performance comparison

Baselines. To show the effectiveness of CPIE, we compare it with the following state-of-the-art OpenIE approaches:

- **ClausIE** [10] adopts clause patterns to handle long-distance relationships.
- **Stanford OpenIE** [2] learns a clause splitter via distant training data.
- **Ollie** [37] utilizes open pattern learning and extracts patterns over dependency path and part-of-speech tags.
- **MinIE** [13] refines tuples extracted by ClausIE by identifying and removing parts that are considered overly specific.

Evaluation Metrics. We randomly sample 100 sentences from the 28,007 input PubMed abstracts for performance comparison. All of the compared benchmarks, as well as our method, will assign a confidence score to each extracted tuple. We rank all the tuples according to their confidence scores. Based on the ranking list, the following measures can be adopted: (1) *P@k* is the ratio of correct tuples in the top *k* extractions. (2) *MAP* is the mean average precision of the whole ranking list. (3) *NDCG@k* is the normalized discounted cumulative gain at rank *k*. The detailed definition of these common ranking measures can be found in [38]. Note that we do not use recall in OpenIE since it is infeasible to know all the "correct" tuples.

For each method, we select the top 100 tuples from its ranking list and manually label them. The annotator is asked to evaluate without knowing which model produced the results, eliminating potential bias in evaluation. Similar to the settings in previous studies [10], one tuple will be judged as correct if it reads smoothly and meets the fact described in the sentence. For example, both ("DISEASE", "is", "DISEASE") and ("DISEASE", "induced by", "CHEMICAL in SPECIES") are correct. However, ("SPECIES", "is", "a CHEMICAL") and ("CHEMICAL", "inhibited", "GENE and") will not be counted since they have logical or syntactical mistakes. Besides, each tuple should describe exactly one proposition. Tuples with zero or more than two propositions (e.g., ("CHEMICAL", "and", "CHEMICAL") and ("DISEASE", "is", "induced by CHEMICAL and has no relationships with GENE")) will be labeled as incorrect.

Table 3: Performance comparison with state-of-the-art OpenIE systems, using Precision (P@k), Mean Average Precision (MAP) and Normalized Discounted Cumulative Gain (NDCG@k). $k = 50, 100$.

	P@50	P@100	MAP	NDCG@50	NDCG@100
ClausIE [10]	0.800	0.860	0.800	0.710	0.875
Stanford [2]	0.800	0.800	0.765	0.714	0.870
Ollie [37]	0.920	0.890	0.917	0.935	0.982
MinIE [13]	0.840	0.830	0.820	0.800	0.918
CPIE	**0.940**	**0.900**	**0.956**	**0.956**	**0.991**

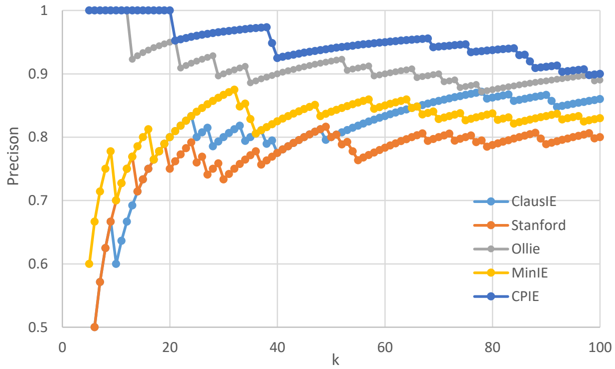

Figure 4: The Precision@*k* curves of different methods.

Results. Table 3 and Figure 4 demonstrate the performances of different OpenIE approaches. Our method CPIE can be seen as clause extraction + meta-pattern discovery. We also tried MetaPAD alone without clause extraction but it doesn't work due to pattern sparsity in biomedical text. In Table 3, our method is consistently the best according to the rank-based measures, especially compared with ClauseIE without meta-pattern discovery. Ollie ranks the second and achieves a similar *P*@100 with our method. However, for *MAP* and *NDCG*@50, CPIE outperforms Ollie by a large margin, indicating we have a rather high precision for top-ranked tuples. In Figure 4, our curve is higher than other baselines for any *k*. Besides, we can observe that for CPIE and Ollie, *P*@*k* decreases with *k*, which means the tuple score is a good indicator of the correctness. From this perspective, the features we defined in Section 5.1 are reliable in evaluating the quality of tuples or meta-patterns.

Distinctiveness and Simplicity. For OpenIE, there are two other important criteria.

- **Distinctiveness**: For the same sentence, we expect that the extracted tuples should have different semantics with each other. It is not satisfying if they are just paraphrasing each other.
- **Simplicity**: Each tuple should clearly explain only one proposition. The ideal cases could be using phrases with two or three words to represent subjects, predicates, and objects.

A direct way to examine distinctiveness of our extractions is to calculate the average Jaccard similarity between extractions from

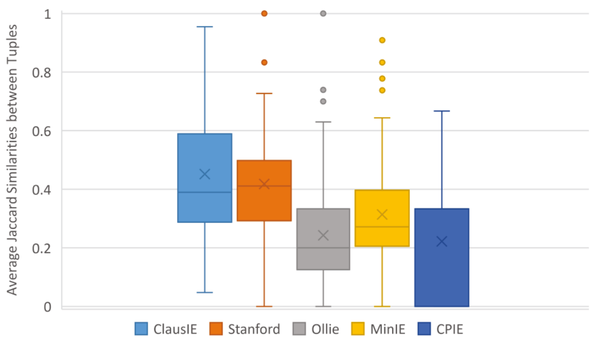

Figure 5: Average Jaccard similarity between extracted tuples in each sentence.

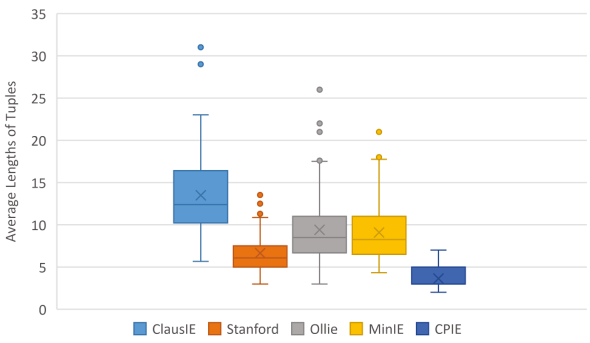

Figure 6: Average length of extracted tuples in each sentence.

the same sentence. Suppose we get n tuples $\{(h_1, r_1, t_1), (h_2, r_2, t_2), \ldots, (h_n, r_n, t_n)\}$ from sentence s[1], the average Jaccard similarity is defined as

$$Avg\ Jacc\ Sim(s) = \frac{1}{\binom{n}{2}} \sum_{1 \le i < j \le n} \frac{|\{h_i, r_i, t_i\} \cap \{h_j, r_j, t_j\}|}{|\{h_i, r_i, t_i\} \cup \{h_j, r_j, t_j\}|}, \quad (5)$$

where $\{h_i, r_i, t_i\}$ denotes the set of tokens appearing in any component of tuple i.

We present the average Jaccard similarity distribution of all sentences in Figure 5, from which we can clearly see that Ollie and CPIE extracts the most distinctive facts as they both consider not to be overly specific. In contrast, ClausIE and Stanford OpenIE suffer for the duplication problem.

To evaluate simplicity, we calculate the average tuple length for each sentence:

$$Avg\ Tup\ Len(s) = \frac{1}{n} \sum_{i=1}^{n} |\{h_i, r_i, t_i\}|. \quad (6)$$

For different approaches, the distributions of the average tuple length are shown in Figure 6. CPIE again performs the best since meta-patterns are considered to be short and clear. For the tuples extracted by ClausIE, the object component is always very complicated since ClauseIE directly segment the whole sentence into several parts. Sometimes the tuples are too long to illustrate one proposition clearly. According to our evaluation protocol, these tuples will be judged as wrong. In fact, even if the long tuples are logically and syntactically correct, it may not help downstream applications.

6.3 Case study

We first look at the quality meta-patterns and the synonymous meta-pattern groups as extracted relation types. Then we examine some extracted relation tuples in detail.

6.3.1 Quality meta-pattern distribution. After candidate meta-pattern generation, we get 185,403 two-entity and three-entity candidate meta-patterns in total. Most of the candidate meta-patterns appear less than 10 times in the corpus. Those low-frequency meta-patterns have the potential to be further resolved and consolidated with the high-frequency meta-patterns. In this study, we ignore those low-frequency meta-patterns, which results in around 1,000 candidate meta-patterns for quality validation. After ranking these candidate meta-patterns with their quality score (a score between 0 and 1), we selected those meta-patterns with a quality score above 0.9 as quality meta-patterns, which results in around 200 quality meta-patterns.

The count distribution of quality meta-patterns is shown in Figure 7. Different types of the meta-patterns are in different colors in the pie chart. For example, “CHEMICAL” means all the entities in the patterns are chemicals, and “GENE, CHEMICAL” means the pattern consists of both chemicals and genes regardless of their order. From the pie chart, we can see that most of the meta-patterns are relationships between gene and chemical, followed by meta-patterns between disease and chemical, and meta-patterns between chemicals.

In Table 4, we list the top 15 quality meta-patterns and their counts. These top meta-patterns describe concrete biomedical relationships between different entity types, which indicates that our quality pattern selection method performs well. The typical raw counts of these quality patterns are around 100 to 300, which is in the medium range. Apparently, the frequency is not the most indicating feature for quality meta-patterns. Informativeness and coverage play a more important role in identifying quality meta-patterns according to our feature importance analysis of the random forest classifier.

6.3.2 Synonymous meta-pattern groups. We group quality meta-patterns into around 40 synonymous groups. In Table 5, we list some examples of the synonymous pattern groups. For example, the first group refers to the relation type “CHEMICAL induce DISEASE”, which includes meta-patterns such as “CHEMICAL causes DISEASE”, “CHEMICAL leads to DISEASE”, “CHEMICAL induces DISEASE” and “CHEMICAL exposure causes DISEASE”. The meta-patterns within each group has very close semantic meaning and each group can be regarded as a specific relation type. These synonymous meta-pattern groups show the effectiveness of CPIE in automatically extracting relation types from massive corpus without supervision. By grouping synonymous meta-patterns together, we reduce the redundancy of extracted relation types and enrich

[1] We only calculate the average Jaccard similarity when $n \geq 2$, and here we only consider 2-ary relations since multi-ary ones may be extensions of them.

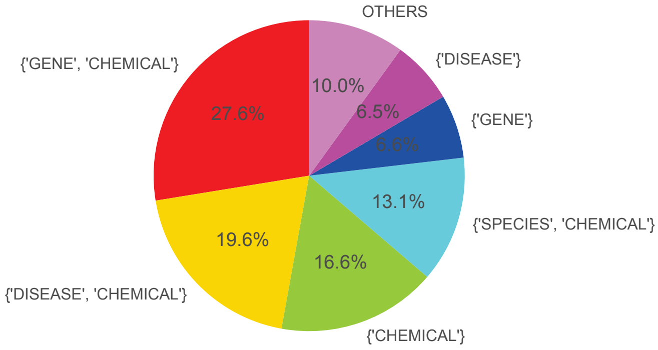

Figure 7: Pie chart of quality meta-pattern counts by group.

Table 4: Top-15 quality meta-patterns.

Quality meta-patterns	Count
CHEMICAL induced DISEASE	306
CHEMICAL inhibited GENE	270
CHEMICAL increased GENE	245
DISEASE be induced by CHEMICAL	233
DISEASE induced by CHEMICAL	181
CHEMICAL inhibits GENE	153
GENE receptor is GENE	145
GENE inhibitor CHEMICAL	138
CHEMICAL inhibited CHEMICAL	126
CHEMICAL inhibited DISEASE	101
DISEASE be induced by CHEMICAL in SPECIES	25
CHEMICAL induces apoptosis in SPECIES DISEASE	25
DISEASE induced by CHEMICAL in SPECIES	20
CHEMICAL against CHEMICAL induced DISEASE	15
SPECIES be treated with CHEMICAL	275

the relation instances for each type, which makes the output more structured for downstream applications.

6.3.3 *Relationship instances extracted by quality meta-patterns.*

For each quality meta-pattern we extracted, we perform pattern matching in the input corpus and successfully identify many relation tuples for each meta-pattern. In Table 6, we list some examples of the quality meta-pattern and its corresponding relation tuples, together with the PMID in which we extracted this tuple. We showed that the tuple extraction is of high accuracy in Section 6.2. For example, the first meta-pattern “CHEMICAL increased GENE” extracts a relation tuple (“forskolin”, “renin”) under this relation type from the Pubmed abstract with PMID “9256163”. The original sentence is: “Forskolin (10 microM), an activator of adenylyl cyclase, and terbutaline (100 microM), a beta2-adrenergic agonist known to increase cAMP levels, also increased renin mRNA and prorenin release.” Although the two entities, “forskolin” and “renin”, are far apart in such a long and complicated sentence, CPIE is able to accurately extract them under the relation of “CHEMICAL increased GENE”. One may note that, in this example, “renin mRNA and prorenin release" could be a more accurate subject in the “increased" relation. It indicates that extending entity recognition to entity phrase recognition for prepossessing may further improve the performance of our system.

Moreover, there are more than one relation type and more than one relation instance in the above sentence. CPIE will also be able to extract the relation tuple (”terbutaline”, “renin”) under the relation type “CHEMICAL increased GENE”, and the relation tuple (”terbutaline”, “cAMP”) under the relation type “CHEMICAL increased CHEMICAL” simultaneously. This is mainly because we first resolve the sentence structure by extracting short clauses, and then extract quality meta-patterns based on the extracted clauses. This example also shows the power of CPIE in dealing with real-world biomedical literature with complicated sentence structures and rich information in an efficient and high-quality way.

7 DISCUSSION

7.1 Biomedical named entity recognition with distant supervision

Current meta-patterns only include the most common biomedical entity types - gene, chemical, and disease. Recognizing more entity types will further enrich the extracted meta-patterns and relation tuples. For example, this is a sentence from PubMed papers with PMID “236533”: “Colectomy is more effective than high-dose steroid therapy in reversing the growth retardation caused by ulcerative colitis and is of greatest value if not delayed too long.” In this sentence, “colectomy” and “steroid therapy” are treatment technologies and “ulcerative colitis” is a disease. If we can recognize the entity type “TREATMENT”, we will be able to extract a relation type “TREATMENT treat DISEASE” from the above sentence.

Most biomedical named entity recognition systems use supervised machine learning models, which require human labeled training dataset for model development. However, it may not be possible to acquire the human labeled training data for all the entity types that we are interested in. One way is to leverage distant supervision, which automatically labels the corpus by some distant examples in the knowledge base and then use this partially labeled corpus for model training. For example, Medical Subject Headings (MeSH) is a knowledge base for biomedical entities. We can find “colectomy” in MeSH ontology, which can serve as a distant supervision example for recognizing the entity type “TREATMENT”. It will greatly benefit our current framework if more entity types can be recognized as the first step of the pipeline.

7.2 Extend meta-pattern extraction

The current framework utilizes ClausIE to extract short clauses from input sentences and resolve the long and complicated sentence structures. However, the ClausIE output is often noisy and redundant, and errors from ClausIE can be propagated down to the next step of meta-pattern extraction. One way is to directly extend the texture meta-pattern extraction method to long and complicated sentence structures. For example, patterns on the dependency-parsing tree of the sentence can be incorporated with sequential textual patterns for long-distance pattern discovery. How to extend the meta-pattern extraction methods to directly extract meta-patterns on such complicated sentences is an interesting problem.

Table 5: Examples of synonymous groups of the quality meta-patterns.

Synonymous group	Meta-patterns
CHEMICAL induce DISEASE	CHEMICAL causes DISEASE
	CHEMICAL leads to DISEASE
	CHEMICAL induces DISEASE
	CHEMICAL exposure causes DISEASE
CHEMICAL inhibit GENE	CHEMICAL decreased GENE
	CHEMICAL decreases GENE
	CHEMICAL inhibition of GENE
	CHEMICAL suppressed GENE
CHEMICAL no effect on GENE	CHEMICAL had no effect on GENE
	CHEMICAL did not affect GENE
DISEASE induced by CHEMICAL in SPECIES	DISEASE induced by CHEMICAL in SPECIES
	DISEASE be induced by CHEMICAL in SPECIES
SPECIES treated with CHEMICAL	SPECIES were pretreated with CHEMICAL
	SPECIES were administered with CHEMICAL
	CHEMICAL treated SPECIES
	SPECIES be induced by CHEMICAL
	SPECIES were exposed to CHEMICAL

Table 6: Examples of relationship instances of the quality meta-patterns.

Meta-pattern	Entity 1	Entity 2	Entity 3	PMID
CHEMICAL increased GENE	forskolin	renin	-	9256163
	nicotine	GM-CSF	-	9606035
	TCDD	TGF-alpha	-	11309286
	CP461	PKG	-	11602670
	beta-naphthoflavone	CYP1B1	-	12843640
CHEMICAL inhibited DISEASE	rapamycin-FKBP	retinoblastoma	-	7532117
	bile salts	cholestasis	-	12644037
	TAS-108	tumor	-	15671561
	zinc	cytotoxicity	-	15922008
	DY-9760e	brain edema	-	16987238
DISEASE be induced by CHEMICAL in SPECIES	pulmonary toxicity	nitrofurantoin	rats	1313237
	colon cancers	PhIP	rats	14507667
	metabolic disorders	PFOA	human	23978341
	carcinogenesis	arsenic	humans	19524636
	liver injury	carbon tetrachloride	rats	17173083
CHEMICAL induces apoptosis in SPECIES DISEASE	arsenic trioxide	human	gastric cancer	11146441
	flavopiridol	human	leukemia	11464216
	isoflurane	rat	pheochromocytoma	18227305
	sodium butyrate	human	hepatoma	15177505
	butyrate	human	hepatoma	15177505
CHEMICAL against CHEMICAL induced DISEASE	dimercaptosuccinic acid	arsenic	toxicity	15998567
	zinc	cadmium	disorders in bone metabolism	23726800
	selenium	cadmium	hematological disturbances	24954678
	resveratrol	cisplatin	testicular damage	28606469
	erdosteine	acetaminophen	renal toxicity	16532256

8 CONCLUSIONS

We propose a novel framework CPIE: Clause+Pattern-guided Information Extraction that automatically extract both relation type and relation tuples with little supervision. CPIE first resolves the long and complicated sentence structures by clause extraction and then uses texture meta-patterns to extract n-ary tuples with entity type information. Quality meta-patterns are selected and synonymous meta-patterns are grouped to produce more structured output for downstream application. Our method achieves the highest precision in comparison with state-of-the-art OpenIE baselines and keeps the distinctiveness and simplicity of extracted relation tuples. Case studies also show the power of CPIE in effectively dealing with real-world biomedical literature with complicated sentence structures and rich information. Future work to improve our framework includes: (1) include more entity types with distant supervision, (2) replace ClausIE with an extended meta-pattern extraction method to reduce error propagation.

ACKNOWLEDGMENTS

The research was sponsored in part by grant 1U54GM114838 awarded by NIGMS through funds provided by the trans-NIH Big Data to Knowledge (BD2K) initiative (www.bd2k.nih.gov), U.S. Army Research Lab. under Cooperative Agreement No. W911NF-09-2-0053 (NSCTA), DARPA under Agreement No. W911NF-17-C-0099, National Science Foundation IIS 16-18481, IIS 17-04532, and IIS-17-41317, and DTRA HDTRA11810026. Any opinions, findings, and conclusions or recommendations expressed in this document are those of the author(s) and should not be interpreted as the views of any U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

REFERENCES

[1] Sophia Ananiadou, Paul Thompson, Raheel Nawaz, John McNaught, and Douglas B Kell. 2014. Event-based text mining for biology and functional genomics. *Briefings in functional genomics* 14, 3 (2014), 213–230.

[2] Gabor Angeli, Melvin Jose Johnson Premkumar, and Christopher D Manning. 2015. Leveraging linguistic structure for open domain information extraction. In *ACL'15*. ACL, 344–354.

[3] Michele Banko, Michael J Cafarella, Stephen Soderland, Matthew Broadhead, and Oren Etzioni. 2007. Open information extraction from the web.. In *IJCAI'07*. AAAI, 2670–2676.

[4] Yonggang Cao, Feifan Liu, Pippa Simpson, Lamont Antieau, Andrew Bennett, James J Cimino, John Ely, and Hong Yu. 2011. AskHERMES: An online question answering system for complex clinical questions. *Journal of biomedical informatics* 44, 2 (2011), 277–288.

[5] Wendy W Chapman and K Bretonnel Cohen. 2009. Guest Editorial: Current issues in biomedical text mining and natural language processing. *Journal of biomedical informatics* 42, 5 (2009), 757–759.

[6] Aaron M Cohen and William R Hersh. 2005. A survey of current work in biomedical text mining. *Briefings in bioinformatics* 6, 1 (2005), 57–71.

[7] Kevin Bretonnel Cohen and Dina Demner-Fushman. 2014. *Biomedical natural language processing*. Vol. 11. John Benjamins Publishing Company.

[8] Allan Peter Davis, Cynthia J Grondin, Robin J Johnson, Daniela Sciaky, Benjamin L King, Roy McMorran, Jolene Wiegers, Thomas C Wiegers, and Carolyn J Mattingly. 2016. The comparative toxicogenomics database: update 2017. *Nucleic acids research* 45, D1 (2016), D972–D978.

[9] Nisansa de Silva, Dejing Dou, and Jingshan Huang. 2017. Discovering inconsistencies in pubmed abstracts through ontology-based information extraction. In *ACM-BCB'17*. ACM, 362–371.

[10] Luciano Del Corro and Rainer Gemulla. 2013. Clausie: clause-based open information extraction. In *WWW'13*. ACM, 355–366.

[11] Anthony Fader, Stephen Soderland, and Oren Etzioni. 2011. Identifying relations for open information extraction. In *EMNLP'11*. ACL, 1535–1545.

[12] Carol Friedman, George Hripcsak, Lyuda Shagina, and Hongfang Liu. 1999. Representing information in patient reports using natural language processing and the extensible markup language. *Journal of the American Medical Informatics Association* 6, 1 (1999), 76–87.

[13] Kiril Gashteovski, Rainer Gemulla, and Luciano Del Corro. 2017. MinIE: minimizing facts in open information extraction. In *EMNLP'17*. ACL, 2630–2640.

[14] Ralph Grishman. 2012. Information Extraction: Capabilities and Challenges.(2012). *Notes prepared for the 2012 International Winter School in Language and Speech Technologies* (2012).

[15] Jinghang Gu, Fuqing Sun, Longhua Qian, and Guodong Zhou. 2017. Chemical-induced disease relation extraction via convolutional neural network. *Database* 2017 (2017).

[16] Udo Hahn, K Bretonnel Cohen, Yael Garten, and Nigam H Shah. 2012. Mining the pharmacogenomics literature–a survey of the state of the art. *Briefings in bioinformatics* 13, 4 (2012), 460–494.

[17] Rave Harpaz, Alison Callahan, Suzanne Tamang, Yen Low, David Odgers, Sam Finlayson, Kenneth Jung, Paea LePendu, and Nigam H Shah. 2014. Text mining for adverse drug events: the promise, challenges, and state of the art. *Drug safety* 37, 10 (2014), 777–790.

[18] Marti A Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *ACL'92*. ACL, 539–545.

[19] Jing Jiang. 2012. Information extraction from text. In *Mining text data*, Charu C Aggarwal and ChengXiang Zhai (Eds.). Springer, 11–41.

[20] Meng Jiang, Jingbo Shang, Taylor Cassidy, Xiang Ren, Lance M Kaplan, Timothy P Hanratty, and Jiawei Han. 2017. MetaPAD: Meta Pattern Discovery from Massive Text Corpora. In *KDD'17*. ACM, 877–886.

[21] Dan Klein and Christopher D Manning. 2003. Accurate unlexicalized parsing. In *ACL'03*.

[22] Martin Krallinger, Alfonso Valencia, and Lynette Hirschman. 2008. Linking genes to literature: text mining, information extraction, and retrieval applications for biology. *Genome biology* 9, 2 (2008), S8.

[23] Qi Li, Meng Jiang, Xikun Zhang, Meng Qu, and Jiawei Han. 2018. TruePIE: Discovering reliable patterns in pattern-based information extraction. In *KDD'18*. ACM.

[24] Zhiheng Li, Zhihao Yang, Hongfei Lin, Jian Wang, Yingyi Gui, Yin Zhang, and Lei Wang. 2016. CIDExtractor: A chemical-induced disease relation extraction system for biomedical literature. In *BIBM'16*. IEEE, 994–1001.

[25] Feifan Liu, Jinying Chen, Abhyuday Jagannatha, and Hong Yu. 2016. Learning for biomedical information extraction: Methodological review of recent advances. *arXiv preprint arXiv:1606.07993* (2016).

[26] Stéphane M Meystre, Guergana K Savova, Karin C Kipper-Schuler, John F Hurdle, et al. 2008. Extracting information from textual documents in the electronic health record: a review of recent research. *Yearb Med Inform* 35, 8 (2008), 128–144.

[27] Tom M Mitchell, William W Cohen, Estevam R Hruschka Jr, Partha Pratim Talukdar, Justin Betteridge, Andrew Carlson, Bhavana Dalvi Mishra, Matthew Gardner, Bryan Kisiel, Jayant Krishnamurthy, et al. 2015. Never Ending Learning.. In *AAAI'15*. AAAI, 2302–2310.

[28] Dana Movshovitz-Attias and William W Cohen. 2012. Bootstrapping biomedical ontologies for scientific text using nell. In *Proceedings of the 2012 Workshop on Biomedical Natural Language Processing*. ACL, 11–19.

[29] Ndapandula Nakashole, Gerhard Weikum, and Fabian Suchanek. 2012. PATTY: a taxonomy of relational patterns with semantic types. In *EMNLP'12*. ACL, 1135–1145.

[30] Victoria Nebot and Rafael Berlanga. 2011. Semantics-aware open information extraction in the biomedical domain. In *Proceedings of the 4th International Workshop on Semantic Web Applications and Tools for the Life Sciences*. ACM, 84–91.

[31] Victoria Nebot and Rafael Berlanga. 2014. Exploiting semantic annotations for open information extraction: an experience in the biomedical domain. *Knowledge and information Systems* 38, 2 (2014), 365–389.

[32] Alexander Nikitin, Sergei Egorov, Nikolai Daraselia, and Ilya Mazo. 2003. Pathway studio – the analysis and navigation of molecular networks. *Bioinformatics* 19, 16 (2003), 2155–2157.

[33] Yifan Peng, Chih-Hsuan Wei, and Zhiyong Lu. 2016. Improving chemical disease relation extraction with rich features and weakly labeled data. *Journal of cheminformatics* 8, 1 (2016), 53.

[34] Jakub Piskorski and Roman Yangarber. 2013. Information extraction: Past, present and future. In *Multi-source, multilingual information extraction and summarization*, Thierry Poibeau, Horacio Saggion, Jakub Piskorski, and Roman Yangarber (Eds.). Springer, 23–49.

[35] Dietrich Rebholz-Schuhmann, Anika Oellrich, and Robert Hoehndorf. 2012. Text-mining solutions for biomedical research: enabling integrative biology. *Nature Reviews Genetics* 13, 12 (2012), 829.

[36] Fabio Rinaldi, Simon Clematide, Hernani Marques, Tilia Ellendorff, Martin Romacker, and Raul Rodriguez-Esteban. 2014. OntoGene web services for biomedical text mining. *BMC bioinformatics* 15, 14 (2014), S6.

[37] Michael Schmitz, Robert Bart, Stephen Soderland, Oren Etzioni, et al. 2012. Open language learning for information extraction. In *EMNLP'12*. ACL, 523–534.

[38] Hinrich Schütze, Christopher D Manning, and Prabhakar Raghavan. 2008. *Introduction to information retrieval*. Vol. 39. Cambridge University Press.

[39] Matthew S Simpson and Dina Demner-Fushman. 2012. Biomedical text mining: A survey of recent progress. In *Mining text data*, Charu C Aggarwal and ChengXiang Zhai (Eds.). Springer, 465–517.

[40] Chih-Hsuan Wei, Hung-Yu Kao, and Zhiyong Lu. 2013. PubTator: a web-based text mining tool for assisting biocuration. *Nucleic acids research* 41, W1 (2013), W518–W522.

[41] Chih-Hsuan Wei, Yifan Peng, Robert Leaman, Allan Peter Davis, Carolyn J Mattingly, Jiao Li, Thomas C Wiegers, and Zhiyong Lu. 2015. Overview of the BioCreative V chemical disease relation (CDR) task. In *Proceedings of the fifth BioCreative challenge evaluation workshop*. Sevilla Spain, 154–166.

[42] Thomas C Wiegers, Allan Peter Davis, and Carolyn J Mattingly. 2014. Web services-based text-mining demonstrates broad impacts for interoperability and process simplification. *Database* 2014 (2014).

[43] Deyu Zhou, Dayou Zhong, and Yulan He. 2014. Biomedical relation extraction: from binary to complex. *Computational and mathematical methods in medicine* 2014 (2014), 1–18.

[44] Qi Zhu, Xiang Ren, Jingbo Shang, Yu Zhang, Frank F Xu, and Jiawei Han. 2018. Open Information Extraction with Global Structure Constraints. In *WWW'18*. ACM, 57–58.

[45] Pierre Zweigenbaum, Dina Demner-Fushman, Hong Yu, and Kevin B Cohen. 2007. Frontiers of biomedical text mining: current progress. *Briefings in bioinformatics* 8, 5 (2007), 358–375.