AN ONTOLOGY BASED REPRESENTATION OF SEMANTIC ANNOTATIONS FOR BIOMEDICAL RELATIONS EXTRACTED FROM SCIENTIFIC DOCUMENTS

by

Berkay Ataman

B.S., Computer Engineering, Istanbul Technical University, 2017

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Computer Engineering

Boğaziçi University

2022

ACKNOWLEDGEMENTS

First and foremost, I would like to state my profound gratitude to my advisor, Assist. Prof. Susan Michele Üsküdarlı for her endless patience, continuous guidance and encouragement throughout our study. It was an honor for me to work with her and I greatly appreciate her efforts, understanding and dedicated involvement.

I am very thankful to my thesis committee members; Assoc. Prof. Arzucan Özgür and Assist. Prof. Zeynep İlknur Karadeniz Erol for their valuable time, thoughful comments and recommendations.

I am very grateful to my mother Zuhal Ataman and my father Mehmet Ataman for their support and love. You have been amazing parents and always supported me. I am able to complete this work thanks to your efforts throughout my life.

I thank to my brother Burak Ataman, and my sister in law Melda Ataman for their friendship and hospitality. Spending time and discussing anything with you guys is a great enjoyment for me.

Finally, I thank to Seren Civan. I love you and am so glad to be with you. Without your endless support and love, I would have never done this.

ABSTRACT

AN ONTOLOGY BASED REPRESENTATION OF SEMANTIC ANNOTATIONS FOR BIOMEDICAL RELATIONS EXTRACTED FROM SCIENTIFIC DOCUMENTS

The sheer volume of scientific literature challenges researchers to identify and utilize the knowledge embedded in them and makes automated extraction and processing necessary. Automated processing becomes especially significant when the extracted information is combined with the linked data resources represented with ontologies. The vast knowledge space represented in Linked Open Data sources provides numerous knowledge discovery opportunities through semantic searching and inference. This thesis aims to extract biomedical entity relations embedded in scientific articles and semantically represent them in a machine-processable manner. For this purpose, we proposed an ontology named Biomedical Entities Evidence (BEE) that represents biomedical entity relationships as well as their provenance in scientific articles. To express the approach's feasibility, we extracted chemical-protein multiclass relations and chemical-disease binary relations. These relations are represented based on BEE ontology. To demonstrate the benefits of ontology-based semantic representation, we have implemented a semantic application prototype that utilizes several Linked Open Data sources and inferred data based on ontologies and custom rules. This prototype was used to evaluate the benefits of the semantic representation by performing information retrieval tasks of varying complexities.

ÖZET

BİLİMSEL BELGELERDEN ÇIKARILMIŞ BİYOMEDİKAL İLİŞKİLER İÇİN ANLAMSAL AÇIKLAMALARIN ONTOLOJİ TEMELLİ TEMSİLİ

Bilimsel literatürün büyük hacmi dolayısıyla, araştırmacıların, makalelerin içlerinde gömülü olan bilgileri tanımlaması ve kullanması zorluk teşkil etmekte ve otomatik çıkarma ve işlemeyi gerekli kılmaktadır. Otomatik işleme, çıkarılan bilgi ontolojilerle temsil edilen bağlantılı veri kaynaklarıyla birleştirilebileceği düşünüldüğünde özellikle önemli hale gelmektedir. Bağlı Açık Veri (LOD) kaynaklarında temsil edilen geniş bilgi alanı, anlamsal arama ve çıkarım yoluyla çok sayıda bilgi keşfi fırsatı sunar. Bu tez, bilimsel makalelerde gömülü biyomedikal varlık ilişkilerini çıkarmayı ve bunları makine tarafından işlenebilir bir şekilde anlamsal olarak temsil etmeyi amaçlamaktadır. Bu amaçla, biyomedikal varlık ilişkilerini ve bunların bilimsel makalelerdeki kökenini temsil eden Biomedical Entities Evidence (BEE) adlı bir ontoloji önerdik. Yaklaşımın uygulanabilirliğini ifade etmek için kimyasal-protein çok sınıflı ilişkileri ve kimyasal-hastalık ikili ilişkilerini çıkardık. Bu ilişkileri BEE ontolojisi aracılığıyla temsil ettik. Ontoloji tabanlı semantik temsilin faydalarını göstermek için, birkaç LOD kaynağını ve ontolojilere ve özel kurallara dayalı çıkarsanan verileri kullanan bir semantik uygulama prototipi geliştirdik. Bu prototip, değişen karmaşıklıklarda bilgi alma görevlerini gerçekleştirerek anlamsal temsilin faydalarını değerlendirmek için kullanıldı.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS/ABBREVIATIONS

BERT	Bidirectional Encoder Representations from Transformers
CORD-19	COVID-19 Open Research Dataset
COVID-19	Coronavirus Disease of 2019
DOI	Digital Object Identifier
LOD	Linked Open Data
NER	Named Entity Recognition
NEN	Named Entity Normalization
RDF	Resource Description Framework
RE	Relation Extraction
SPARQL	SPARQL Protocol and RDF Query Language

1. INTRODUCTION

The number of articles published digitally in the scientific literature is dramatically increasing. The overall growth rate of scientific literature is estimated to be 4.10% per annually and doubling every 17 years [1]. State of emergencies like the outbreak of Coronavirus Disease in 2019 (COVID-19) pandemic cause a surge of scientific information as well. As of December 2021, over 200,000 articles were published related to COVID-19 just in PubMed, a free search engine of biomedical topics [2] [3]. The abundance of scientific literature makes it difficult for scientists to keep up with knowledge embedded in articles. Finding relevant information is far from trivial. As the number of publications increases, the need for systems to process knowledge in the literature is becoming critical. There are many natural language processing approaches to extract and process knowledge embedded in articles. Some of the popular methods are based on term frequency-inversed document frequency (TF-IDF) [4], latent Dirichlet allocation (LDA) [5], and deep learning [6].

Biomedical science has a significant impact on human life due to the focus on preventing and treating diseases and disabilities; therefore, much research is being done in this area. MEDLINE is a bibliographic database of life sciences containing over 30 million articles. In 2019, the average daily article uploads to MEDLINE was approximately 2,700 [7]. Biomedical articles often refer to biomedical entities, such as chemicals and proteins, that are somehow related. The knowledge embodied in such articles about biomedical entity interactions is vital when pursuing therapeutic drug development, clinical disease diagnosis, and understanding the mechanism of diseases [8] [9].

In computer science, ontologies specify formal and explicit representations of the concepts and relations for a given domain [10]. They are used to represent and process domain-specific information. The Web Ontology Language (OWL) is a logic-based language for specifying ontologies, which enables reasoning about the knowledge

expressed with ontologies. A common use of ontologies is treating them as references to standard vocabularies, such as for named entity linking and normalization. In such tasks the ontologies are resources which are subjected to dictionary lookups [11] or to construct word vectors based on concepts [12]. There are numerous ontologies that represent great many concepts, such as drugs, diseases, radiology, and anatomy [13]. Other than using as a reference resource, ontologies are also used for the semantic representation of embedded knowledge to express extracted information with concepts, relations between concepts, and how they are associated [14].

Computers have limited capabilities to capture and exploit the conceptualizations of data and the meaning of content. An unstructured article is a set of literal strings which limits the processing of the concepts and relations, and the integration of existing data distributed over the Web. For instance, relationships between concepts must be explicit for a query like "amino-acids whose activities decreased by hydroperoxide". Data sources must refer to the same concepts to integrate information distributed over the Web for a query like "articles that mention about chemicals that treat arterial-hypertension". Ontology-based semantic representation provides explicit conceptualization, which enables reasoning to infer conclusions like relationships between concepts and interoperability between data sources that refer to the same concept.

The Semantic Web is an extension of the World Wide Web to make internet data machine-readable; thus, applications can automate tasks without human intervention. In the context of the Semantic Web, Linked Open Data (LOD) is a set of principles for linking and sharing data generated by heterogeneous sources. As of May 2020, there are over 1300 datasets published in Linked Open Data Cloud in different research areas like life sciences, economics, linguistics. One of the earliest adopters of LOD are biomedical researchers due to the technical challenges they face to integrate heterogeneous biomedical knowledge sources [15]. Analyzing different data sources to reconcile the mappings of the same concepts, same relations, and same entities should not be the task of a researcher. For this purpose, knowledge in LOD is represented with ontologies explicitly; thus, computers can connect data resources in a consistent way.

The knowledge of biomedical entity interactions is embedded in scientific literature and not readily available for computer processing. Ontology-based representation of biomedical entity interactions will enable computer reasoning to infer conclusions and interoperability with other knowledge sources on the Web.

This thesis proposes an ontology-based representation approach to semantically represent biomedical entity relations in biomedical articles in a machine-processable manner. For this purpose, we introduce an ontology called Biomedical Entities Evidence (BEE) that represents biomedical entity relations and interactions with biomedical articles. Several biomedical ontologies were reused in BEE ontology to comply with the Semantic Web practices. Additionally, best practices of ontology development were utilized to implement BEE ontology [16].

In order to demonstrate the utility of ontology-based representation, an application is developed to annotate biomedical entity relations in biomedical articles and to represent semantically. The application annotates documents with chemicals, proteins, diseases, and biomedical relationships. Later, it represents articles based on BEE ontology. Biomedical article abstracts of COVID-19 Open Research Dataset (CORD-19) [17] have been processed for a proof of concept.

To demonstrate the benefits of ontology-based representation, a web application prototype is developed which processes semantically represented data according to the user needs. The prototype presents biomedical articles and related biomedical entities referred in articles with evidence to the user under several predefined query templates. The prototype utilizes several external data sources in Linked Open Data to enrich captured knowledge.

Main contributions of the thesis are:

(i) 1,728 multiclass chemical-protein relations and 819 binary chemical-disease relations are extracted from over 52 thousand biomedical articles associated with

1,023 chemicals, 604 proteins, and 364 diseases.

(ii) proposed Biomedical Entities Evidence (BEE) ontology to represent biomedical entity relationships with provenance in scientific articles semantically

(iii) over 54 thousand semantic triples corresponding to the extracted relations from which 174 thousand triples are inferred for automated processing (expansion of 4.21%)

(iv) a web based prototype to demonstrate the semantic processing opportunities created by ontologically represented biomedical relations

(v) several tasks related to biomedical relations of various complexities to evaluate the expressiveness of BEE and the utility of representing biomedical relations with it.

The remainder of this thesis is structured as follows: Chapter 2 presents a review of the literature related to the ontology-based representation of biomedical knowledge. Chapter 3 provides background information about several approaches, services, and tools related to this thesis. Chapter 4 introduces the approach to annotate biomedical articles with chemical, protein, and disease relations and introduces the BEE ontology. Chapter 5 introduces the implementation of biomedical entity relation annotation, semantic representation and utilization. Chapter 6 focuses on the assessment of the representation of biomedical relations extracted form scientific documents with use of the proposed ontology. Chapter 7 presents our observations and experiences regarding the approach we have proposed and the potentials for further development. Finally, in Chapter 8 we provide concluding remarks.

2. RELATED WORK

This chapter contains the related works and their comparison with this work. The related work chapter is divided into two main sections. In the Section 2.1 utilization of Semantic Web in several works described. In Section 2.2 several semantic annotation works on biomedical domain have described.

2.1. Utilization of Semantic Web

In 2001 the Semantic Web was introduced by Tim Berners Lee et al. [18], to make the internet machine-readable. Several technologies were introduced for the Semantic Web. Resource Description Framework (RDF) was developed to represent data in machine-processable standards. SPARQL Protocol and RDF Query Language (SPARQL) were introduced to retrieve and manipulate data stored in RDF format.

While RDF format enables machine readability, it does not solve the data interpretation problem since every data source determines its own semantics on data. Ontologies emerged due to the need to share domain-specific models and knowledge. Ontologies define formal semantics of information with standardized terms and relationships between concepts and contain inference rules of data. Web Ontology Language (OWL) was developed to author ontologies for modeling knowledge in different domains. As well as modeling specific domains like life sciences, health, and finance; knowledge bases with encyclopedic data such as Wikidata introduced as part of the Linked Open Data (LOD) project.

Semantic Web approaches are applied to many domains. Yıldırım and Uskudarli [19] extracted topics of social media posts and represented with *Topico* ontology to enable machine processing, reasoning, and integration of Linked Open Data. In this way, microblog post insights are discovered beyond the explicitly represented. Roldan-Garcia et al. [20] proposed a case-centric ontology to represent liver patient cases. They

reused several ontologies from subdomains of medicine and represented patient data by *LICO* ontology. They demonstrated the success of ontological representation with the information retrieval task by using semantic reasoner. Aggelen et al. [21] transformed parliament proceedings into RDF format to make machine-processable and published documents on Linked Open Data with the name of *LinkedEP*. With this approach, users could formulate more complex queries.

2.2. Semantic Annotation in Biomedical Domain

Semantic annotation refers to the process of attaching data about relevant concepts to the unstructured content. Semantic annotation for text documents aims to identify entities as real-world concepts. Thereby, unstructured text can be processed by computers. Machine processable text facilitates many tasks like information retrieval, classification, or interpretation. Semantic annotation task is a popular research area for the biomedical domain. As Jovanovic and Bagheri [22] states, semantic annotation techniques are used on biomedical articles and electronic medical records due to their growing size and need for computer assistance.

A semantic annotation consists of two main tasks, entity recognition, and entity linking. Recent annotators usually rely on machine learning techniques to identify and link named entities. Dictionaries and ontologies are also used for the linking process of an entity.

There are services to annotate given text or document semantically. National Center for Biomedical Ontology (NCBO) annotator [23] is a famous example of such service, which is a web service that has two stages for the annotation process. Firstly, the input text is processed on MGrep [24], a concept recognizer tool that uses dictionaries to identify entities. UMLS Metathesaurus and BioPortal ontologies are used to build concept dictionaries. In the second stage, the NCBO annotator defines a semantic similarity between linked concept and other concepts in the ontology and try to enrich annotation data by adding similar concepts as annotations. However, this service

only annotates documents and does not propose a method to benefit from extracted data. This work's main contribution is the ontology-based representation of semantic annotations; thus, data can be utilized semantically. Below several applications and their approaches to annotate and benefit from biomedical data are described.

2.2.1. Textpresso

An early work of ontology-based semantic annotation in the biomedical domain is Textpresso [25] which is an ontology-based text mining and search engine platform. Textpresso processes every sentence individually to identify biological concepts and relationships of concepts. Predefined regular expressions are used to identify concepts and relationships. Every identified concept is marked with special XML tags that indicate the concept class, and individual sentences are stored as marked for further processing. When a user queries any concept, a lookup to ontology is executed to gather subclasses of a concept and a lookup to marked sentences executed to fetch related sentences. Textpresso ontology consists of biological entities and relationships incorporated with Gene Ontology [26].

Textpresso uses ontologies to fetch hierarchical relationships of classes. It does not represent data in the Semantic Web standards but represents via XML tags, whereas this work proposes an ontology to represent biomedical relation annotations machine-processable data. By relying on Semantic Web standards, such representation enables reasoning and data enrichment from Linked Open Data.

2.2.2. Vapur

Vapur [27] is an inverted-index based search engine for chemical-protein relations that occur in CORD-19 abstracts. Vapur uses BERN [11], specialized deep learning named entity recognition and normalization tool, to identify and normalize entities in abstracts. Later, it extracts relations between identified chemicals and proteins that occur in the same sentence by introducing a fine-tuned BioBERT model. Vapur

extracts binary relations of entities to identify biochemically related molecules, even whose association type cannot be classified under any of the predetermined set of relation classes. Vapur stores data of related molecule triplets and documents as an inverted-index; thus, users can retrieve documents by querying chemical or protein entities and relations.

Vapur uses ontologies to normalize entities and stores extracted data as inverted-index. Our work's domain is similar to Vapur, where both systems annotation target is related biomedical entities occurring in articles. However, our work aims to represent extracted annotations in a machine-processable manner. For this purpose, we proposed an ontology to represent data to enable inference and integration of Linked Open Data sources. This way, users can formulate complex queries containing knowledge that is not extracted.

2.2.3. PubChem

PubChem [28] [29] is a repository of chemical substances. Descriptions, biomedical activities, and biomedical annotations of those substances are stored in a repository. It contains three databases: Substances, BioAssays, and Compounds. Substances database contains information of a chemical compound's name, synonyms, and external identifiers. BioAssay database contains experimental results and experimental descriptions of a chemical. Compounds database contains the chemical structure of compounds. Content covered in PubChem Substances and Compounds databases converted into PubChemRDF [30] with the semantic relationships between compounds and substances. Various standard ontologies are used in PubChemRDF Ontology to achieve interoperability, including Chemical Entities of Biological Interest (ChEBI) and Semanticscience Integrated Ontology (SIO). They published a SPARQL query endpoint to retrieve RDF data [31].

Our work diverges from the PubChemRDF with the representation scope. PubChemRDF represents a curated biomedical entity graph and their relations. They

refer to literature articles if a biomedical entity is mentioned in an article but does not represent relations referred in articles. However, our work represents biomedical entity relations that occurred in literature articles. Our work also demonstrates the reasoning power of ontologies by developing rule sets while PubChemRDF does not benefit from rule sets.

2.2.4. DisGeNET

DisGeNET [32] is a dataset of genes and variants associated with human diseases. Association data is extracted from MEDLINE articles. BeFree [33] text mining tool used to extract gene-disease associations (GDAs). Supporting evidence is explicitly stated for every GDA to make the DisGeNET evidence-based knowledge discovery platform. To comply with the Semantic Web standards and to make data machine-readable, they published data through DisGeNET-RDF [34] [35] linked dataset. For reusability purposes, common vocabularies and ontologies are used in data models such as NCI thesaurus for medical vocabulary and Semanticscience Integrated Ontology (SIO) for general science knowledge. The DisGeNET ontology uses SIO to integrate other linked datasets such as Bio2RDF Linked Data. Finally, DisGeNET-RDF SPARQL endpoint [36] allows querying of RDF Data and linking external sources.

DisGeNET and our work share similar scope and approach of biomedical entity relation representation since both express biomedical interactions occurring in articles with evidence. The reusability of existing ontologies is similar where both works represent biomedical entities from existing ontologies and leverage SIO ontology to represent the article-evidence relationship. However, this work represents types of chemical-protein interactions and chemical-disease interactions while DisGeNET does not represent extracted gene-disease interaction types. Representing interaction type of biomedical entity pair enables various inference rules. While DisGeNET solely benefits from ontology's class hierarchy for inference, our work takes advantage of both ontology and custom rules prepared for proposed ontology.

3. BACKGROUND INFORMATION

This section provides core knowledge about several methods, tools, resources, and services key to understanding ontology-based representation and semantic annotation.

3.1. Ontology

An ontology is a representation method of a particular domain by defining a set of concepts and relationships, and possible constraints about the area of concern [37].

Ontologies consist of classes, properties, relations, and axioms. Ontologies represent knowledge explicitly by defining information in the area; thus, computers can process and reason over data. Furthermore, ontologies are the agreement on common terminology, and it helps data integration between different knowledge sources by keeping standards on data.

3.2. Resource Description Framework

The Resource Description Framework (RDF) [38] [39] is a Semantic Web standard used for modeling and exchanging data for web resources. It provides notations to describe resources and relationships by URIs. An RDF is a directed graph of a triple statement consisting of subject, predicate, and object. Properties that used for expressing a triple statement are *rdf:subject*, *rdf:predicate* and *rdf:object*. While resource URIs can express the whole three components, *rdf:object* can be expressed by a literal value too.

3.3. Web Ontology Language

The Web Ontology Language (OWL) [40] [41] is a language to develop ontologies. Knowledge represented by OWL is logic-based; thus, it can be processed by computers.

An ontology describes classes and axioms which place constraints on sets of individuals with the help of OWL. These logic-based definitions provide reasoning such as class membership, equivalency, consistency of knowledge, and classification.

3.4. SPARQL Protocol and RDF Query Language

SPARQL Protocol and RDF Query Language (SPARQL) [42] [43] is a set of standards that provide query language and protocols to process data stored in RDF format. Specifications of SPARQL are determined by World Wide Web Consortium [37], and it is one of the fundamental technology of the Semantic Web. SPARQL can query data of any RDF Store such as graph databases, triple stores, or knowledge bases.

An example query of SPARQL is shown in Figure 3.1. Example query fetches company's employee information. Results returned for related query shown in Table 3.1.

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?name ?email
WHERE {
    ?employee  a          foaf:Person .
    ?employee  foaf:name  ?name .
    ?employee  foaf:mbox  ?email .
}
```

Figure 3.1. SPARQL query to fetch employee names and emails.

Table 3.1. SPARQL query result for employee names and emails.

?name	**?mail**
Alice	alice@company.com
Bob	bob@company.com

3.5. Turtle

Turtle [44] [45] is a syntax and document format to express the Resource Description Framework data model. It comprises a triple statement's subject, predicate, and object and expresses components through URIs. Turtle is a common alternative to N-Triples, JSON-LD, and RDF/XML to store RDF data.

In Figure 3.2, Turtle syntax states that "Bob" is a person with name "Bob" and with email address of "bob@company.com". Here *ex:bob* is a subject, *foaf:name* is a predicate and literal *"Bob"* is an object.

```
@prefix ex: <http://example.org/>
@prefix foaf: <http://xmlns.com/foaf/0.1/>

ex:bob
    a foaf:Person ;
    foaf:name "Bob";
    foaf:mbox "bob@company.com"
```

Figure 3.2. Turtle representation of Bob's name and email.

3.6. Inference

Inference [46] [47] is the process of discovering new relationships through asserted facts. Asserted facts can be defined as explicit data or known data, whereas inferred facts can be defined as implicit data or interpreted data. On the Semantic Web, data is represented as resource relationships, and inference discovers new relationships between resources by executing a set of rules.

There are mainly two inference sources in the Semantic Web approaches. Ontologies have formal semantics where they enable inference of new data. This kind of inference usually generates the classification of classes and relationships. As well as ontologies, rule sets can be developed for inference. Developing a rule set is similar to logic programming, where a programmer asserts statements and a reasoner solves this asser-

tion. Semantic Web Rule Language (SWRL) is a popular language to develop such rule sets.

Considering an ontology contain simple rule of `(every Cat isA Mammal)`. If this ontology applied to a dataset that contain `(Tom isA Cat)` statement, then `(Tom isA Mammal)` statement can be inferred.

3.7. GraphDB

GraphDB [46] [48] is an efficient, scalable graph database and knowledge discovery tool with RDF and SPARQL support. It consists of three main modules; the Workbench for web-based administration tool, the Engine for query optimization and reasoning, and finally, the Connectors to enable usage of external service.

GraphDB reasoner is based on forward-chaining of entailment rules with the goal of total materialization. Forward-chaining is a reasoning strategy of applying inference rules to asserted facts and performing deductive inference. Due to inferred data calculated on the update of knowledge base, databases that use this strategy perform well at query time. Total materialization is a reasoning goal of applying inference rules to the asserted statements to infer new statements and re-applying the same rules to asserted statements and inferred statements until there is no inferred data.

While GraphDB does not support Semantic Web Rule Language (SWRL), it provides configuration of custom rule sets in its own language similar to SWRL. By configuring a custom ruleset, it is possible to execute special inference rules and axioms.

3.8. Linked Open Data

One of the Semantic Web objectives is to make data accessible and reusable. Semantic Web technologies solve this problem by connecting different data sources

through standardized formats. Collection of connected datasets on the web referred as Linked Open Data (LOD) [49] [50].

One of the most significant data sources of Linked Open Data is Wikidata which is a collaboratively edited knowledge graph with over 96 million resources [51] [52]. Finally, Wikidata exposes a SPARQL endpoint [53] and allows reusability of data.

3.9. Bidirectional Encoder Representations from Transformers

Bidirectional Encoder Representations from Transformers (BERT) [54] is a machine learning technique based on transformers with state-of-the-art results for most of the natural language processing tasks. It has been developed to solve limited context capturing problems of unidirectional encoding techniques.

BERT pre-trained on two tasks; masked language modeling, where random tokens are masked for token prediction, and next sentence prediction. BERT is a fine-tuning based representation model that can achieve state-of-the-art results for specific tasks by fine-tuning.

3.10. Utilized Ontologies

3.10.1. Chemical Entities of Biological Interest

Chemical Entities of Biological Interest (ChEBI) [55] is a molecular entity ontology that focuses on small chemical compounds. There are no proprietary data in ontology, and it is publicly accessible. ChEBI contains ontological classification where classes of entities and their parents are specified. Finally, ChEBI contains 59k annotated compounds [56].

Several data sources combined to create ChEBI, such as *ChEMBL* and *IntEnz*. Every item in ChEBI explicitly referenced to the source ontology.

3.10.2. National Cancer Institute Thesaurus

NCI Thesaurus (NCIt) [57] is an ontology describing the cancer domain, including cancer-related diseases, findings, and abnormalities. It is a recognized standard for biomedical coding and reference, and it covers terminology for clinical care, primary research, and public information.

There are more than 12k types defined in NCIt as amino acid, peptide, or protein class [58].

3.10.3. Medical Subject Headings

The Medical Subject Headings (MeSH) [59] thesaurus is a hierarchically-organized vocabulary that focuses on biomedical and health-related information. It is maintained by the National Library of Medicine [60].

MeSH contains approximately 80k terms, and it is updated annually [61].

3.10.4. Molecular Interactions

Molecular Interactions(MI) [62] is a controlled vocabulary for the annotation of complex biological interactions.

3.10.5. Semanticscience Integrated Ontology

The Semanticscience Integrated Ontology (SIO) [63] is an upper-level ontology to describe types and relations for knowledge representation of arbitrary objects, processes, and their attributes in the domains of chemistry, biology, biochemistry, and bioinformatics.

SIO provides a vocabulary for Bio2RDF and SADI projects, and it is freely available for users.

3.11. Ontology Prefixes

Prefixes are used in this thesis to refer to various ontologies and data namespaces with the format of "prefix:" Each prefix and corresponding namespace listed in Table 3.2. In this thesis, prefixes are used to shorten URIs.

Table 3.2. Prefixes used in the thesis and their corresponding namespaces.

Prefix	**Namespace**
bee	http://soslab.cmpe.boun.edu.tr/ontologies/bee
obo	http://purl.obolibrary.org/obo/
ncit	http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl
sio	http://semanticscience.org/resource/
wd	http://www.wikidata.org/entity/
wdt	http://www.wikidata.org/prop/direct/
xsd	http://www.w3.org/2001/XMLSchema
rdf	http://www.w3.org/1999/02/22-rdf-syntax-ns
rdfs	http://www.w3.org/2000/01/rdf-schema

4. MODEL

This chapter presents an ontology developed to represent biomedical relations in biomedical scientific articles. This ontology, named Biomedical Entities Evidence (BEE), introduces the concepts for representing chemical-protein and chemical-disease relations in scientific articles.

The following sections describe the overall process of article annotation and semantic representation of biomedical entity relations and how the ontology was designed as well as the details of the concepts and properties it covers.

4.1. Overall Process of Semantic Representation

Overall process of semantic representation of biomedical entity relations in biomedical articles consists of three modules: Preprocessing, Representation, Semantic Application. In Figure 4.1 three modules of the overall process can be seen. Set of unstructured articles fed into Preprocessing module and biomedical entity relation annotations are extracted. An article consists of title, abstract, digital object identifier (DOI), and publish date. It is unstructured because the article's data is not modeled to be processed by computers. In this context, annotation is a representation of biomedically related entities with the information of linked concepts and the provenance information of entities. Annotations collected in first module fed into Representation module to be expressed semantically in machine-processable manner based on Biomedical Entities Evidence (BEE). Lastly, the Semantic Application module allows users to benefit from semantic annotations by reasoning on ontologically represented data and integrating Linked Open Data [50].

The aim of the Preprocessing and the Representation modules is to represent articles semantically. In comparison, the aim of the Semantic Application module is to favor semantically represented data by leveraging the Semantic Web approaches. In

further sections, approaches of modules are described in detail.

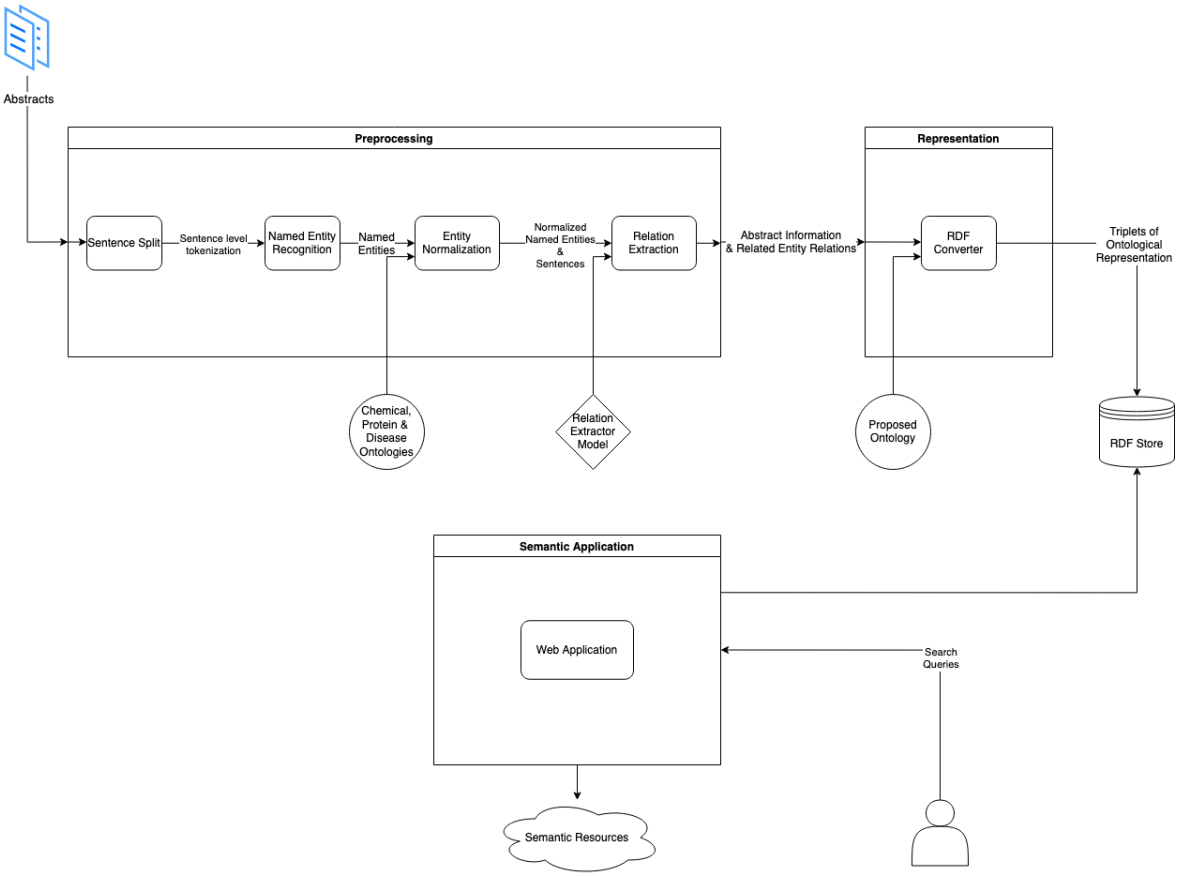

Figure 4.1. Overall process of ontology based representation of annotations.

4.1.1. Preprocessing

The task of Preprocessing module is to identify and link biomedical entities within articles to ontological concepts and extract relation types of biomedical entities in the same sentences. As a result, the collection of unstructured articles is transformed into structured annotations. Algorithm 4.3 summarizes the processing phase for a given set of unstructured articles. A scientific article consists of title, abstract, digital object identifier(DOI), and publish date in this work. Annotation of an article has four steps: sentence splitting, named entity recognition, named entity normalization, and relation extraction. Those steps are processed sequentially since outputs of a phase are the

inputs of the next phase.

The first step, sentence splitting, is the system's starting point. As stated before, this work represents relations of biomedical entities that occur in the same sentence. Due to this constraint, abstracts of articles are split into sentences and processed sentence by sentence in further steps. For a set of articles `A` and an article `a` from the set, `sentences` function returns the sentences of `a` (Line 9).

The second step identifies named biomedical entities that referred in sentences of an article. While sentences contain many named entities types, this work is only interested in chemicals, proteins, and diseases. For a given sentence `s`, a set of identified named entities are returned from `entities` function (Line 11). A named entity `e`, contains the exact location of the entity in the sentence and the assigned type. Any entity assigned other than chemical, protein, or disease is ignored and not returned from the function.

The third step normalizes identified entities to unique ontology concepts. `TO` being the target ontology, contains a set of concepts. A concept consists of a unique identifier and label. Function `normalize(entities[s], TO)` normalizes set of entities into concepts of target ontology and returns set of normalized entities `ne` (Line 14). Every element of `ne` consist of identifier and label of the ontology concept addition to `e`. Each biomedical entity type has its own target ontology since it will be represented in the semantic representation phase by a unique ontology concept. Entities that could not be normalized to any ontology concept are ignored, and the algorithm continues with normalized entities (Line 15).

The last step extracts the relations of entities that occur in the same sentence. Each chemical-protein and chemical-disease entity pair of a sentence has potential to be related; thus, firstly, every possible pair created (Lines 18-19). `extractRelations(s, chemProtPairs, chemDisPairs)` returns list of related pairs with assigned relation type (Line 20). Every related pair `rp` consist of identifiers of entities and relation type.

The assigned relation type varies according to the type of each pair. While chemical-protein pairs are assigned to a set of relations, chemical-disease pairs are assigned to binary relations. Details of relation types are described in Section 4.1.2.

Set of annotation data generated after identification and normalization of named entities and extraction of relations between entities. This data generated from article information, normalized entity data and extracted relations by `createAnnotations` function (Line 22). Figure 4.2 shows an example of Preprocessing module's annotation output in JSON format. Article *PMC3537594* processed through steps of Preprocessing. As shown in the listing, output data consist of information about the article such as DOI, publish date, title, URI, and body information, as well as entity pair relations. Annotation of chemical-protein relation can be seen between Lines 6-21. Here a relation identified between chemical *"lyrocine"* which is normalized to *CHEBI_6601* concept and protein *"HDAC"* which is normalized to *C16682* concept. Location information of identified entities are attached as *startPos* and *endPos* variables. Relation type between entities extracted as inhibition(*CPR:4*). Likewise, identified chemical-disease relation can be found between Lines 22-37.

```
"doi": "10.1186/1475−2867−12−49",
"uri": "https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3537594/",
"title": "Lycorine induces cell−cycle arrest in the G0/G1...",
"body": "Lycorine, a natural alkaloid ...",
"publishedAt": "20120923",
"chemProtRelations": [{
   "chemical": {
     "ontologyId": "CHEBI_6601",
     "ontologyText": "lycorine",
     "text": "lycorine",
     "startPos": 162,
     "endPos": 170 },
   "protein": {
     "ontologyId": "C16682",
     "ontologyText": "histone deacetylase",
     "text": "HDAC",
     "startPos": 55,
     "endPos": 59 },
   "relationType": "CPR:4",
   "sentenceIndex": 16
 }],
"chemDisRelations": [{
   "chemical": {
     "ontologyId": "CHEBI_6601",
     "ontologyText": "lycorine",
     "text": "lycorine",
     "startPos": 48,
     "endPos": 56 },
   "disease": {
     "ontologyId": "D059447",
     "ontologyText": "cell cycle checkpoints",
     "text": "cell−cycle arrest",
     "startPos": 98,
     "endPos": 115 },
   "relation_type": true,
   "sentence_index": 12
 }]
```


Figure 4.2. Annotated article.

```
1: Input: A                                                          ▷ article set
2: Input: TO                                          ▷ target ontologies to normalize
3: Output: O                                                  ▷ annotated article set
4: for each a in A do
5:     sentences ← []                                          ▷ sentences of an article
6:     entities ← []                                  ▷ identified biomedical entities
7:     ne ← []                                        ▷ normalized biomedical entities
8:     ne' ← []                  ▷ identified and normalized biomedical entities
9:     sentences ← sentences(a)                                      ▷ split sentences
10:    for each s in sentences do
11:        entities[s] ← entities(s)                  ▷ identify biomedical entities
12:    end for
13:    for each s in sentences do
14:        ne[s] ← normalize(entities[s], Target Ontologies)     ▷ normalize entities
15:        ne'[s] ← entities[s] ∩ ne[s]          ▷ continue with normalized entities
16:    end for
17:    for each s in sentences do
18:        chemProtPairs ← createChemicalProteinPairs(ne'[s])
19:        chemDisPairs ← createChemicalDiseasePairs(ne'[s])
20:        relations[s] ← extractRelations(s, chemProtPairs, chemDisPairs)
21:    end for
22:    O[a] ← createAnnotations(a, ne', relations)
23: end for
24: return O
```

Figure 4.3. Algorithm of preprocessing for set of articles.

4.1.2. Representation

The Representation module converts set of biomedical entity relation annotations of an article (represented in Figure 4.2 with JSON) into semantic representation based on BEE. The output data format is the Resource Description Framework (RDF), a standard model for data exchanging on the Web.

As Kiryakov et al. stated [64], to represent an annotation semantically, there are two main prerequisites. Related entities that are linked to their semantic concepts and an ontology that define the domain classes and their relations. Section 4.1.1 described the process of capturing related entities and corresponding ontology concepts. For the second prerequisite, Biomedical Entities Evidence(BEE) ontology has been proposed and described in Section 4.2; thus, biomedical entity relations in articles can be represented semantically.

Annotations collected in the previous module already contain corresponding ontology concepts of biomedical entities. However, biomedical relation types are extracted from articles but not matched to any ontology concept. Biomedical relations and their ontology concepts are mapped beforehand since relation types are limited and do not need automation to normalize. As stated before, this work identifies only types of chemical-protein relations while chemical-disease relations are considered as a binary relation. Chemical-protein relation types are expressed by Molecular Interactions(MI) ontology. Extracted relation types of chemical-protein and their mappings to ontology concepts are shown in Table 4.1.

In RDF, triple components are expressed by URIs except for value literals. Biomedical articles already exist over the Web; thus, articles are expressed by their unique URIs. An example of an article insertion statement has shown in Listing 4.4. Here, article *PMC3537594* represented with resource URI of *https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3537594*. "pmc:" prefix used to shorten the URI. In Figure 4.4, title, DOI, publish date and type of related article stored as RDF triples.

Table 4.1. The mappings of extracted chemical-protein relations (CPR) to the concepts in the Molecular Interactions (MI) ontology.

Relation Name	**Extracted Relation**	**Ontology Concept**
Activation	CPR:3	MI:0624
Inhibition	CPR:4	MI:0623
Agonist	CPR:5	MI:0625
Antagonist	CPR:6	MI:0626
Substarate	CPR:9	MI:0502
Cofactor	CPR:8	MI:0682
Regulator	CPR:2	MI:2274

```
PREFIX bee: <http://soslab.cmpe.boun.edu.tr/ontologies/bee#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX pmc: <https://www.ncbi.nlm.nih.gov/pmc/articles#>
INSERT {
        pmc:PMC3537594  rdf:type bee:Article
                        bee:hasTitle 'Lycorine induces cell-cycle..' ;
                        bee:hasFriendlyId '10.1186/1475-2867-12-49' ;
                        bee:publishedAt '20120923'^^xsd:dateTime .
}
```

Figure 4.4. Insertion statement of an article.

Chemicals, proteins, diseases, and chemical-protein relations are expressed in RDF triple format by their corresponding ontology URIs to link entities to concepts shared across the Web. For instance, *"lyrocine"* is a chemical with corresponding ontology class of *CHEBI_6601* from ChEBI ontology; thus, it represented in RDF triples by *http://purl.obolibrary.org/obo/CHEBI_6601* URI. To shorten the queries, "obo:" prefix is used. As a reminder, chemical-disease relations are considered as binary relations, so they do not include a relation type that is linked to an ontology class.

Unique identifiers are generated to represent entity instances created in the Pre-processing module and do not have any existing resource over the Web. Instances of *bee:RelationEvidence*, *bee:Evidence* and *bee:BiomedicalEntityPairRelation* classes are

this kind of instances. Every instance of *bee:RelationEvidence* and *bee:Evidence* have random unique identifier. Figure 4.5 shows the insertion example of relation evidence and its evidences. Here a random identifier is created at the application level and injected for *bee:RelationEvidence* into the insertion query. For *bee:Evidence* instances, random identifiers are created at the database level.

```
PREFIX bee: <http://soslab.cmpe.boun.edu.tr/ontologies/bee#>
PREFIX rdf: <http://www.w3.org/1999/02/22−rdf−syntax−ns#>
PREFIX pmc: <https://www.ncbi.nlm.nih.gov/pmc/articles#>
PREFIX sio: <http://semanticscience.org/resource/>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

INSERT {
        bee:523fb0b4−8838−4cfb−9c2c−aa9b6fcb61a8
                rdf:type bee:RelationEvidence ;
                bee:hasProteinEvidence ?protRef ;
                bee:hasChemicalEvidence ?chemRef .

        pmc:PMC3537594
                sio:SIO_000772 bee:523fb0b4−8838−4cfb−9c2c−aa9b6fcb61a8 .

        ?protRef    rdf:type bee:Evidence ;
                bee:hasStartPosition 55^^xsd:nonNegativeInteger ;
                bee:hasEndPosition 59^^xsd:nonNegativeInteger ;
                bee:hasSentenceIndex 16^^xsd:nonNegativeInteger ;
                bee:hasTextRepresentation 'HDAC'^^xsd:string .

        ?chemRef    rdf:type bee:Evidence ;
                bee:hasStartPosition 162^^xsd:nonNegativeInteger ;
                bee:hasEndPosition 170^^xsd:nonNegativeInteger ;
                bee:hasSentenceIndex 16^^xsd:nonNegativeInteger ;
                bee:hasTextRepresentation 'lycrocine'^^xsd:string .
        WHERE {
           BIND (IRI(CONCAT(bee:, strUUID())) AS ?chemRef) .
           BIND (IRI(CONCAT(bee:, strUUID())) AS ?protRef) .
        }
}
```

Figure 4.5. Insertion statement of a relation evidence.

An instance of a *bee:BiomedicalEntityPairRelation* consists of two related entities and a relation type. Thus same biomedical entities with the same relationship type refer to the same biomedical entity pair instance. Creating a unique identifier for a biomedical entity pair relation depends on referred entities and their relationship type. For protein instance p_i, chemical instance c_i, relation type instance r_i, and biomedical

relation of instances $(c_i \leftrightarrow p_i : r_i)$ then $(c_1 \leftrightarrow p_1 : r_1) \neq (c_1 \leftrightarrow p_1 : r_2)$.

An example of chemical-protein insertion statement given in Figure 4.6 where, chemical *"Lycorine"* and protein *"Histone Deacetylase"* have *inhibition*(*CPR:4*) relation in article *PMC3537594*. Identifier of the *bee:ChemProtRelation* instance has been generated and injected into statement which is *bee:51af0d94469622b03a64279ed0ed213b*. This same biomedical entity pair relation can exist across multiple articles, and every representation uses the same instance. If the same biomedical entities have a different relation type, then a new pair relation instance is created.

Figure 4.7 shows an example chemical-disease insertion statement where no relation type declared since it is a binary relation.

```
PREFIX bee: <http://soslab.cmpe.boun.edu.tr/ontologies/bee#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX obo: <http://purl.obolibrary.org/obo/>
PREFIX ncit: <http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#>

INSERT {
        bee:51af0d94469622b03a64279ed0ed213b
                rdf:type bee:ChemicalProteinRelation ;
                bee:hasChemical obo:CHEBI_6601 ;
                bee:hasProtein  ncit:C16682;
                bee:hasRelationType obo:MI_0623 .

        bee:523fb0b4-8838-4cfb-9c2c-aa9b6fcb61a8
                bee:hasChemProtRelation bee:51af0d94469622b03a64279ed0ed213b .
}
```

Figure 4.6. Insertion statement of an chemical-protein inhibition relation.

```
PREFIX bee: <http://soslab.cmpe.boun.edu.tr/ontologies/bee#>
PREFIX rdf: <http://www.w3.org/1999/02/22−rdf−syntax−ns#>
PREFIX obo: <http://purl.obolibrary.org/obo/>
PREFIX mesh: <http://purl.bioontology.org/ontology/mesh/>

INSERT {
        bee:991fdb927258e154cbd1a5ee5909def0
                rdf:type bee:ChemicalDiseaseRelation ;
                bee:hasChemical obo:CHEBI_6601 ;
                bee:hasDisease  mesh:D059447.

        bee:08294a9a−3753−4fad−9c23−1ba59c1851a5
                bee:hasChemDisRelation bee:991fdb927258e154cbd1a5ee5909def0 .
}
```

Figure 4.7. Insertion statement of an chemical-disease relation.

4.2. Biomedical Entity Evidence Ontology

Biomedical Entity Evidence(BEE) ontology has been designed to represent related chemical-protein and related chemical-disease entities within an article with relation's evidence. Evidence of a biomedical relation states the position of entities in the article to help the researcher assess the result. A biomedical entity could be represented with different surface forms in the article, or an article could result from a query because of an inferred data; thus, displaying entities in article helps researchers to evaluate the results clearer if biomedical entities' evidence is expressed.

BEE ontology defines concepts to express biomedical entities and biomedical entity relations within articles. It is also designed to connect to Linked Open Data and enrich represented information with existing domain knowledge. For both ontology reusability and interoperability, BEE uses existing ontologies. The Chemical Entities of Biological Interest [55] ontology describes chemical entities. The National Cancer Institute Thesaurus [57] ontology refers to protein entities. Medical Subject Headings [59] ontology used to express diseases. While chemical-disease relations are captured as binary relations and do not express any relation type, the Molecular Interactions [62] ontology is used to express chemical-protein relation types. Finally, Semanticscience Integrated Ontology [63] ontology provides various object properties that enable the expression between articles and biomedical relations.

The visualization of the BEE ontology shown in the Figure 4.8. While rounds represent ontology classes, squares represent data types of literals. Object properties and data properties are represented via arrows. Dashed arrows represent *rdfs:subClassOf* relation, where the head of the arrow points to the parent class.

In further sections, classes of BEE ontology and relations of classes have been explained and for simplicity of explanation, whole ontology graph in Figure 4.8 has been separated into subgraphs in each section. The prefix *bee:* has been used to refer to the BEE namespace.

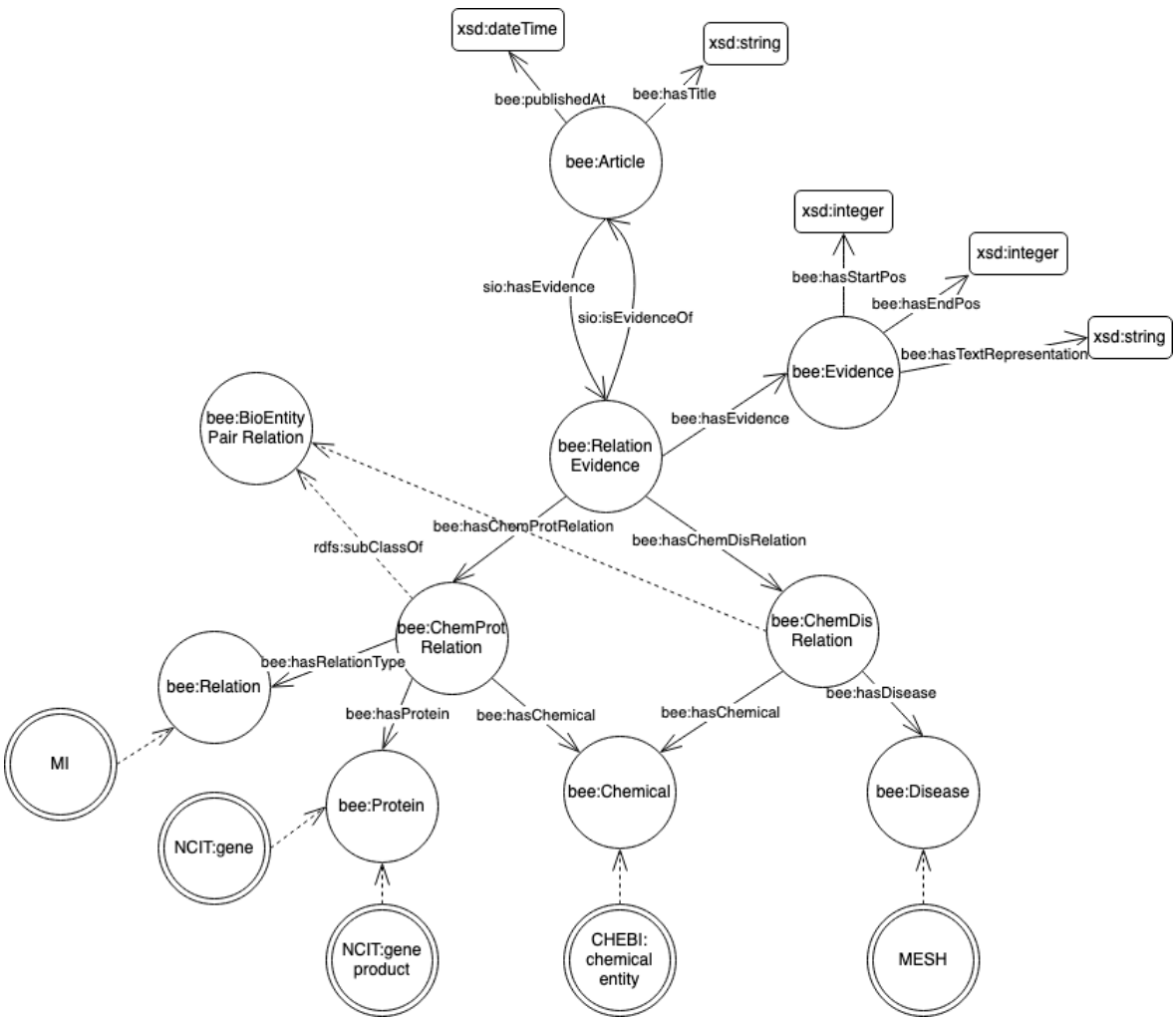

Figure 4.8. The main classes and properties of BEE ontology.

In the development of BEE ontology, seven steps of Ontology Development 101 [16] applied step by step as described below.

(i) *Determine the domain and scope of the ontology:* Proposed BEE ontology represents biomedical entity pair relations in scientific articles with evidence. Scope of biomedical entity pair relations are chemical-protein and chemical-disease relations. The evidence of pairs are descriptors of entities in articles. While a chemical-disease relation can be binary relation, a chemical-protein relation can be any predefined biomedical relation.

(ii) *Consider reusing existing ontologies:* Biomedical ontology development is a popular area of research, and there are various well-defined ontologies in the area. Reusing existing ontologies increases standardization of data and increases interoperability of data sources. Chemical Entities of Biological Interest [55] ontology is used to describe chemical entities. National Cancer Institute Thesaurus [57] ontology used to express protein entities. Medical Subject Headings [59] ontology used to represent disease entities. For chemical-protein relations, Molecular Interactions [62] ontology used. As well as biomedical entities, Semanticscience Integrated Ontology [63] used to represent object properties of various components.

(iii) *Enumerate important terms in the ontology:* Terms of BEE ontology specify biomedically related entities in scientific articles with evidences. These terms are article, evidence, chemical-protein relation, chemical-disease relation, chemical, protein, disease, relation type.

(iv) *Define the classes and the class hierarchy:* For enumerated terms, classes and class hierarchies are defined with a bottom-up approach. The bottom-up approach defines most specific classes first and defines parent classes later. For example, first, *bee:ChemProtRelation* and *bee:ChemDisRelation* classes are defined, and later, *bee:BiomedicalEntityPairRelation* class is defined as a parent class to share common features of child classes. Figure 4.8 shows classes and class hierarchies. Dashed arrows represent class hierarchy, where the arrow's head points to the parent class.

(v) *Define the properties of classes:* This step describes the internal structure of

classes. A class has object properties to describe relationships with other classes and data properties to describe its data. The following are several examples of object properties and data properties. The *bee:Article* has evidences, and it contains *sio:hasEvidence* object property, which ranges to *bee:RelationEvidence*. Likewise, *bee:Article* have a title which described with *bee:hasTitle* that ranges to *xsd:string* data value.

(vi) *Define the facets of the slots:* Properties of a concept that describe features of the concepts is called the slot, and facets of the slots are restrictions of concept features. An ontology describes several facets of slots like cardinality, allowed values, and domain and range of a concept. Cardinality defines the number of values a class can have. Cardinality features of BEE ontology are listed below.

- *bee:RelationEvidence* can have exactly one *bee:BiomedicalEntityPairRelation*
- *bee:RelationEvidence* can have exactly two *bee:Evidence*
- *bee:ChemProtRelation* can have exactly one *bee:Protein* and exactly one *bee:Chemical*
- *bee:ChemDisRelation* can have exactly one *bee:Disease* and exactly one *bee:Chemical*

(vii) *Create instances:* Instances of BEE classes are created using the protoype described in Section 5.

Representation of an article through BEE ontology shown in Figure 4.9. Article's annotation data can be seen in Section 4.1.1. As shown in listing, article contain one chemical-protein relation with *inhibition* relation type and one chemical-disease relation.

```
@prefix bee:  <http://soslab.cmpe.boun.edu.tr/ontologies/bee#>
@prefix sio:  <http://semanticscience.org/resource/>
@prefix xsd:  <http://www.w3.org/2001/XMLSchema#>
@prefix obo:  <http://purl.obolibrary.org/obo/>
@prefix ncit: <http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#>
@prefix mesh: <http://id.nlm.nih.gov/mesh/>

<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3537594/>
    a bee:Article;
    bee:hasTitle "Lycorine induces cell−cycle arrest in the G0/G1 phase in K562
     cells via HDAC inhibition"^^xsd:string;
    bee:publishedAt "2012−09−03"^^xsd:dateTime;
    bee:hasFriendlyId "10.1186/1475−2867−12−49"^^xsd:string;
    sio:'has evidence'
    [
        a bee:RelationEvidence;
        bee:hasChemicalEvidence [
            a bee:Evidence;
            bee:startPosition "162"^^xsd:nonNegativeInteger;
            bee:endPosition "170"^^xsd:nonNegativeInteger;
            bee:sentenceIndex "16"^^xsd:nonNegativeInteger;
            bee:textRepresentation "lycorine"^^xsd:string;
        ];
        bee:hasProteinEvidence [
            a bee:Evidence;
            bee:startPosition "55"^^xsd:nonNegativeInteger;
            bee:endPosition "59"^^xsd:nonNegativeInteger;
            bee:sentenceIndex "16"^^xsd:nonNegativeInteger;
            bee:textRepresentation "HDAC"^^xsd:string;
        ];
        bee:hasChemProtRelation bee:51af0d94469622b03a64279ed0ed213b .
    ];
```

Figure 4.9. BEE Ontology representation of an article.

```
    sio:'has evidence'
    [
        a bee:RelationEvidence;
        bee:hasChemicalEvidence [
            a bee:Evidence;
            bee:startPosition "48"^^xsd:nonNegativeInteger;
            bee:endPosition "56"^^xsd:nonNegativeInteger;
            bee:sentenceIndex "12"^^xsd:nonNegativeInteger;
            bee:textRepresentation "lycorine"^^xsd:string;
        ];
        bee:hasDiseaseEvidence [
            a bee:Evidence;
            bee:startPosition "830"^^xsd:nonNegativeInteger;
            bee:endPosition "832"^^xsd:nonNegativeInteger;
            bee:sentenceIndex "12"^^xsd:nonNegativeInteger;
            bee:textRepresentation "cell–cycle arrest"^^xsd:string;
        ];
        bee:hasChemDisRelation bee:da8f688df8bf4a8a926e1642ea96b63c .
    ];

bee:51af0d94469622b03a64279ed0ed213b
    a bee:ChemProtRelation;
    bee:hasChemical     obo:CHEBI_6601;
    bee:hasProtein      ncit:C16682;
    bee:hasRelationType obo:MI_0623 .

bee:da8f688df8bf4a8a926e1642ea96b63c
    a bee:ChemDisRelation;
    bee:hasChemical     obo:CHEBI_6601;
    bee:hasDisease      mesh:D059447 .
```

Figure 4.9. BEE Ontology representation of an article. (cont.)

4.2.1. Article

The *bee:Article* class describes scientific articles. An article consists of an abstract, a title, and a publish date. BEE ontology represents biomedical relations occurring in articles and their evidence. Relation evidences of an article represented by *sio:hasEvidence* object property. This property's range is *bee:RelationEvidence* class with no cardinality restriction. Title and publish date are represented with respectively *bee:hasTitle* and *bee:publishedAt* data properties. *bee:hasTitle* range is *xsd:string* and *bee:publishedAt* range is *xsd:dateTime*.

In Figure 4.10, article class and relation to other classes, and data properties can be seen. In Table 4.2 descriptions of article class and relationships are shown.

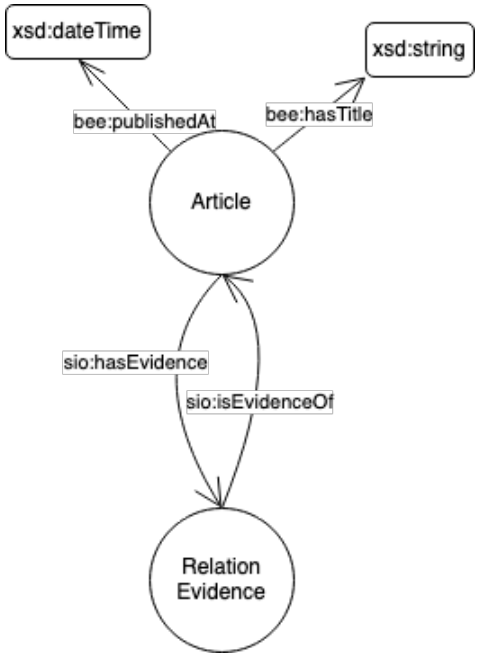

Figure 4.10. *bee:Article* properties and related classes.

Table 4.2. Description of article class and properties.

Term	**Type**	**Description**
bee:Article	Class	Article is an scientific document published in literature
sio:hasEvidence	Object Property	Describes that article refers to a biomedical entity pair relation
bee:publishedAt	Data Property	Specifies the publish date of the article
bee:hasTitle	Data Property	Specifies the title of the article

4.2.2. Relation Evidence

Articles consist many types of claims. Biomedical entity relations are one type of claims that are represented by BEE ontology. BEE ontology represents such claims with their provenance information which is called as *bee:Evidence* in this domain. The *bee:RelationEvidence* class represents the relationship between an article, a related biomedical entity pair, and mentioned entities' evidence. Figure 4.11 describes the

relation evidence class and its relations. In the Table 4.3 descriptions of relation evidence class and its relations are shown.

An article consists of many relation's evidence; however, a relation's evidence can belong to only one article because it represents the entity's location in the related article. Hence there is one to many relation between *bee:Article* and *bee:RelationEvidence* classes. *sio:isEvidenceOf* object property represents the relationship of *bee:RelationEvidence* class to *bee:Article* class and it is the inverse property of *sio:hasEvidence* defined in Section 4.2.1.

As BEE ontology represents biomedical entity pairs in articles, a relation evidence of an article refers to exactly two *bee:Evidence* classes via *bee:hasEvidence*'s child object property. *bee:hasEvidence* class has three child object properties that explicitly states the entity's type which are *bee:hasChemicalEvidence*,*bee:hasProteinEvidence* and *bee:hasDiseaseEvidence*. All three properties's domain is *bee:RelationEvidence* and range is *bee:Evidence*.

The *bee:RelationEvidence* class also have relationship with any child class of *bee:BiomedicalEntityPairRelation* class. Child classes are, *bee:ChemProtPairRelation* or *bee:ChemDisPairRelation*. Such relation provided by *bee:hasChemProtRelation* or *bee:hasChemDisRelation* object properties. A relation evidence instance has to consist of precisely one pair relation instance.

Table 4.3. Description of relation evidence class and properties.

Term	**Type**	**Description**
bee:RelationEvidence	Class	Relation evidence refers to related entity pair and its evidences
sio:isEvidenceOf	Object Property	Describes that relation evidence instance is the evidence of an article
bee:hasEvidence	Object Property	Indicates that evidence belongs to this relation evidence instance
bee:hasChemicalEvidence	Object Property	Indicates that evidence mentions a chemical
bee:hasProteinEvidence	Object Property	Indicates that evidence mentions a protein
bee:hasDiseaseEvidence	Object Property	Indicates that evidence mentions a disease
bee:hasChemProtRelation	Object Property	Indicates that related entity pairs are chemical and protein
bee:hasChemDisRelation	Object Property	Indicates that related entity pairs are chemical and disease

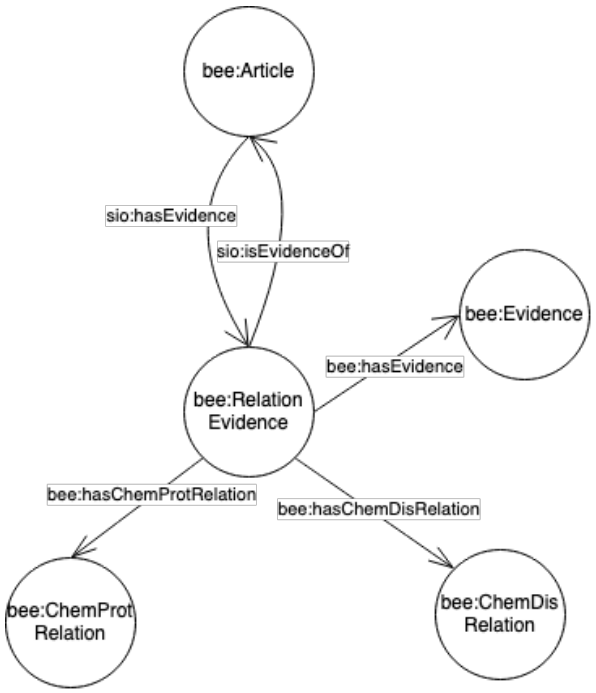

Figure 4.11. *bee:RelationEvidence* properties and related classes.

4.2.3. Evidence

Articles represented based on BEE ontology consists of biomedical entity relations claims. *bee:Evidence* class represents provenance of this claims which referred in the article. An evidence instance consists of referred entity's surface form, starting position, and ending position in the article. This data represented; thus, referred entities can be presented to the users.

Figure 4.12 shows the evidence class and its relations. Moreover, in the Table 4.4 evidence class and associated properties are described.

4.2.4. BiomedicalEntity

bee:BiomedicalEntity is a parent class of entities referred in articles. There are three child classes which are detailed below.

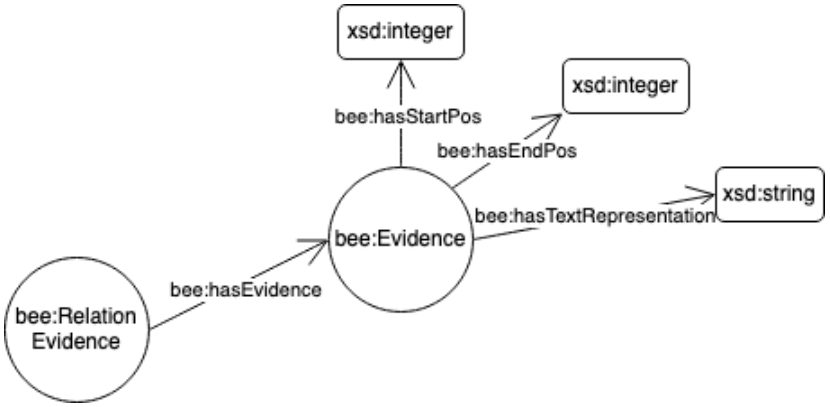

Figure 4.12. *bee:Evidence* properties and related classes.

Table 4.4. Description of evidence class and properties.

Term	**Type**	**Description**
bee:Evidence	Class	Evidence of the mentioned entity in article
bee:hasStartPos	Data Property	Describes the start position of mentioned entity
bee:hasEndPos	Data Property	Describes the end position of mentioned entity
bee:hasTextRepresentation	Data Property	Specifies the surface form of mentioned entity in article

4.2.4.1. Chemical. Chemicals represented by this class. *bee:Chemical* have *ChEBI: chemical entity* as subclass for reusability of ChEBI classes.

4.2.4.2. Protein. Proteins represented by this class. *bee:Protein* have *NCIT:gene* and *NCIT:gene product* as subclasses for reusability of NCIt classes.

4.2.4.3. Disease. Diseases represented by this class. MeSH ontology classes used as subclass of *bee:Disease* for reusability of MeSH classes.

4.2.5. RelationType

bee:RelationType represents relationship type of a biomedical entity pair. It has child classes from Molecular Interactions [62] ontology for reusability of existing ontologies. These classes covered in Section 4.1.2.

4.2.6. Biomedical Entity Pair Relation

This work represents related chemical protein pairs and related chemical disease pairs. While those pairs have different properties, such as relation type of the pair, both are child class of *bee:BiomedicalEntityPairRelation* class. This class consists of two biomedical entities and optionally a relation type that describes the relationship of entities. It has two child classes that represent related entities explicitly by their types which are *bee:ChemProtRelation* and *bee:ChemDisRelation*.

4.2.6.1. ChemProtRelation. *bee:ChemProtRelation* class represents the relationship between a chemical entity and a protein entity. Figure 4.13 shows the related classes and properties of *bee:ChemProtRelation*.

bee:ChemProtRelation class consist of exactly one *bee:Protein* and exactly one *bee:Chemical* classes. This relationships respectively provided by *bee:hasProtein* and *bee:hasChemical* object properties.

As described before, two biomedical entities of this class are biomedically related, and this relationship is provided by *bee:hasRelationType* object property.
bee:ChemProtRelation have to have exactly one *bee:Relation* class. All three object properties range to external ontologies in the implementation as shown in Figure 4.13 to improve interoperability of the ontology.

Relation type between chemical and protein entities has been represented as object property of *bee:ChemProtRelation* other than object property between *bee:Protein*

and *bee:Chemical* classes considering that there could be several different relation types between same protein and chemical instances. BEE ontology represents such pair relations as different relation instances because BEE does not represent biomedical entity relations as facts, but it represents as claims. As a result of scientific literature nature, facts could change over time, and there could be contradictions.

In the Table 4.5 related classes and associated properties of *bee:ChemProtRelation* class are described.

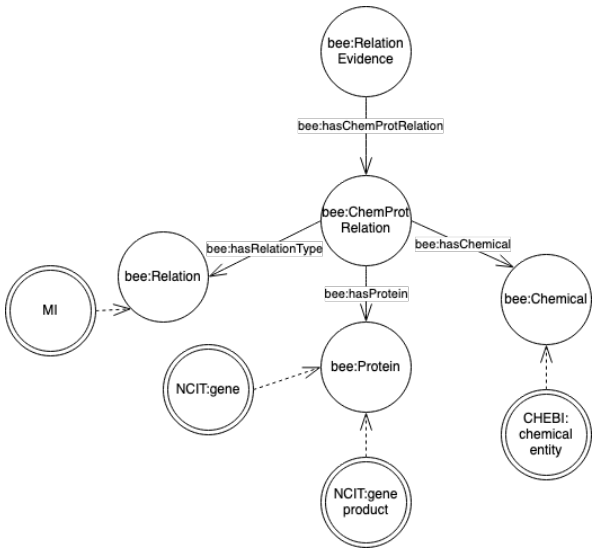

Figure 4.13. *bee:ChemProtRelation* properties and related classes.

Table 4.5. Description of ChemProtRelation class and properties.

Term	**Type**	**Description**
bee:ChemProtRelation	Class	Chemical-Protein relation
bee:hasChemical	Object Property	Indicates the chemical entity of pair
bee:hasProtein	Object Property	Indicates the protein entity of pair
bee:hasRelationType	Object Property	Indicates the relation type of pair

4.2.6.2. ChemDisRelation. *bee:ChemDisRelation* class represents relationship between a chemical and a disease. Figure 4.14 shows the *bee:ChemDisRelation* and its relations.

bee:ChemDisRelation class does not contain any object property for relation type as this work represents only binary relation between chemical entities and disease entities. An instance of *bee:ChemDisRelation* already shows the existence of a relation between identified entities. *bee:ChemDisRelation* consist of exactly one *bee:Chemical* class and exactly one *bee:Disease* class and relationships respectively provided by *bee:hasChemical* and *bee:hasDisease* object properties. In the Table 4.6 related classes and associated proper-ties of *bee:ChemDisRelation* class are described.

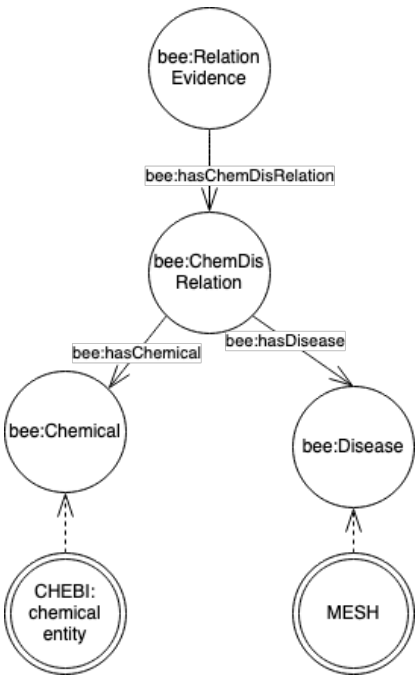

Figure 4.14. *bee:ChemDisRelation* properties and related classes.

Table 4.6. Description of ChemDisRelation class and properties.

Term	**Type**	**Description**
bee:ChemDisRelation	Class	Chemical-Disease relation
bee:hasChemical	Object Property	Indicates the chemical entity of pair
bee:hasDisease	Object Property	Indicates the disease entity of pair

5. IMPLEMENTATION

This chapter presents the prototype implementation of ontology-based representation and utilization of semantic annotations. Implementation approach based on modules that described in Section 4.1. In order to annotate biomedical articles, a preprocessing module was implemented, which identifies and normalizes chemical, protein, disease entities and extracts relations of chemical-protein pairs, and chemical-disease pairs occur in the same sentences. The annotations are semantically represented based on Biomedical Entities Evidence(BEE) ontology. Semantic data is processed by a semantic web application that utilizes Linked Open Data sources.

The following sections describe the implementation details of preprocessing pipeline, representation approach, and semantic web application.

5.1. Preprocessing

The implementation of preprocessing is based on the algorithm in Figure 4.3. This module consists of four steps: sentence splitting, named entity recognition, named entity normalization, and relation extraction. After preprocessing module, article annotation data is fed into the next module, which is RDF Converter to represent and store data.

5.1.1. Sentence Splitting

SciSpacy [65] has been used for sentence splitting task and named entity recognition task. Additionally, it is also used for the named entity normalization task of diseases since it has a built-in normalization module to MeSH ontology which is the target ontology of diseases in this work.

The sentence segmentation task has several challenges, especially in the biomedical

domain. Articles contain many abbreviated names and noun compounds with punctuations. Also, different citation styles are used in the bodies of articles. *SciSpacy* uses a pre-trained dependency parser for the biomedical domain, which obviates the need for rule-based sentence splitting.

5.1.2. Named Entity Recognition

Named entity recognition is the task of identifying and classifying named entities in given text into predefined categories. *SciSpacy* [65] used for named entity recognition task, which is a Python library developed for the processing of biomedical text. For the recognition task, it uses a transition-based chunking model where multi-token name representations are constructed and used [66].

In the *SciSpacy* library, there are several models for different domains. Every model is pre-trained with related datasets. This work uses two different pre-trained *SciSpacy* models. In order to identify chemical and protein entities, *en_ner_bionlp13cg_md* model has been used which is trained on *BIONLP13CG* dataset. *BIONLP13CG* dataset contains manual annotated PubMed articles relate to hallmarks of cancer [67]. To identify disease related biomedical entities, *en_ner_bc5cdr_md* model has been used which trained on *BC5CDR* dataset. Domain experts manually annotated 1500 PubMed articles with chemical and disease mentions to curate *BC5CDR* dataset [68].

In the process of identifying biomedical entities, *SciSpacy* annotates named entities with location information and category information. Category information describes the type of biomedical entity. Table 5.1 shows the related model name and possible categories *SciSpacy* can annotate. Our prototype maps the *SciSpacy* categories into entity types, which is shown in the last column. *en_ner_bionlp13cg_md* contain 16 possible categories. Every identified entity category with *SIMPLE_CHEMICAL* considered as *chemical* and any category described in Table 5.1 for *en_ner_bionlp13cg_md*'s *SciSpacy Categories* column considered as *protein*. Any recognized entity with a category other than these is ignored for further processing. For *en_ner_bc5cdr_md*,

there are 2 possible categories which are *CHEMICAL* and *DISEASE*. Due to this work uses previous model to identify chemicals, identified entities with *CHEMICAL* category are ignored and identified entities with *DISEASE* category are considered as *diseases*.

Table 5.1. SciSpacy models with possible category sets.

Model Name	**SciSpacy Categories**	**Identified Entity Type**
bionlp13cg_md	SIMPLE_CHEMICAL	Chemical
	GENE_OR_GENE_PRODUCT, AMINO_ACID, CANCER, CELL, CELLULAR_COMPONENT, ORGANISM_SUBSTANCE	Protein
bc5cdr_md	CHEMICAL	-
	DISEASE	Disease

5.1.3. Named Entity Normalization

The named entity normalization task links identified named entities into concepts. The proposed prototype is a semantic similarity-based unsupervised named entity normalization method which is based on [12]. Semantically similar words have similar word embedding vectors, which also applies to named entities and ontology concepts. An ontology concept is represented with its ontology concept term, which is the concept's name.

Suppose ontology concept term's embedding is close to the named entity's embedding in vector space. In that case, both are considered semantically similar, and the named entity could be normalized to mentioned ontology concept. Figure 5.1 shows the flow of normalization prototype.

While calculating ontology concept terms' and named entities' vectors, a pre-trained word embedding was used, as shown in Figure 5.1. *BioWordVec* used as

pre-trained word embedding for implementation. *BioWordVec* is a publicly available biomedical word embedding that is trained on 23,714,373 PubMed documents and 2,083,180 clinical notes from MIMIC-lll Clinical database [69] [70].

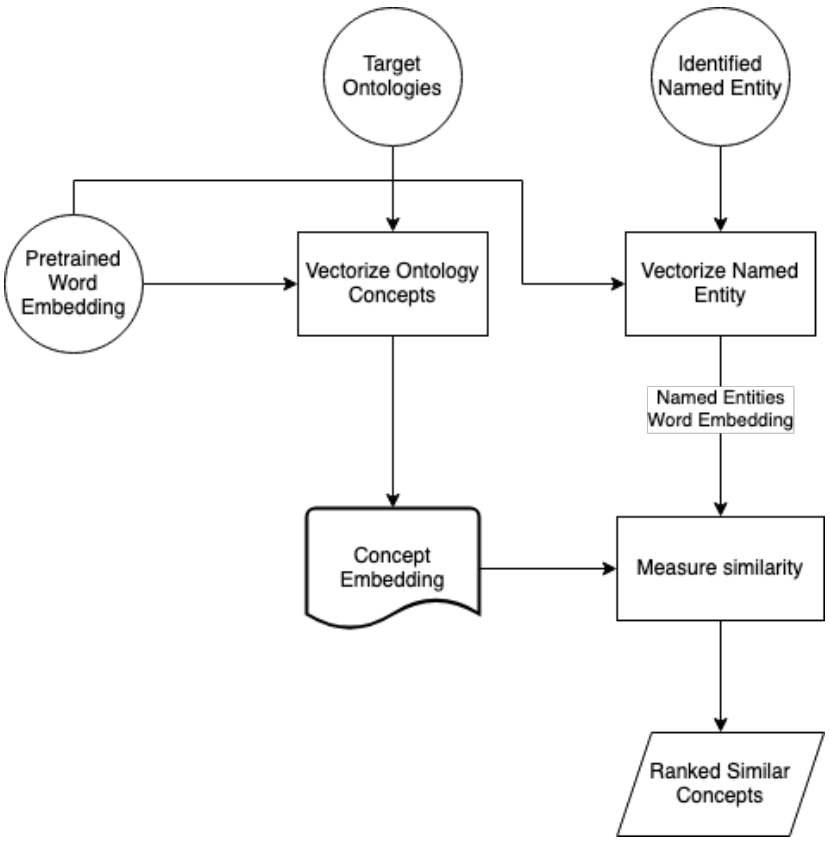

Figure 5.1. Normalization flow of prototype.

The first step of the prototype is the construction of concepts embedding. Concept embedding is the vector space of concepts acquired from target ontologies. The vectors in concept embedding are based on the text of ontology concepts called ontology concept terms. In order to construct this embedding, firstly, an ontology concept term is preprocessed. The preprocessing consists of lowering words, replacing punctuation in words with space, and filtering outing stop words. Output text of preprocessing is then searched in the pre-trained word embedding. If the ontology concept term exists in pre-trained word embedding, its vector is added to concept embedding. Nonexistent concepts are ignored. For concept terms with multi-words, the average vector of each

word's vector is calculated. The multi-word term is also ignored if any word does not exist in the pre-trained word embedding. Lastly, concept embedding is saved as a file to be used later.

The second step is normalizing entities through previously constructed concept embedding. First, the named entity's word embedding must be calculated. Similar to calculating ontology concept term embedding, same preprocessing procedure was executed on the named entity. Preprocessing output is then searched in pre-trained word embedding to find the named entity's vector. For multi-word entities, each word's vector is summed and divided with the number of words to calculate average embedding.

In order to find the most suitable concept to a named entity, semantic similarities of ontology concepts to named entity embedding are calculated and ranked according to their scores. Semantic similarity score calculated via cosine similarity, where most similar embeddings are the ones with lowest cosine angle in vector space [71]. While ranking ontology concepts, top k ontology concepts retrieved with at least 0.9 similarity score and named entity normalized to a most similar one.

5.1.4. Relation Extraction

Relation extraction task is the identification of relational facts between previously detected entities [72].

This work aims to extract relations between chemical-protein pairs and relations between chemical-disease pairs in a given sentence. While chemical-protein relations classified as multiclass relations, chemical-disease relations classified as binary relation. Multiclass relations type details are described in Section 4.1.2.

The proposed approach uses deep learning models to classify given sentence into a relation type. Approach requires two different classifier models one for chemical-

protein multiclass relation classification and one for chemical-disease binary relation classification. Both classifier models are fine-tuned *BioBERT* models. *BioBERT* is a transformer model for language representation which pretrained on a large biomedical based corpora [73]. Fine-tuning and classification methods are based on *Vapur* [27].

5.1.4.1. Dataset. Chemical-protein multiclass relation extraction model trained on *ChemProt* dataset [74]. 2,432 abstracts related to chemical-protein interactions are manually annotated by domain experts in terms of chemical and protein entities and their interactions. 62,147 entities and 15,769 interactions annotated by experts.

Chemical-disease binary relation extraction model trained on *CDR* dataset [68]. Similar to the *ChemProt* dataset, it is also manually annotated by domain experts in terms of chemical entities, disease entities, and their binary interactions. The dataset consists of 1500 abstracts with 28785 entity mentions and 4038 relations.

For both relation extraction tasks, a *BioBERT* model fine-tuned. Fine-tuning *BioBERT* is based on *Vapur*'s relation extraction method except they extract relations of chemical-protein interactions as binary while this work extracts multiclass relations. For chemical-disease interactions, this work also extracts binary relations, thus fine-tuning process is same as *Vapur*.

5.1.4.2. Fine-tuning. Two different *BioBERT* models fine-tuned with datasets described in Section 5.1.4.1. For fine-tuning, firstly, every sentence preprocessed by labeling named entities. Chemicals labelled with <e1> and </e1> whereas proteins and diseases labeled with <e2> and </e2>. This process encode location and type information of the entity for *BioBERT* model.

As in *Vapur*, for binary classification an additional layer of binary log-softmax classifier added to *BioBERT* [27]. For multi-class classification case, additional layer of multi-class log-softmax classifier trained. Both model use cross-entropy as loss function.

Fine-tuning results of chemical-protein relation extraction shown in Table 5.2 by precision, recall and f1-scores evaluated on test folds of *ChemProt* dataset.

Table 5.2. BioBERT fine-tuning results for multiclass classification.

Relation Type	**Precision**	**Recall**	**F1-Score**
CPR:1	0.648	0.705	0.675
CPR:2	0.583	0.502	0.539
CPR:3	0.737	0.725	0.731
CPR:4	0.811	0.809	0.810
CPR:5	0.758	0.709	0.733
CPR:7	0.848	0.753	0.797
CPR:8	0.550	0.366	0.440
CPR:9	0.200	0.071	0.105
No Relation	0.873	0.909	0.890
Overall	0.716	0.661	0.687

5.2. Representation

Representation module converts semantically annotated data into RDF triples through proposed BEE ontology. RDF triplets stored in GraphDB instance. Annotations converted into SPARQL queries at the application level and executed at the database level to create and store relevant RDF triplets. The key points to generate instance identifiers are described in Section 4.1.2. *bee:BiomedicalEntityPairRelation* identifier created after hashing the concatenation of biomedical entity identifiers and the relation type identifier.

In this work, the inference of new relationships has two main sources: proposed ontology and custom ruleset. GraphDB reasoning engines strategy for inference is forward-chaining of entailment rules with the goal of total materialization; thus, new relationships are inferred in insertion time. Inference that based on ontology occurs due

to relationships defined between properties such as inverse relationships or transitive relationships. The custom ruleset is defined on the GraphDB system with a language similar to logic programming. Rules are detailed in Section 5.2.1.

5.2.1. Custom Rules

Relation between biomedical entities is not explicitly stated on BEE ontology; however, two entities that are part of the same *bee:BiomedicalEntityPairRelation* are considered as related. This relationship's inference rule is shown in Figure 5.2. This rule infers a relationship between biomedically related entities with *bee:isRelated* property.

```
Prefices
    rdf : http://www.w3.org/1999/02/22-rdf-syntax-ns#
    bee : http://soslab.cmpe.boun.edu.tr/ontologies/bee#

Id: is_related_entities
  rel <bee:hasBiomedicalEntity> x
  rel <bee:hasBiomedicalEntity> y
-----------------------------------
  x <bee:isRelated> y [Constraint x != y]
```

Figure 5.2. Custom rule to infer entity relationship.

This work does not extract any knowledge about the relation of protein-disease pairs; however, a protein instance and a disease instance are considered as related if they both have a relationship with the same chemical instance. Inferring such knowledge makes protein-disease relations explicit and available for processing. Figure 5.3 describes the rule to infer *bee:isRelated* property between such protein-disease pairs.

The relationship between articles and biomedical entities is not explicitly stated in BEE ontology. This relationship inferred as *bee:refers to* property by rule displayed in Listing 5.4.

```
Prefices
    rdf : http://www.w3.org/1999/02/22−rdf−syntax−ns#
    bee : http://soslab.cmpe.boun.edu.tr/ontologies/bee#

Id: prot_dis_rel
  c <rdf:type> <bee:Chemical>
  p <rdf:type> <bee:Protein>
  d <rdf:type> <bee:Disease>
  c <bee:isRelated> p
  c <bee:isRelated> d
−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−
  d <bee:isRelated> p
```

Figure 5.3. Custom rule to infer protein-disease relationship.

```
Prefices
    rdf : http://www.w3.org/1999/02/22−rdf−syntax−ns#
    bee : http://soslab.cmpe.boun.edu.tr/ontologies/bee#

Id: article_refersto_ent
  ent <rdf:type> <bee:BiomedicalEntity>
  ent <bee:isBiomedicalEntityOf> rel
  rel <bee:isBiomedicalEntityPairRelationOf> evi
  evi <sio:SIO_000773> a
−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−
  a <sio:SIO_000628> ent
```

Figure 5.4. Custom rule to infer relationship between article and entity.

A biomedical entity pair relation that consist of *inhibition* or *antagonist* relation decrease activity of target entity. Likewise, *stimulant* relation increase activity of target entity. Figure 5.5 describes set of rules to infer additional type for *bee:BiomedicalEntity PairRelation*. As shown in listing, every pair relation that has relation type of *obo: MI_ 0623 (inhibition)* or *obo:MI_ 0626 (antagonist)* inferred as *bee:ActivityDecreaser PairRelation* class. Every pair relation that has relation type of *obo:MI_ 0624* (stimulant) inferred as *bee:ActivityIncreaserPairRelation* class.

The subject of a biomedical entity is considered as the subject of the article that refers to entity. Subjects are not extracted from articles in this work; thus, knowledge in DBpedia is utilized to infer this data. Figure 5.6 shows the rule to infer article subjects. In order to infer this relationship, we downloaded 15,363 triples from DBpedia that contain knowledge of chemical entity' subjects and imported them into our data source.

```
Prefices
    rdf : http://www.w3.org/1999/02/22-rdf-syntax-ns#
    bee : http://soslab.cmpe.boun.edu.tr/ontologies/bee#
    obo : http://purl.obolibrary.org/obo/

Id: inhibition

  r <rdf:type> <bee:BiomedicalEntityPairRelation>
  r <bee:hasRelationType> <obo:MI_0623>
----------------------------------------------------
  r <rdf:type> <bee:ActivityDecreaserPairRelation>

Id: antagonist

  r <rdf:type> <bee:BiomedicalEntityPairRelation>
  r <bee:hasRelationType> <obo:MI_0626>
----------------------------------------------------
  r <rdf:type> <bee:ActivityDecreaserPairRelation>

Id: stimulant

  r <rdf:type> <bee:BiomedicalEntityPairRelation>
  r <bee:hasRelationType> <obo:MI_0624>
----------------------------------------------------
  r <rdf:type> <bee:ActivityIncreaserPairRelation>
```

Figure 5.5. Custom rule to infer additional type of pair relation.

```
Prefices
    rdf : http://www.w3.org/1999/02/22-rdf-syntax-ns#
    bee : http://soslab.cmpe.boun.edu.tr/ontologies/bee#
    sio : http://semanticscience.org/resource/
    dct : http://purl.org/dc/terms/

Id: article_subject

  a <rdf:type> <bee:Article>
  a <sio:SIO_000772> evi
  evi <bee:hasBiomedicalEntityPairRelation> rel
  rel <bee:hasBiomedicalEntity> ent
  ent <dct:subject> s
----------------------------------------------------
  a <dct:subject> s
```

Figure 5.6. Custom rule to infer article subject.

5.3. Semantic Application

A prototype web application was developed to show the benefits of ontological representation of semantic annotations. This application is capable of querying predefined evaluation tasks with parameters. Content of predefined evaluation tasks detailed in Section 6. Query results are shown through HTML pages. An example figure of Web Application can be seen in Figure 5.7. The application's backend is developed using Python, and the frontend is developed using HTML and Javascript.

The user can enter different types of input parameters for every predefined task, and application queries semantic data stored in the GraphDB RDF repository. Data inference is executed at the database level and at insertion time since the GraphDB uses forward-chaining strategy.

Semantic Application enriches stored data from Linked Open Data sources for several predefined tasks. It is also done at the database level thanks to federated query [75] standard of W3C [76]. A federated query is a specification to merge data distributed across the Web. In this work, Wikidata and DBpedia are used as an external data sources.

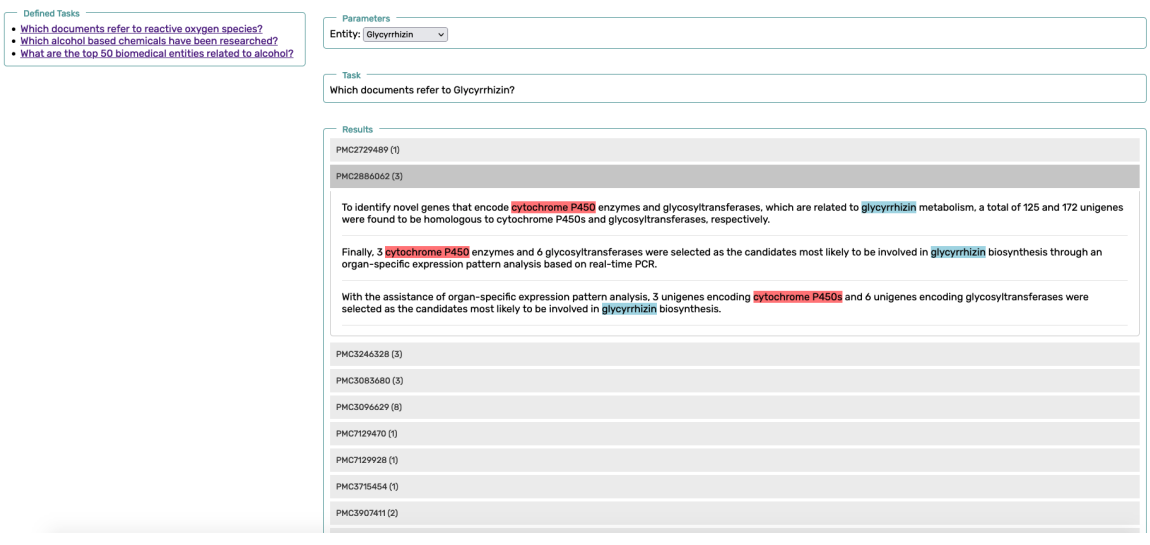

Figure 5.7. Main page of the semantic web application.

6. EXPERIMENTS AND RESULTS

This chapter presents the benefits of ontology-based representation of biomedical entity relation annotations. Several evaluation tasks defined as use cases to examine the usefulness of our approach and results are presented through semantic web application prototype, which is described in Section 5.3. Evaluation tasks were executed on semantic annotations that were generated from The Covid-19 Open Research Dataset.

The following sections describe the utilized dataset for experiments and analyze the evaluation tasks in detail.

6.1. Dataset

Abstracts of The Covid-19 Open Research Dataset(CORD-19) annotated to demonstrate the utility of the proposed approach. CORD-19 is a dataset of scientific documents related to historical coronavirus and novel Covid-19 disease [17]. It is regularly updated with recent papers about the domain. In this work, a snapshot of April 17, 2020, has been used, which contains around 52 thousand documents with around 43 thousand abstracts.

CORD-19 dataset annotated by the approach described in Section 4.1. Annotation process resulted in 1,728 multiclass chemical-protein relations, 819 binary chemical-disease relations extracted with unique 1,023 chemicals, 604 proteins, and 364 diseases. These annotations semantically represented based on Biomedical Entities Evidence (BEE) ontology. We explicitly represented over 54,000 semantic triples, resulting in over 174,000 new semantic triple inferences with an expansion ratio of 4,21. The semantic triples are stored in the GraphDB RDF repository instance.

6.2. Evaluation Tasks

In order to examine the ontology-based semantic representation approach, several information retrieval tasks are defined as use cases. These tasks have been evaluated using GraphDB SPARQL endpoint. The results are displayed through semantic application prototype, which is described in Section 5.3.

To demonstrate the utility of ontology-based representation of annotations, we provide a comparative analysis with an annotation application (AA) in terms of the effort required to perform the same tasks. AA has been considered to store biomedical relation annotations in the database without further representation approach; thus, storing the output of Section 4.1.1. That output contains a set of chemical-protein relations and a set of chemical-disease relations for every article. Each biomedical relation contains two entities with required information such as provenance data and concept identifiers of normalized ontology.

6.2.1. Task 1: Which documents refer to *reactive oxgyen species*?

This task queries articles that refer to the selected biomedical entity. A biomedical entity can be expressed in articles with different surface forms, such as abbreviations or synonyms. While searching for a biomedical entity, every article that refers to any surface form of a biomedical entity should be queried.

Ontology-based representation expresses biomedical entities by their concepts. While the surface form of an entity differs, it is possible to search by its concept.

Figure 6.1 shows the SPARQL query for related task for *Reactive Oxygen Species* (*ROS*) which is a chemical with concept identifier *CHEBI:26523*. In this query, every article referring to *ROS* is fetched. In addition to the article, target entity' and related entity' evidence are fetched to present the results to the user.

As seen in Figure 6.2, *ROS* mentioned in article *PMC6113620* with different surface forms. By semantically annotating article and representing based on BEE ontology, entities can be queried by concept identifiers while surface forms differ.

AA representation is also capable of querying biomedical entities by their concepts since annotations already contain concept identifiers.

```
SELECT
    ?article ?articleId ?sentenceIndex ?relType ?chem ?chemRep ?chemStart
    ?chemEnd ?prot ?protRep ?protStart ?protEnd ?dis ?disStart ?disEnd
WHERE{
    ?article sio:SIO_000772 ?relEvidence;
            bee:hasFriendlyId ?articleId.
    ?relEvidence bee:hasChemicalEvidence ?chemRef;
                bee:hasBiomedicalEntityPairRelation ?pairRel .
    ?pairRel bee:hasChemical ?chem;
            bee:hasRelationType ?relType.
    ?chemRef bee:hasTextRepresentation ?chemRep;
            bee:hasStartPosition ?chemStart;
            bee:hasEndPosition ?chemEnd;
            bee:hasSentenceIndex ?sentenceIndex.
    {   ?relEvidence bee:hasProteinEvidence ?protRef.
        ?pairRel bee:hasProtein ?prot.
        ?protRef bee:hasTextRepresentation ?protRep;
                bee:hasStartPosition ?protStart;
                bee:hasEndPosition ?protEnd. }
    UNION
    {   ?relEvidence bee:hasDiseaseEvidence ?protRef.
        ?pairRel bee:hasDisease ?dis.
        ?disRef  bee:hasTextRepresentation ?disRep;
                bee:hasStartPosition ?disStart;
                bee:hasEndPosition ?disEnd. }


    #CHEBI_26523: Reactive Oxygen Species
    FILTER(?chem = obo:CHEBI_26523)
}
```

Figure 6.1. SPARQL query of Task 1 to fetch articles that refer to given entity.

6.2.2. Task 2: Which articles research about *alcohol*?

This task fetches articles that research about members of a given biomedical entity class.

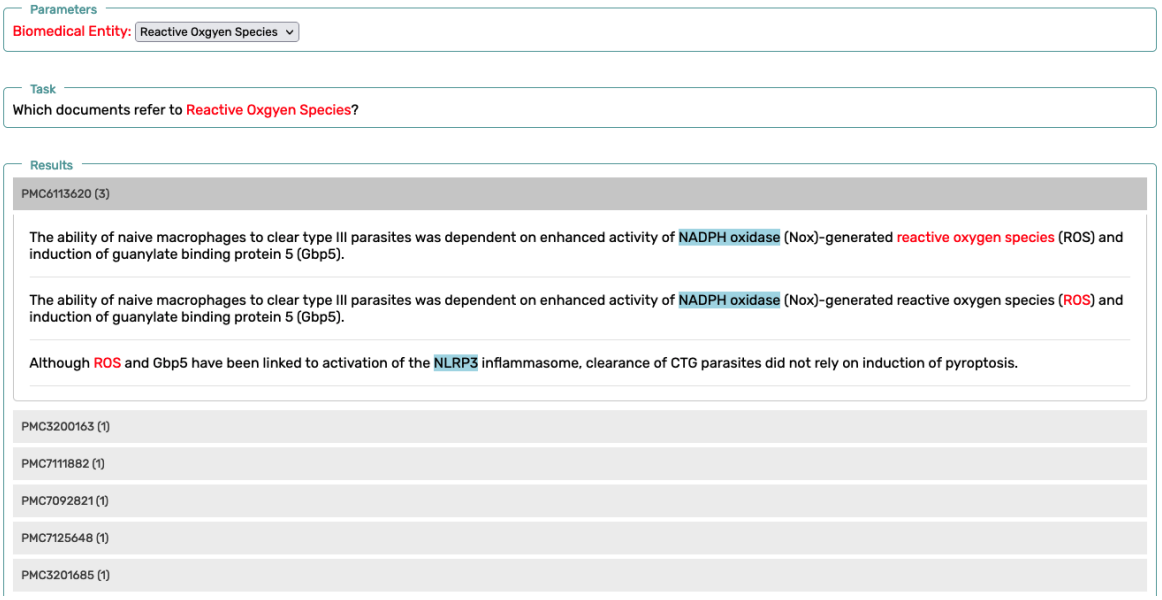

Figure 6.2. Result of Task 1 for chemical *Reactive Oxgyen Species*.

There can be many members of a biomedical entity class. In ChEBI ontology, *alcohol* class have 3957 members. In order to query articles that research on *alcohol* concept, every member of *alcohol* have to be searched in articles however there are no asserted knowledge of *alcohol* class children extracted from articles.

In this work, articles are annotated with biomedical entity concepts and represented based on BEE ontology. Chemicals are represented by ChEBI ontology concepts. Since, *rdf:subClassOf* predicate is a transitive relation, every concept that is a subclass of a parent concept is also a subclass of its parent's parent concepts. With ontology-based representation, this transitive relation is already inferred. Subclass information can be utilized from an external data source representing chemicals with ChEBI ontology by linking two data sources over chemical concepts.

Figure 6.3 shows the example SPARQL query for chemical class *alcohol*. This query connected an external data source that serves ChEBI ontology and our RDF repository over chemical concepts. For this example, we served ChEBI ontology locally to replicate external data sources and utilize them through a federated query as shown in Lines 9-13. By linking two data sources, every article that refers to any child class of *alcohol* class has been joined into our repository in Line 11. For display purposes,

list of *alcohol* children ordered by their occurrences.

Figure 6.4 shows the example display of results for chemical *alcohol*. It can be seen that various members of *alcohol* class captured and presented to the user.

To achieve the same results with *AA* representation, firstly, every child class of *alcohol* has to be fetched from an external source or an ontology. Later, each child concept has to be searched in annotations to query articles. Performing the same task requires two steps development.

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX bee: <http://soslab.cmpe.boun.edu.tr/ontologies/bee#>
PREFIX sio: <http://semanticscience.org/resource/>

SELECT
    ?article ?doi ?entityChild ?entityLabel
WHERE
{
    SERVICE <http://localhost:7200/repositories/chebi>
    {
        ?entityChild rdfs:subClassOf+ obo:CHEBI_30879 ;
                                rdfs:label ?entityLabel .
    }
    ?rel bee:hasChemical ?entityChild .
    ?relEvidence bee:hasBiomedicalEntityPairRelation ?rel .
    ?article sio:SIO_000772 ?relEvidence ;
            bee:hasFriendlyId ?doi .
}
LIMIT 1000
```

Figure 6.3. SPARQL query of Task 2 to fetch articles that research about entity.

Figure 6.4. Result of Task 2 for chemical *alcohol*.

6.2.3. Task 3: Which *hydrolase genes* proteins have biomedical relations with chemical *angiotensin 2*?

This task queries members of a selected protein that have biomedical relations with the selected chemical.

A scientist can research relations of a chemical with a protein class which requires querying associations between the given chemical instance and every member of the given protein class. The member of selected protein class information is not extracted from articles thus it is not an asserted knowledge.

Figure 6.5 shows the query of Task 3 for chemical *angiotensin 2* and protein class *hydrolase genes*. Every appropriate *bee:BiomedicalEntityPairRelation* have been queried where one biomedical entity is *angiotensin 2* and the other is any child of *hydrolase genes*. For members of *hydrolase genes*, an external data source is linked to our data source in Line 13. Since both data sources represent protein entities based on NCIt ontology concepts, they are easily connected. We served NCIt ontology from a locally hosted data source to replicate external LOD source behaviour. Due to ontology-based representation, *rdf:subClassOf* relation between *hydrolase genes* and

every member of that concept are already inferred. This query uses inverse relationship feature of BEE ontology in Lines 8-8 since *bee:isBiomedicalEntityOf* is inferred due to inverse relation with *bee:hasChemical* and *bee:hasProtein*.

Figure 6.6 shows the results for chemical *angiotensin 2*. *ACE2 Gene*, which is one of the child of *hydrolase genes* protein class, have four different relationship with *angiotensin 2* according to extracted articles. Other members of *hydrolase genes* are also showed in application.

To perform the same task with *AA* representation, firstly, it is necessary to determine the members of the *hydrolase gene* class from an external source. Later, the database should be queried for every annotation that consists of biomedical relation of chemical *angiotensin 2* and any member of the protein *hydrolase gene* concept.

```
PREFIX obo: <http://purl.obolibrary.org/obo/>
PREFIX bee: <http://soslab.cmpe.boun.edu.tr/ontologies/bee#>

SELECT
    ?relatedEntity ?relType (COUNT(*) as ?c)
WHERE {
    ?target bee:isBiomedicalEntityOf _:r.
    ?relatedEntity  bee:isBiomedicalEntityOf _:r.
    _:r bee:hasRelationType ?relype.

    SERVICE <http://localhost:3030/ncit/sparql>
    {
        ?relatedEntity  rdfs:subClassOf+ ncit:C25804;
                        rdfs:label ?name.
    }

    FILTER(?target != ?relatedEntity
            && ?target = obo:CHEBI_48432)
}
GROUP BY(?relatedEntity ?relType)
HAVING (COUNT(*) > 1)
```

Figure 6.5. SPARQL query of Task 3 to fetch biomedical entities that have more than one relation type with angiotensin 2.

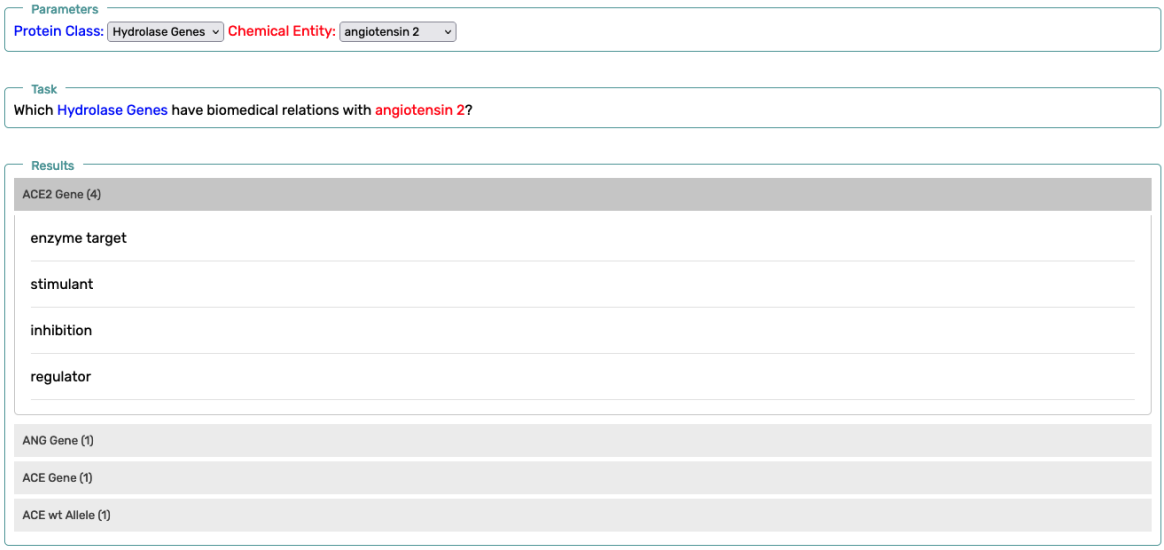

Figure 6.6. Result of Task 3 for chemical *angiotensin 2*.

6.2.4. Task 4: Which articles refer to chemicals that treat *arterial tension*?

This task identifies articles that refer to chemicals that treat the selected disease. This work does not extract the treatment relation of chemicals, and it is not represented by BEE ontology. However, a scientist can view articles about chemicals based on their treatment relation with diseases. In order to represent this kind of knowledge, external data sources should be utilized.

One of the fundamentals of the Semantic Web is reusing domain knowledge. While BEE ontology does not cover any treatment data, Linked Open Data (LOD) resources that express this knowledge can be utilized. Combining our data repository and LOD enable such information for processing.

In this task, Wikidata used as LOD source. In Figure 6.8, an example of Wikidata model for *arterial hypertension* and relation with our triplestore have been shown. As *carvedilol* represented by ChEBI ontology concept in our triplestore and also in Wikidata, both data sources can be integrated over this class.

In Figure 6.7, Wikidata and our data repository are linked over *?chem* variable in Line 16, which represents any chemical that is referred in an article. On Wikidata,

arterial hypertension have relationship with *carvedilol* by *wdt:P2176* predicate which represents *drug used for treatment*, and *carvedilol* have relationship with its ChEBI concept by *wdtn:P683* predicate. Federated query service provide integration to Wikidata SPARQL service.

In Figure 6.9, articles that refer to chemicals that treat *arterial hypertension* have shown. Chemicals colored in red in evidence sentences while related entity to detected chemical colored with blue background.

To achieve the same results with *AA* representation, every chemical that is referred in articles should be queried from the database. Later, every chemical should be queried on an external source to learn the treatment relation with *arterial hypertension*. Whereas two data sources are connected over the ChEBI concept, and a single query achieved the same results with ontology-based representation.

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX bee: <http://soslab.cmpe.boun.edu.tr/ontologies/bee#>
PREFIX wdt:  <http://www.wikidata.org/prop/direct/>
PREFIX sio: <http://semanticscience.org/resource/>
PREFIX wdtn: <http://www.wikidata.org/prop/direct-normalized/>

SELECT
    ?article ?chem
WHERE {
  ?article      sio:SIO_000772 _:relEvidence .
  _:relEvidence bee:hasBiomedicalEntityPairRelation _:rel .
  _:rel         bee:hasChemical ?chem .

  SERVICE <https://query.wikidata.org/sparql> {
        wd:Q41861 wdt:P2176 _:wkChem .
        _:wkChem  wdtn:P683 ?chem .
  }
}
LIMIT 1000
```

Figure 6.7. SPARQL query of Task 4 to identify articles that refer to chemicals that treat *arterial hypertension*.

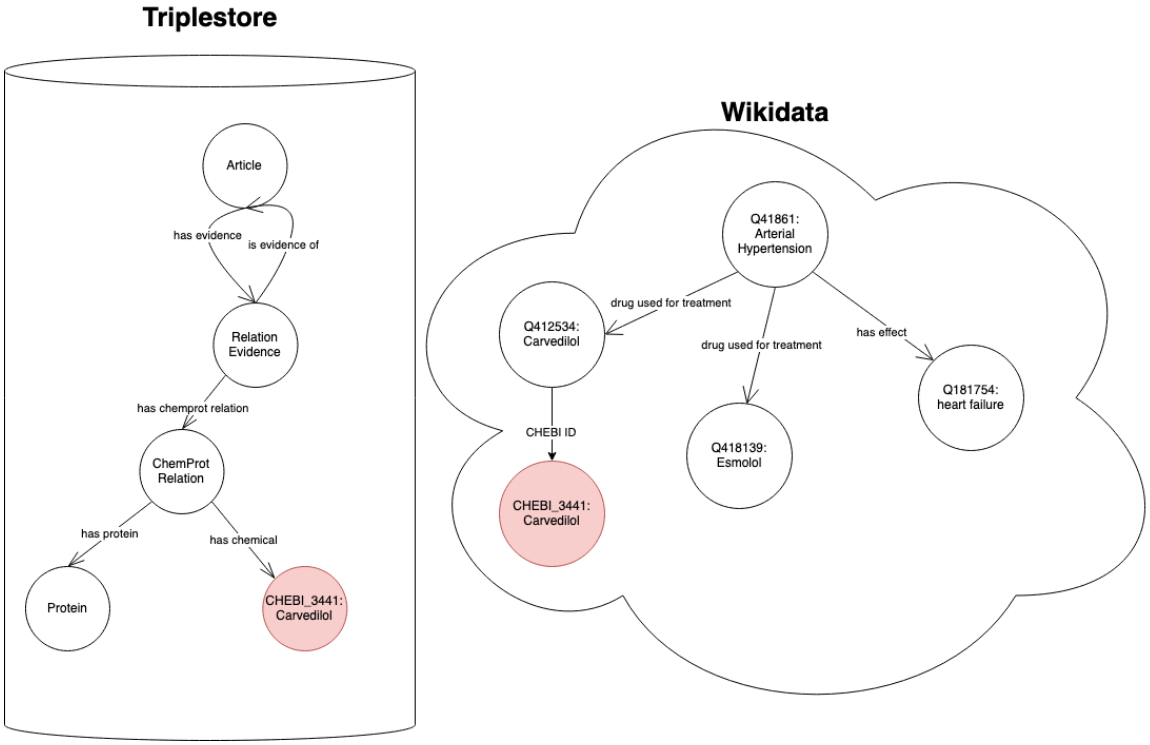

Figure 6.8. Relation between Wikidata and our triplestore over chemical *carvedilol*.

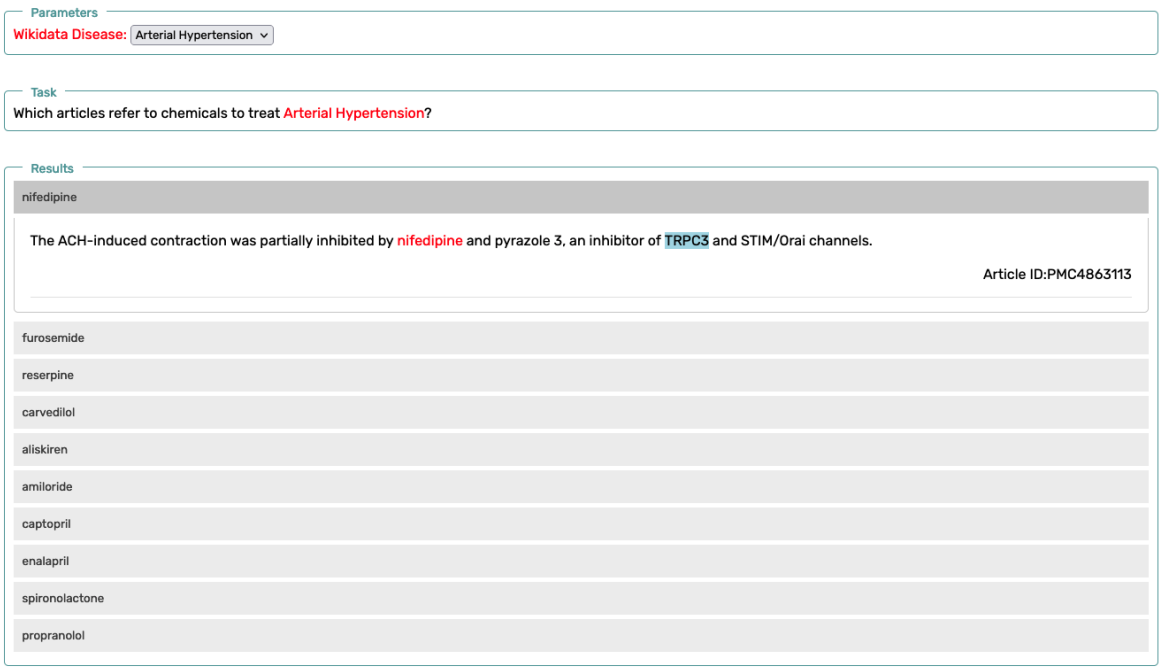

Figure 6.9. Result of Task 4 for *arterial hypertension* which is a disease stated in Wikidata.

6.2.5. Task 5: Which related biomedical entities have contradicting evidences?

This task queries related biomedical entities that have conflicting evidences in different biomedical articles.

In a chemical-protein association, *inhibition* and *antagonist* relations decrease activity of protein and *stimulant* relation increase activity of protein. In this work, these relation types defined as contradicting relations for same chemical-protein associations since effects are opposite.

Ontology based semantic representation of annotations enable inference of new data and make knowledge ready. Every *bee:BiomedicalEntityPairRelation* that contain *inhibition* or *antagonist* relations of chemical-protein pairs are inferred as *bee:Activity DecreaserRelation*. Likewise, every *bee:BiomedicalEntityPairRelation* that contain *stimulant* relation of chemical-protein pairs are defined as *bee:ActivityIncreaserRelation*. These rules defined on triplestore as described in Section 5.2 and inference is done by GraphDB reasoning engine.

In Figure 6.10 related SPARQL query shown. In this query, *_:pr1* variable represents activity decreasing pair relations and *_:pr2* variable represents activity increasing pair relations of same chemical-protein pair. Articles refer to these relations are also queried to display user. Reasoner engine is already inferred types of pair relations as *bee:ActivityDecreaserRelation* and *bee:ActivityIncreaserRelation* based on rules defined in Figure 5.5.

Figure 6.11 shows the results of Task 5. Every accordion header displays a chemical-protein pair where chemicals are wrapped into a blue circle and proteins are wrapped into a red circle. In the accordion, every row separated by a divider displays two sentences from different articles as evidence of contradicting relations of the same chemical-protein pair. The first sentence container displays decreasing activity relation example, and the second sentence container displays an increasing

activity relation example. Relation types are displayed as the header of the sentence containers. Chemicals expressed with blue background and protein expressed in red font in sentences. Finally, article identifiers that contain sentences are also displayed in sentence containers. As shown in Figure 6.11, chemical *angiotensin 2* and protein *ACE2 gene* have 2 contradicting relation pairs in biomedical articles. In *PMC4231883* biomedical entities have *inhibition* relation which is a activity decreaser. In *PMC-3321295* entities have *stimulant* relation which is a activity increaser.

To achieve same results with *AA* representation, a development needed to query annotations with *inhibition* or *antagonist* relations as activity decreaser relations. Also, another development will be needed to query *stimulant* relations as activity increaser relations. Later chemical-protein pairs that occur in two sets have to be identified. However, inferred data due to ontology rules provide available data to process and in a single query results are achieved with ontology-based representation.

```
PREFIX rdf: <http://www.w3.org/1999/02/22−rdf−syntax−ns#>
PREFIX bee: <http://soslab.cmpe.boun.edu.tr/ontologies/bee#>
PREFIX sio: <http://semanticscience.org/resource/>

SELECT
    ?articleDec ?relDec ?articleInc ?relInc ?c ?p
WHERE {
    _:pr1 rdf:type bee:ActivityDecreaserPairRelation ;
          bee:hasChemical ?c ;
          bee:hasProtein    ?p ;
          bee:hasRelationType ?relDec.

    _:pr2 rdf:type bee:ActivityIncreaserPairRelation ;
          bee:hasChemical ?c ;
          bee:hasProtein ?p ;
          bee:hasRelationType ?relInc.

    ?articleDec sio:SIO_000628 _:pr1 .
    ?articleInc sio:SIO_000628 _:pr2 .
}
```

Figure 6.10. SPARQL query of Task 5 to fetch articles that contain contradicting relations.

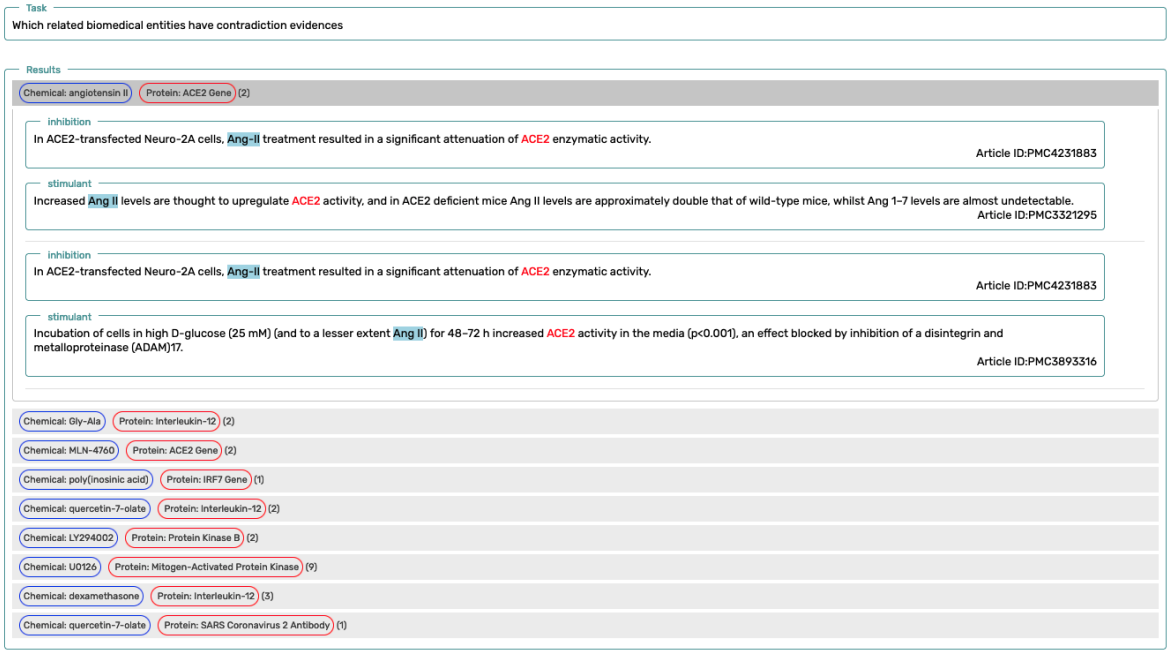

Figure 6.11. Chemical-protein associations that have contradicting evidences.

6.2.6. Task 6: Which protein-disease pairs are related?

This task queries related protein-disease pairs in articles. In this thesis, we do not extract relationships between protein and disease entities, and BEE ontology does not represent protein-disease pairs explicitly. However, it is valuable data for researchers to analyze related proteins and diseases.

Ontology based representation of annotations enable inference. Based on the rules described in Section 5.2 *bee:isRelated* object property can be inferred between biomedical entities.

By the rule defined in Figure 5.2 *bee:isRelated* property inferred between every related entity that share same *bee:BiomedicalEntityPairRelation*. With the rule described in Listing 5.3, also relation between protein-disease pair can be inferred if they are related to same chemical.

In Figure 6.12 related SPARQL query shown. Here, related protein-disease pairs are extracted by *bee:isRelated* predicate. Also, chemicals responsible for this inferred

relationship are queried to display to the user. GraphDB reasoner infers *bee:isRelated* property at insertion time, and data is ready to process at query time.

In Figure 6.13, chemicals and protein-disease pairs that have relationship due to stated chemicals are shown. Chemical *zidovudine* cause two inferred relationships which are protein *reverse transcriptase* and *tonsillitis*, and protein *RNA directed RNA polymerase* and disease *tonsillitis*.

To achieve the same results, *AA* have to fetch chemical-protein pairs and chemical-disease pairs separately. Later, chemicals that exist in both sets should be identified, and proteins and diseases that relate to those chemicals should be combined. Protein-disease relations are not ready to use in *AA* representation. However, such implicit information is already inferred and available with ontology-based representation.

```
PREFIX bee: <http://soslab.cmpe.boun.edu.tr/ontologies/bee#>

SELECT
    ?dis ?prot ?chem (count(*) as ?c)
WHERE
{
    ?dis bee:isRelated ?prot .
    ?prot rdf:type bee:Protein ;
          bee:isRelated ?chem .
    ?dis  rdf:type bee:Disease ;
          bee:isRelated ?chem .
    ?chem rdf:type bee:Chemical.
    }
GROUP BY ?dis ?prot ?chem
```

Figure 6.12. SPARQL query of Task 6 to fetch related protein-disease pairs.

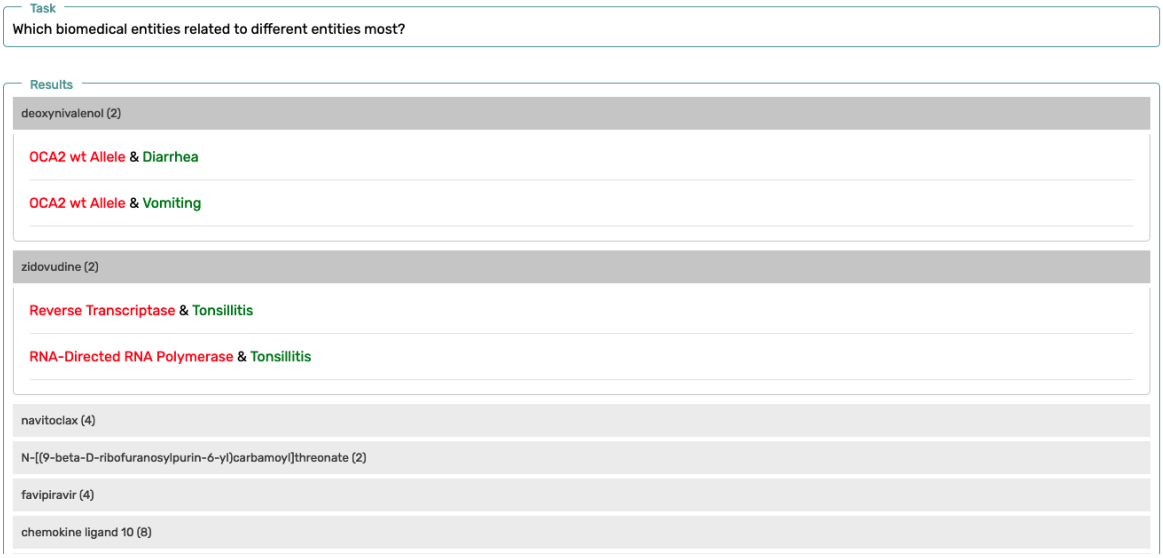

Figure 6.13. Inferred protein-disease pairs and chemicals that cause the relation.

6.2.7. Task 7: Which drugs have active chemical ingredients that decrease the activity of *Peptidase Genes*?

This task queries drugs that contain active chemical ingredients, which decrease the activity of the selected protein class.

A researcher may need to search about particular relation of members of a protein class and related drugs. While our work does not have any drug information, it exists on external Web sources.

Ontology-based representation of annotations enables the integration of Linked Open Data sources over chemicals represented by ChEBI ontology concepts. For this task, knowledge in Wikidata is utilized, which represents the information of chemical that is an active ingredient in drug relationship by *active ingredient in (wdt:P3780)* predicate. Chemicals are also represented by their ChEBI concepts in Wikidata and in our RDF repository. As well as the drug-chemical relationship, children of the selected protein class need to be obtained from an external source to execute this task.

Figure 6.14 shows the related SPARQL query. Query fetches every article that contains activity decreasing biomedical relation with appropriate biomedical entities

in Line 22. Relations that contain a child of *peptidase genes* class are filtered by linking an external data source over *?prot* variable in Line 18. The query shows that the external source and our RDF repository represent proteins by NCIt ontology concepts. Likewise, chemicals that are the active ingredient in any drugs are also filtered by linking Wikidata over chemicals represented by ChEBI ontology. In Line 22 two data sources are linked over *?chem* variable. Line 23 shows the triple to query *active ingredient in* relationship between chemicals and drugs. In this federated query, the drug label is also fetched.

In Figure 6.15, drugs that contain an active chemical ingredient that decrease the activity of *Peptidase Genes* are shown. Biomedical entities displayed with a blue background are chemicals that are active ingredients in shown drugs, and biomedical entities displayed with red font are proteins that are members of *Peptidase Genes*. As shown in figure, *Dasatinib* chemical is an active ingredient of *Sprycel* drug and decrease activity of *Src kinase* protein in article *PMC3692534*.

To achieve the same result with AA representation, firstly, annotations of activity decreasing relations should be fetched from the database. Later, every member of the *peptidase genes* class should be fetched, and the annotation set should be filtered. Finally, remainder chemicals should be queried in an external source to learn if they are active ingredients in any drug or not. A development to query activity decreasing relations and a development to utilize two different external sources are needed to achieve the same results of ontology-based representation.

```
PREFIX rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX bee:  <http://soslab.cmpe.boun.edu.tr/ontologies/bee#>
PREFIX sio:  <http://semanticscience.org/resource/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX wdtn: <http://www.wikidata.org/prop/direct-normalized/>
PREFIX wdt:  <http://www.wikidata.org/prop/direct/>

SELECT
    ?a ?drug ?drugLabel ?prot ?chem
WHERE
{
    ?a    sio:SIO_000628    _:rel.
    _:rel rdf:type       bee:ActivityDecreaserPairRelation ;
          bee:hasProtein  ?prot;
          bee:hasChemical ?chem .

    SERVICE <http://localhost:3030/ncit/sparql>{
        ?prot rdfs:subClassOf* ncit:C17018 .
    }

    SERVICE <https://query.wikidata.org/sparql>{
      _:wk  wdtn:P683 ?chem;
            wdt:P3780 ?drug;
      ?drug rdfs:label ?drugLabel filter(lang(?drugLabel)="en").
    }
}
```

Figure 6.14. Query of Task 7 to query drugs decreasing activity of *Peptidase Genes*.

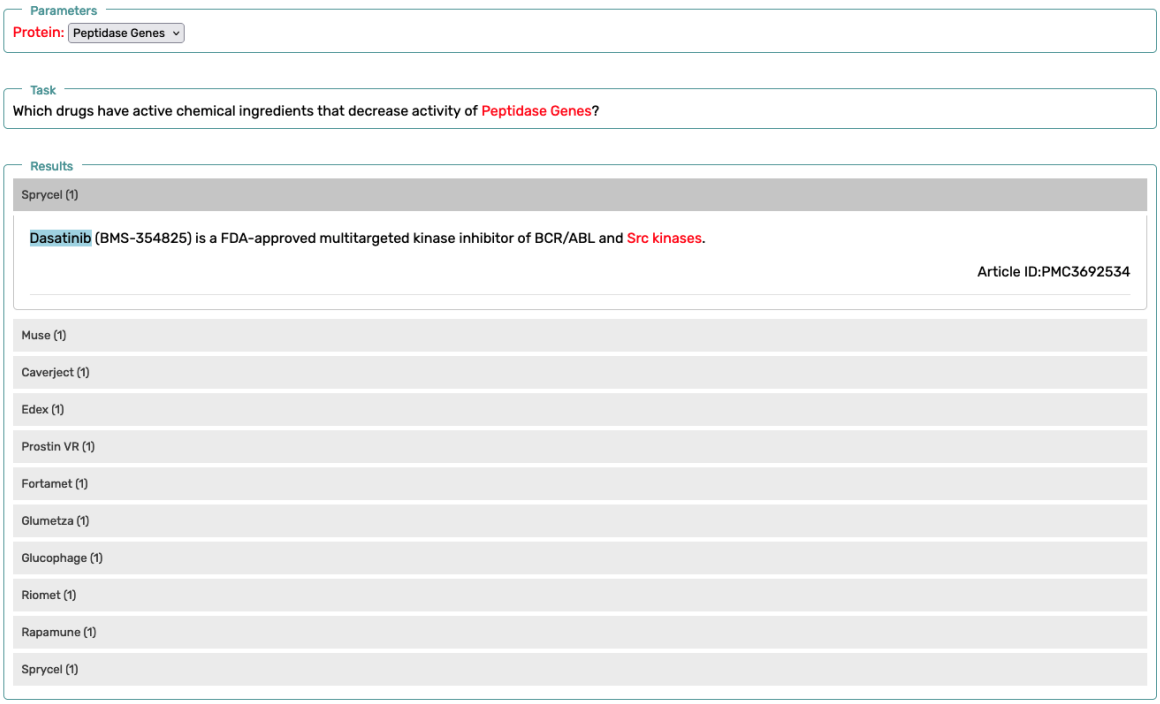

Figure 6.15. Drugs containing chemical ingredients that decrease peptidase genes.

6.2.8. Task 8: What are the most popular subjects of articles that refer to orphan drugs inhibiting any protein?

This task queries the most popular subjects of articles that refer to orphan drugs inhibiting any protein.

According to Food and Drug Administration (FDA), an orphan drug is a pharmaceutical agent that is intended for the treatment, prevention, or diagnosis of rare occurred diseases. This work neither extracts subjects of articles nor has information regarding orphan drugs. However, combining both data could be beneficial for a researcher.

We considered that subject of a biomedical entity is also the subject of an article that refers to a related entity. DBpedia is utilized to fetch the subject of entities, where the relationship of chemical entities and subjects are represented by *dct:subject* predicate. Likewise, orphan drugs are considered as chemical substances that are subject to *dbc:Orphan_drugs* concept in DBpedia. In order to infer the subject of the articles, a rule is implemented as shown in Figure 5.6. Every chemical substance with *dct:subject* relationship is downloaded from DBpedia and imported into our data source to infer article subjects.

Listing 6.16 shows the related SPARQL query. This query fetches every article containing inhibition relation and chemicals used in orphan drugs. For this task, DBpedia was utilized to acquire orphan drug knowledge of chemical entities. Line 20 shows the triple to query chemicals with the subject of orphan drugs. Line 23 shows the triple to fetch article subjects that the reasoner engine has already inferred.

In Figure 6.15, the most popular subjects of articles that contain orphan drugs which inhibit protein entities are shown. In the header of accordions subject's labels are displayed. In the accordions, chemicals which are considered as *Orphan Drug* are displayed with a blue background, and inhibited proteins are displayed with red

font. *World Health Organization essential medicines* subject is one of the most popular subjects that referred in 4 different articles that contain relation of orphan drugs inhibiting proteins.

To perform the same task with *AA* representation, firstly, every annotation that contains inhibition of a protein should be fetched from the database. Later, every chemical substance that is a member of orphan drugs should be fetched from external sources, and annotations that contain this chemical should be filtered. Finally, the subject of every chemical in the remainder annotations should be fetched from an external source as well. Three development steps are needed to achieve the same results of ontology-based representation.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX bee: <http://soslab.cmpe.boun.edu.tr/ontologies/bee#>
PREFIX sio: <http://semanticscience.org/resource/>
PREFIX obo: <http://purl.obolibrary.org/obo/>
PREFIX dct: <http://purl.org/dc/terms/>
PREFIX dbc: <http://dbpedia.org/resource/Category:>

SELECT
    ?article ?subject ?chem ?prot
WHERE {
    ?article  rdf:type bee:Article;
              sio:SIO_000772 _:evi.

    _:evi bee:hasBiomedicalEntityPairRelation _:rel.
    _:rel bee:hasChemical ?chem;
          bee:hasProtein ?prot;
          bee:hasRelationType obo:MI_2274.

    GRAPH <https://dbpedia.org/sparql/> {
        _:c dct:subject dbc:Orphan_drugs ;
            owl:sameAs ?chem.
    }
    ?article dct:subject ?subject;
             FILTER(?subject != dbc:Orphan_drugs).
}
```

Figure 6.16. SPARQL query of Task 8 to fetch subjects of articles that refer to orphan drugs that inhibit proteins.

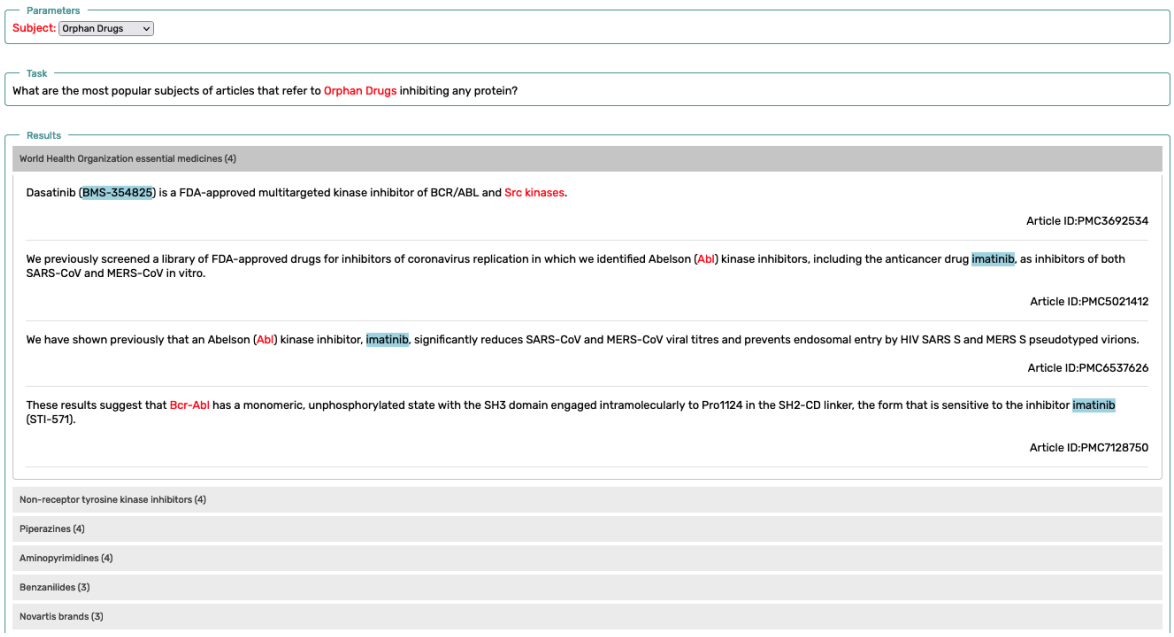

Figure 6.17. Most popular subjects of articles that contain orphan drugs that inhibit protein entities.

6.2.9. Task 9: Which proteins are related to Coronavirus-associated diseases?

This task queries the proteins that are related to Coronavirus-associated diseases. As stated before, this work does not extract the relations between proteins and diseases. However, a researcher may want to investigate the relation of a particular set of diseases and proteins. As well as protein-disease relation, this work also does not cover Coronavirus-associated diseases that must be obtained from an external source.

Due to ontology-based representation of diseases, coronavirus-associated disease knowledge utilized from DBpedia where every disease that have *dct:subject* relationship with *dbc:Coronavirus-associated_diseases* concept are considered as coronavirus-associated disease.

Disease concepts are represented by MeSH ontology in our data source and in DBpedia; thus, both data sources are available for connection over disease concepts. Besides the utilization of DBpedia, ontology-based representation enables inferring new relationships between every protein and disease entity that have a relationship with the same chemical. To explicitly represent relation, *bee:isRelated* relationship is inferred

between related protein and disease entities based on the predefined rule that is shown in Figure 5.2.

Listing 6.18 shows the related SPARQL query. Inferred relation of disease-protein entities is queried in Line 11. DBpedia data is linked into our data source over disease concept with *owl:sameAs* predicate as shown in Line 16. In Line 15, diseases that are subject to the Coronavirus-associated disease concept are selected. Before executing this query, we downloaded and imported DBPedia data into our data source and used it as a graph. Additionally, chemicals that cause the disease-protein relations are also returned to display to the user.

Figure 6.19 shows the example results of proteins related to Coronavirus-associated diseases. In the headers of accordions, protein labels are displayed. Inside the accordions, each row displays the related disease in a green round and the chemical that cause the relation between protein and disease in a blue round. In the first row of first accordion of Figure 6.19, relation between *Interleukin-12* protein and *Bronchiolitis* disease identified due to *Ala-Pro* chemical.

To achieve the same results with AA representation, firstly, every coronavirus-associated disease must be fetched from an external source. Later, every chemical-protein annotation and chemical-disease annotation containing any coronavirus-associated disease should be queried from the database. Finally, for every chemical, related proteins and diseases should be combined. To achieve the same results of ontology-based representation, a development to fetch external source and a development of algorithm to identify related protein and diseases are needed with AA representation.

```
PREFIX rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX bee:  <http://soslab.cmpe.boun.edu.tr/ontologies/bee#>
PREFIX dct: <http://purl.org/dc/terms/>
PREFIX dbc: <http://dbpedia.org/resource/Category:>
PREFIX owl: <http://www.w3.org/2002/07/owl#>

SELECT
    ?dis ?pro ?che
WHERE {

    ?dis bee:isRelated ?pro.
    ?pro rdf:type bee:Protein.

    GRAPH<https://dbpedia.org/sparql/>{
        _:d dct:subject dbc:Coronavirus-associated_diseases;
            owl:sameAs ?dis.
    }

    ?che bee:isRelated ?dis;
         bee:isRelated ?pro;
         rdf:type bee:Chemical.
}
```

Figure 6.18. SPARQL query of Task 9 to identify proteins that related to Coronavirus associated diseases.

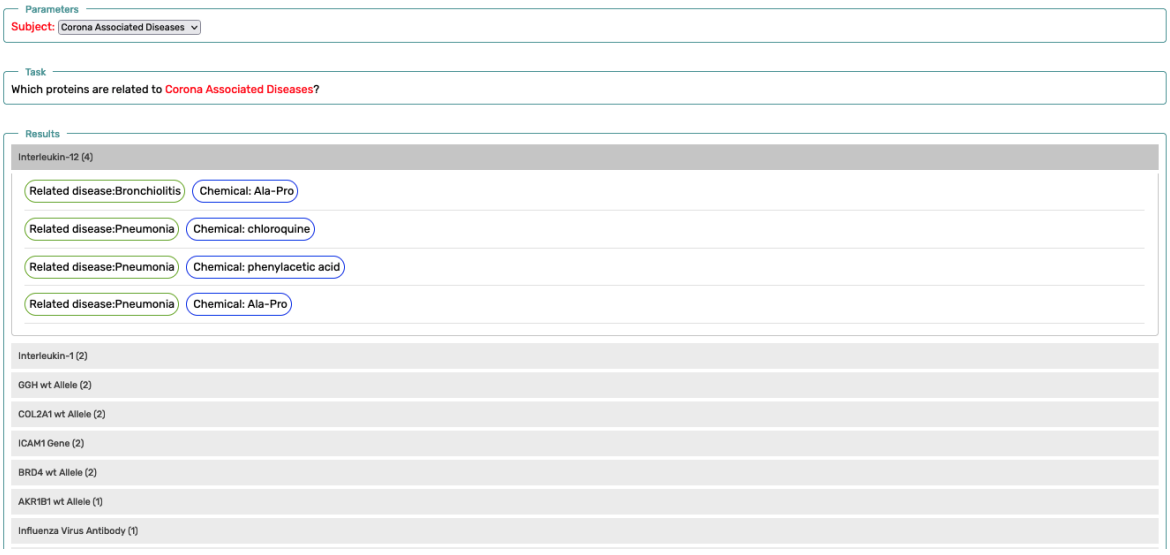

Figure 6.19. Proteins that are related to coronavirus-associated diseases and chemicals that cause the relation.

7. DISCUSSION AND FUTURE WORK

The proposed approach is discussed in this chapter, and several future directions are presented.

7.1. BEE Ontology

The main objective of this thesis was to semantically represent biomedical entity relations that occur in scientific articles in a machine-processable manner. This information is represented based on the proposed BEE ontology to benefit from conceptualization. Evaluations showed that ontology-based representation provided successful automated tasks of inferring new relationships and integrating Linked Open Data (LOD). BEE ontology enabled inference of relationships based on custom rules and class' hierarchy. Reusing existing biomedical ontologies in BEE enabled interoperability with knowledge sources like Wikidata and DBpedia.

Ontologies are developed to represent an area of concern, and designing a detailed ontology is challenging. Various stakeholders have to participate in the development, such as domain experts and ontology developers. Biomedical Entities Evidence(BEE) designed as a particular ontology to represent biomedical entity relation claims in scientific articles. Other than biomedical entity relation claims, many beneficial claims exist in scientific articles like electronic medical records and patient cases. Designing a general ontology to describe any claim that occurs in scientific articles can enable more powerful information representation and utilization. Single ontology can represent biomedical entity relations, electronic medical records (EMRs), and patient cases. As well as other claims, BEE ontology can also represent various different biomedical entity relations like chemical-protein pathways or chemical-disease multiclass relations.

BEE ontology represents provenance of biomedical entity relations as evidences. In this version of ontology, only entity's provenance information represented however

provenance of relation information can be represented as a future work. Such information will be useful for researchers.

Another future work about representation of provenance information is representing discontinuous entities. This version version only covers representation continous entities by expressing starting and ending position of whole phrase. In future works, *bee:Evidence* class can represent list of words that create discontinuous entity where each word's provenance represented separately.

7.2. Semantic Data Utilization

We developed a web application prototype that provides predefined information retrieval tasks to demonstrate the benefits of ontology-based semantic representation. Predefined tasks allow limited input types for each query. Although, allowing user-generated free text queries can enhance the user experience and allow more sophisticated queries to be built.

We presented the conceptualization benefits of ontology-based semantic representation. Besides, there are many statistical-based approaches to represent articles, such as modeling articles based on latent Dirichlet allocation to identify topics or representing articles based on deep learning to cluster according to their similarities. While ontology-based semantic representation can handle many data utilization tasks, to tackle problems like lack of semantic information incompleteness, ontology-based and statistical-based semantic representations can be used together to utilize information better.

7.3. Annotation of Articles

In order to represent information in scientific articles in a machine-processable manner based on ontologies, firstly, biomedical concepts and their relations in articles have to be extracted. Increasing the accuracy of named entity recognition, named entity

normalization, and relation extraction tasks in the article annotation pipeline directly affects the success of the utilization of knowledge since represented information is the extracted annotations. As a future work, state of the art techniques can be utilized for every annotation step and accuracy can be increased.

In the relation extraction task, only biomedical entities that occur in the same sentences are considered; however, developing a system to extract relations in different sentences can increase the number of captured and expressed data.

8. CONCLUSION

In this thesis, we introduced an approach to represent biomedical entity relations that occur in scientific articles semantically. For this purpose, we proposed Biomedical Entities Evidence(BEE) that represents chemical-protein and chemical-disease relations in scientific articles with evidence. BEE designed with the best practices of ontology development, and several biomedical ontologies are reused in the implementation. Ontology-based representation enabled inference of new data and interoperability with other knowledge sources across the Web.

We introduced an approach to annotating articles with biomedical entity relations evidence for ontology-based representation. In this approach, firstly, chemicals, proteins, and diseases occurring in a scientific article are identified. Later, identified entities normalized to concepts of each type's target ontology. Finally, biomedical relationship types are extracted for every chemical-protein and chemical-disease entity pair that occurs in the same sentence. For demonstration purposes, we extracted biomedical relations that occur in The Covid-19 Open Research Dataset(CORD-19) abstracts and represented based on BEE ontology.

We implemented a web application prototype to demonstrate the benefits of ontology-based representation of biomedical entity relations in scientific articles. The prototype provided several information retrieval tasks of predefined queries and presented the advantages of ontology-based semantic representation.

In conclusion, scientific articles contain useful embedded information not available for the automated use of computers. There are various approaches to extracting and using the information in scientific articles. Frequently, machine learning based methods are used in the area. However, knowledge in scientific articles can be utilized more effectively through ontologies for computers. Semantic representation of knowledge in articles based on ontologies provides many automated tasks. One task is to infer

new data, either ontology-based or rule-based. Another task is the interoperability with Linked Open Data (LOD) sources. Due to the representation of articles based on concepts and relationships, data sources across the Web can be connected to represented data, and those data sources' existing knowledge can be leveraged. We demonstrated chemical-protein and chemical-disease relations that occur in biomedical articles. However, any information could be represented using appropriate ontologies; thus, other data sources could be leveraged as well.

REFERENCES

1. Bornmann, L., R. Haunschild and R. Mutz, "Growth Rates of Modern Science: A Latent Piecewise Growth Curve Approach to Model Publication Numbers from Established and New Literature Databases", *Humanities and Social Sciences Communications*, Vol. 8, No. 1, pp. 1–15, 2021.

2. Chen, Q., A. Allot and Z. Lu, "LitCovid: An Open Database of COVID-19 Literature", *Nucleic Acids Research*, Vol. 49, No. D1, pp. 1534–1540, 2021.

3. PubMed, "PubMed", `https://pubmed.ncbi.nlm.nih.gov/`, accessed in December 2021.

4. Zhu, Z., J. Liang, D. Li, H. Yu and G. Liu, "Hot Topic Detection Based on a Refined TF-IDF Algorithm", *IEEE Access*, Vol. 7, pp. 26996–27007, 2019.

5. Cheng, X., Q. Cao and S. S. Liao, "An Overview of Literature on COVID-19, MERS and SARS: Using Text Mining and Latent Dirichlet Allocation", *Journal of Information Science*, pp. 1–17, 2020.

6. Shorten, C., T. M. Khoshgoftaar and B. Furht, "Deep Learning Applications for COVID-19", *Journal of Big Data*, Vol. 8, No. 1, pp. 1–54, 2021.

7. Jin, Q., Z. Yuan, G. Xiong, Q. Yu, H. Ying, C. Tan, M. Chen, S. Huang, X. Liu and S. Yu, "Biomedical Question Answering: A Survey of Approaches and Challenges", *ACM Computing Surveys*, Vol. 55, pp. 1–36, 2022.

8. Lung, P.-Y., Z. He, T. Zhao, D. Yu and J. Zhang, "Extracting Chemical–Protein Interactions from Literature Using Sentence Structure Analysis and Feature Engineering", *Database*, Vol. 2019, 2019.

9. Hurle, M., L. Yang, Q. Xie, D. Rajpal, P. Sanseau and P. Agarwal, "Computational

Drug Repositioning: From Data to Therapeutics", *Clinical Pharmacology & Therapeutics*, Vol. 93, No. 4, pp. 335–341, 2013.

10. Gruber, T. R., "A Translation Approach to Portable Ontology Specifications", *Knowledge Acquisition*, Vol. 5, No. 2, pp. 199–220, 1993.

11. Kim, D., J. Lee, C. So, H. Jeon, M. Jeong, Y. Choi, W. Yoon, M. Sung and J. Kang, "A Neural Named Entity Recognition and Multi-Type Normalization Tool for Biomedical Text Mining", *IEEE Access*, Vol. PP, p. 1, 2019.

12. Karadeniz, İ. and A. Özgür, "Linking Entities Through an Ontology Using Word Embeddings and Syntactic Reranking", *BMC Bioinformatics*, Vol. 20, No. 1, p. 156, 2019.

13. Whetzel, P. L., N. F. Noy, N. H. Shah, P. R. Alexander, C. Nyulas, T. Tudorache and M. A. Musen, "BioPortal: Enhanced Functionality via New Web Services from the National Center for Biomedical Ontology to Access and Use Ontologies in Software Applications", *Nucleic Acids Research*, Vol. 39, No. Web Server issue, pp. W541–5, 2011.

14. R., G. and V. Uma, "Ontology-based Knowledge Representation Technique, Domain Modeling Languages and Planners for Robotic Path Planning: A Survey", *ICT Express*, Vol. 4, No. 2, pp. 69–74, 2018.

15. Kamdar, M. R., J. D. Fernández, A. Polleres, T. Tudorache and M. A. Musen, "Enabling Web-scale Data Integration in Biomedicine Through Linked Open Data", *NPJ Digital Medicine*, Vol. 2, No. 1, p. 90, 2019.

16. Noy, N. and D. Mcguinness, "Ontology Development 101: A Guide to Creating Your First Ontology", *Knowledge Systems Laboratory*, Vol. 32, 2001.

17. Wang, L. L., K. Lo, Y. Chandrasekhar, R. Reas, J. Yang, D. Eide, K. Funk, R. Kinney, Z. Liu, W. Merrill, P. Mooney, D. A. Murdick, D. Rishi, J. Sheehan,

Z. Shen, B. Stilson, A. D. Wade, K. Wang, C. Wilhelm, B. Xie, D. Raymond, D. S. Weld, O. Etzioni and S. Kohlmeier, "CORD-19: The Covid-19 Open Research Dataset", *Computing Research Repository*, Vol. abs/2004.10706, 2020.

18. Berners-Lee, T., J. Hendler and O. Lassila, "The Semantic Web", *Scientific American*, Vol. 284, No. 5, pp. 34–43, 2001.

19. Yıldırım, A. and S. Uskudarli, "Microblog Topic Identification Using Linked Open Data", *PLOS One*, Vol. 15, No. 8, p. e0236863, 2020.

20. Roldan, M. d. M., S. Uskudarli, N. Marvasti, B. Acar and J. Aldana Montes, "Towards an Ontology-Driven Clinical Experience Sharing Ecosystem: Demonstration with Liver Cases", *Expert Systems with Applications*, Vol. 101, 2018.

21. Aggelen, A., L. Hollink, M. Kemman, M. Kleppe and H. Beunders, "The Debates of the European Parliament as Linked Open Data", *Semantic Web*, Vol. 8, pp. 271–281, 2016.

22. Jovanović, J. and E. Bagheri, "Semantic Annotation in Biomedicine: The Current Landscape", *Journal of Biomedical Semantics*, Vol. 8, No. 1, 2017.

23. Jonquet, C., N. H. Shah and M. A. Musen, "The Open Biomedical Annotator", *Summit on Translational Bioinformatics*, Vol. 2009, pp. 56–60, 2009.

24. Dai, M., N. H. Shah, W. Xuan, M. A. Musen, S. J. Watson, B. Athey and F. Meng, "An Efficient Solution for Mapping Free Text to Ontology Terms", *AMIA Summit on Translational Bioinformatics*, 2008.

25. Müller, H.-M., E. E. Kenny and P. W. Sternberg, "Textpresso: An Ontology-based Information Retrieval and Extraction System for Biological Literature", *PLoS Biology*, Vol. 2, No. 11, p. e309, 2004.

26. Ashburner, M., C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin and G. Sherlock, "Gene Ontology: Tool for the Unification of Biology. The Gene Ontology Consortium", *Nature Genetics*, Vol. 25, No. 1, pp. 25–29, 2000.

27. Köksal, A., H. Dönmez, R. Özçelik, E. Ozkirimli and A. Özgür, "Vapur: A Search Engine to Find Related Protein - Compound Pairs in COVID-19 Literature", *Computer Research Repository*, Vol. abs/2009.02526, 2020.

28. Wang, Y., J. Xiao, T. O. Suzek, J. Zhang, J. Wang and S. H. Bryant, "PubChem: A Public Information System for Analyzing Bioactivities of Small Molecules", *Nucleic Acids Research*, Vol. 37, No. Web Server issue, pp. W623–33, 2009.

29. Pubchem, "About Pubchem", `https://pubchemdocs.ncbi.nlm.nih.gov/about`, accessed in December 2021.

30. Fu, G., C. Batchelor, M. Dumontier, J. Hastings, E. Willighagen and E. Bolton, "PubChemRDF: Towards the Semantic Annotation of PubChem Compound and Substance Databases", *Journal of Cheminformatics*, Vol. 7, No. 1, p. 34, 2015.

31. Pubchem, "RDF", `https://pubchemdocs.ncbi.nlm.nih.gov/rdf$_5-2`, accessed in December 2021.

32. Piñero, J., À. Bravo, N. Queralt-Rosinach, A. Gutiérrez-Sacristán, J. Deu-Pons, E. Centeno, J. García-García, F. Sanz and L. I. Furlong, "DisGeNET: A Comprehensive Platform Integrating Information on Human Disease associated Genes and Variants", *Nucleic Acids Research*, Vol. 45, No. D1, pp. D833–D839, 2017.

33. Bravo, À., J. Piñero, N. Queralt-Rosinach, M. Rautschka and L. I. Furlong, "Extraction of Relations Between Genes and Diseases from Text and Large-scale

Data Analysis: Implications for Translational Research", *BMC Bioinformatics*, Vol. 16, No. 1, p. 55, 2015.

34. Queralt-Rosinach, N., J. Piñero, l. Bravo, F. Sanz and L. I. Furlong, "DisGeNET-RDF: Harnessing the Innovative Power of the Semantic Web to Explore the Genetic Basis of Diseases", *Bioinformatics*, Vol. 32, No. 14, pp. 2236–2238, 2016.

35. DisGeNET, "DisGeNET RDF", `https://www.disgenet.org/rdf`, accessed in December 2021.

36. DisGeNET, "DisGeNET Sparql Endpoint", `http://rdf.disgenet.org/sparql/`, accessed in December 2021.

37. W3C, "Vocabularies", `https://www.w3.org/standards/semanticweb/ontology`, accessed in December 2021.

38. W3C, "RDF", `https://www.w3.org/RDF/`, accessed in December 2021.

39. Wikipedia Contributors, "Resource Description Framework Information", `https://en.wikipedia.org/wiki/Resource_Description_Framework`, accessed in December 2021.

40. Antoniou, G. and F. Van Harmelen, "Web Ontology Language: Owl", *Handbook on Ontologies*, pp. 67–92, Springer, 2004.

41. W3C, "Owl", `https://www.w3.org/OWL/`, accessed in December 2021.

42. W3C, "SPARQL Query Language for RDF Overview", `https://www.w3.org/TR/2013/REC-sparql11-overview-20130321/`, accessed in December 2021.

43. Wikipedia Contributors, "SPARQL", `https://en.wikipedia.org/wiki/SPARQL`, accessed in December 2021.

44. Wikipedia Contributors, "Turtle (Syntax)", https://en.wikipedia.org/wiki/Turtle_(syntax), accessed in December 2021.

45. W3C, "Turtle", https://www.w3.org/TR/turtle/, accessed in December 2021.

46. Ontotext, "Introduction to the Semantic Web", https://graphdb.ontotext.com/documentation/standard/introduction-to-semantic-web.html\#introduction-to-semantic-total-materialization, accessed in December 2021.

47. W3C, "Inference", https://www.w3.org/standards/semanticweb/inference.html, accessed in December 2021.

48. Ontotext, "General", https://graphdb.ontotext.com/documentation/free/, accessed in December 2021.

49. Bizer, C., T. Heath, K. Idehen and T. Berners-Lee, "Linked Data on the Web (LDOW2008)", *Proceedings of the 17th International Conference on World Wide Web*, WWW '08, p. 1265–1266, Association for Computing Machinery, New York, NY, USA, 2008.

50. W3C, "Data", https://www.w3.org/standards/semanticweb/data, accessed in December 2021.

51. Wikipedia Contributors, "Wikidata Information", https://en.wikipedia.org/wiki/Wikidata, 2021, accessed in December 2021.

52. Wikidata, "Statistics of Wikidata", https://www.wikidata.org/wiki/Wikidata:Statistics/en, accessed in December 2021.

53. Wikidata, "Wikidata Query Service", https://query.wikidata.org/, accessed in December 2021.

54. Devlin, J., M. Chang, K. Lee and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding", *Computing Research Repository*, Vol. abs/1810.04805, 2018.

55. Degtyarenko, K., P. de Matos, M. Ennis, J. Hastings, M. Zbinden, A. McNaught, R. Alcántara, M. Darsow, M. Guedj and M. Ashburner, "ChEBI: A Database and Ontology for Chemical Entities of Biological Interest", *Nucleic Acids Research*, Vol. 36, No. 1, pp. D344–D350, 2007.

56. EBI Web Team, "Chebi", `https://www.ebi.ac.uk/chebi/statisticsForward.do`, accessed in December 2021.

57. Fragoso, G., S. de Coronado, M. Haber, F. Hartel and L. Wright, "Overview and Utilization of the NCI Thesaurus", *Comparative and Functional Genomics*, Vol. 5, No. 8, pp. 648–654, 2004.

58. National Library of Medicine, "Umls Metathesaurus - NCI (NCI Thesaurus) - Statistics", `https://www.nlm.nih.gov/research/umls/sourcereleasedocs/current/NCI/stats.html`, accessed in December 2021.

59. Lipscomb, C. E., "Medical Subject Headings (MeSH)", *Bulletin of the Medical Library Association*, Vol. 88, No. 3, pp. 265–266, 2000.

60. National Library of Medicine, "National Library of Medicine - National Institutes of Health", `https://www.nlm.nih.gov/`, accessed in December 2021.

61. National Library of Medicine, "Frequently Asked Questions About Indexing", `https://www.nlm.nih.gov/bsd/indexfaq.html\#keywords`, accessed in December 2021.

62. Van Roey, K., S. Orchard, S. Kerrien, M. Dumousseau, S. Ricard-Blum, H. Hermjakob and T. J. Gibson, "Capturing Cooperative Interactions with the PSI-MI Format", *Database*, Vol. 2013, 2013.

63. Dumontier, M., C. J. Baker, J. Baran, A. Callahan, L. Chepelev, J. Cruz-Toledo, N. R. Del Rio, G. Duck, L. I. Furlong, N. Keath, D. Klassen, J. P. McCusker, N. Queralt-Rosinach, M. Samwald, N. Villanueva-Rosales, M. D. Wilkinson and R. Hoehndorf, "The Semanticscience Integrated Ontology (SIO) for Biomedical Research and Knowledge Discovery", *Journal of Biomedical Semantics*, Vol. 5, No. 1, p. 14, 2014.

64. Kiryakov, A., B. Popov, I. Terziev, D. Manov and D. Ognyanoff, "Semantic Annotation, Indexing, and Retrieval", *Journal of Web Semantics*, Vol. 2, No. 1, pp. 49–79, 2004.

65. Neumann, M., D. King, I. Beltagy and W. Ammar, "ScispaCy: Fast and Robust Models for Biomedical Natural Language Processing", *Proceedings of the 18th BioNLP Workshop and Shared Task*, pp. 319–327, Association for Computational Linguistics, 2019.

66. Lample, G., M. Ballesteros, S. Subramanian, K. Kawakami and C. Dyer, "Neural Architectures for Named Entity Recognition", *Computing Research Repository*, Vol. abs/1603.01360, 2016.

67. Pyysalo, S., T. Ohta, R. Rak, A. Rowley, H.-W. Chun, S.-J. Jung, S.-P. Choi, J. Tsujii and S. Ananiadou, "Overview of the Cancer Genetics and Pathway Curation Tasks of BioNLP Shared Task 2013", *BMC Bioinformatics*, Vol. 16, No. S10, p. S2, 2015.

68. Li, J., Y. Sun, R. J. Johnson, D. Sciaky, C.-H. Wei, R. Leaman, A. P. Davis, C. J. Mattingly, T. C. Wiegers and Z. Lu, "BioCreative V CDR Task Corpus: A Resource for Chemical Disease Relation Extraction", *Database*, Vol. 2016, 2016.

69. Zhang, Y., Q. Chen, Z. Yang, H. Lin and Z. Lu, "BioWordVec, Improving Biomedical Word Embeddings with Subword Information and MeSH", *Scientific Data*, Vol. 6, No. 1, p. 52, 2019.

70. Chen, Q., "NCBI-NLP/BioSentVec: BioWordVec and BioSentVec: Pre-trained Embeddings for Biomedical Words and Sentences", `https://github.com/ncbi-nlp/BioSentVec`, 2020, accessed in December 2021.

71. Wikipedia Contributors, "Cosine Similarity", `https://en.wikipedia.org/wiki/Cosine_similarity`, 2021, accessed in December 2021.

72. Wang, Q., Z. Mao, B. Wang and L. Guo, "Knowledge Graph Embedding: A Survey of Approaches and Applications", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 29, No. 12, pp. 2724–2743, 2017.

73. Lee, J., W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So and J. Kang, "BioBERT: a Pretrained Biomedical Language Representation Model for Biomedical Text Mining", *Bioinformatics*, Vol. 36, No. 4, pp. 1234–1240, 2019.

74. Krallinger, M., O. Rabal, S. A. Akhondi, M. P. Pérez, J. Santamaría, G. P. Rodríguez, G. Tsatsaronis, A. Intxaurrondo, J. A. B. López, U. K. Nandal, E. M. van Buel, A. Chandrasekhar, M. Rodenburg, A. Lægreid, M. A. Doornenbal, J. Oyarzábal, A. Lourenço and A. Valencia, "Overview of the BioCreative VI Chemical-Protein Interaction Track", *Proceedings of the Sixth BioCreative Challenge Evaluation Workshop*, Vol. 1, pp. 141–146, 2017.

75. W3C, "SPARQL Query Language Federated Query", `https://www.w3.org/TR/2013/REC-sparql11-federated-query-20130321/`, accessed in December 2021.

76. W3C, "W3C", `https://www.w3.org/`, accessed in December 2021.