

Knowledge Detection and Discovery using Semantic Graph Embeddings on Large Knowledge Graphs generated on Text Mining Results

Jens Dörpinghaus*[†], Marc Jacobs[†]

* German Center for Neurodegenerative Diseases (DZNE), Bonn, Germany, Email: jens.doerpinghaus@dzne.de

[†] Fraunhofer Institute for Algorithms and Scientific Computing, Schloss Birlinghoven, Sankt Augustin, Germany

Abstract—Knowledge graphs play a central role in big data integration, especially for connecting data from different domains. Bringing unstructured texts, e.g. from scientific literature, into a structured, comparable format is one of the key assets. Here, we use knowledge graphs in the biomedical domain working together with text mining based document data for knowledge extraction and retrieval from text and natural language structures. For example cause and effect models, can potentially facilitate clinical decision making or help to drive research towards precision medicine. However, the power of knowledge graphs critically depends on context information. Here we provide a novel semantic approach towards a context enriched biomedical knowledge graph utilizing data integration with linked data applied to language technologies and text mining. This graph concept can be used for graph embedding applied in different approaches, e.g. with focus on topic detection, document clustering and knowledge discovery. We discuss algorithmic approaches to tackle these challenges and show results for several applications like search query finding and knowledge discovery. The presented remarkable approaches lead to valuable results on large knowledge graphs.

I. INTRODUCTION

IN THIS paper we will present a novel approach towards knowledge detection and discovery using semantic graph embeddings on large knowledge graphs. The idea of semantic graph embeddings was initially introduced in [1], the theoretical background in [2] and the algorithms which are used as a basis for our approach were introduced in [3]. Combining these results, we will present a novel heuristic approach and present experimental results on a large scale knowledge graph from the biomedical field, see [4]. This graph is build upon text mining results on biomedical literature databases. The real-world use cases were collected from scientific projects.

A knowledge graph has a comprehensible topological representation given by nodes and edges, but this is usually not a very precise representation of the real world. A more generic approach can be constructed by using classes. Thus the basic idea is to divide a knowledge graph in different knowledge layers either directly given by the data (like documents, authors) or manually defined. For example biological relations might be associated with an ontology (ontology layer), they can be annotated to a document with named entity recognition (NER, annotation layer) and they might belong to a domain specific language layer (for example BEL, biological expression language, layer). See figure 1 for an illustration.

This approach is similar to the idea of molecular information layers described in [5]. To sum up, we build linked data from different data sources and ontologies. We use text mining and natural language processing approaches to make these linked data information interoperable, findable and re-usable. Thus, every data type from a data source implies a different layer and those layers are either linked with relations given in the data source or by text mining.

The testing system is based on Neo4j and holds a dense large scale labeled property graph with more than 75M nodes and 960M edges. This graph is based on biomedical knowledge graphs as described in [6] and [7].

This paper is divided into six sections. The first section gives a brief overview of the state of the art and related work. The second section describes the theoretical background and the methods used for our novel approach. We will introduce knowledge graphs, semantic graph embeddings and algorithms. In the third section, we present applications from real world use cases like search query finding and generating and optimisation of cluster labels. The fourth section is dedicated to experimental results on artificial and real-world scenarios. Our conclusions are drawn in the final section.

We will propose two novel algorithmic approaches which present promising performance. The results show a significant improvement over the existing engine without using context information.

II. RELATED WORK

In recent decades the field of natural language processing (NLP) and knowledge discovery as well as the related fields data mining and the management of information systems is emerging. Several authors like Manning et al. [8] or Clarc et al. [9] give an overview about the algorithmic part of computational linguistics and NLP. In addition there is a constant interest in using graphs for these problems, see [10].

In scientific research, expert systems provide users with several methods for knowledge discovery. They are widely used to find relevant or novel information. For example, medical and biological researchers try to find molecular pathways, mechanisms within living organisms or special occurrences of drugs or diseases. Using expert knowledge as an input, researches usually consider an initial idea and some content like papers or other documents. The most common approach

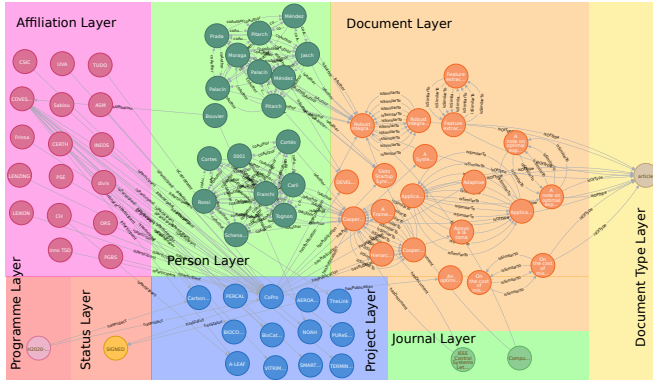


Fig. 1: (Illustration of some knowledge graph layers found in the testing environment. Here, we can see some document-specific layers which are combined from several data sources (PubMed, DBLP, H2020): Document Type Layer, Journal Layer, Person (Author) Layer. Other layers are specific to the H2020 project data obtained from EU Open Data Portal: Project Layer, Status Layer, Programme Layer and the Affiliation Layer. We notice several intersections, for example Quentin_Bouvier is no Author, but has both an affiliation and is associated with the project NOAH.

is inquiring a search engine to find closely related information. Thus two question are most frequently asked: "How can I find these documents?" to adjust the search query for knowledge discovery or "What are these documents all about?" to find the topic. Both questions are heavily related to the context of documents. Meta-data like authors, keywords and text are used to retrieve results of a query using a search engine. Current research in NLP and text mining usually does not directly focus on finding a search query from a given corpus, although a lot of research has been done on the analyses of a given search query, see [11] or the analyses of queries on different databases, see for example [12] for PubMed data. Topic labeling – or cluster labeling – is under constant research in several research areas.

There is a considerable amount of literature on both problems. Many studies have been published on probabilistic or machine-learning-approaches, see [13], [14] or [15]. In addition, in recent years there has been growing interest in providing users with suggestions for more specific or related search queries, see [16]. We already mentioned [17] but most research focuses on artificial intelligence (AI), machine learning (ML) or deep learning (DL) approaches, see [18] or [19]. Our aim is a precise solution without a prior learning step giving a deeper insight in the data and the context of this data.

Here, knowledge graphs are becoming a key instrument for knowledge discovery and modeling. These approaches rely on structured data, e.g. about related proteins or genes, and form cause-and-effect networks or – if enriched with literature data and other linked datasources – knowledge graphs. A key aspect of analysis on these graphs is the missing context.

III. METHOD

A. Knowledge Graph

Knowledge graphs play in general an important role in recent knowledge mining and discovery. A *knowledge graph* (sometimes also called a *semantic network*) is a systematic way to connect information and data to knowledge on a more abstract level than language graphs. It is thus a crucial concept on the way to generate knowledge and wisdom, to search within data, information and knowledge. The context is a significant topic to generate knowledge or even wisdom. Thus, connecting knowledge graphs with context is a crucial feature.

Many authors tried to give a definition of knowledge graphs, but still a formal definition is missing, see [20]. In [21] the authors compared several definitions, but the only formal definition was related to RDF graphs which does not cover labeled property graphs. Thus, here we propose a very general definition of a knowledge graph using graph theory:

Definition III.1. (Knowledge Graph) We define a knowledge graph as graph $G = (E, R)$ with entities $e \in E = \{E_1, \dots, E_n\}$ coming from a formal structure E_i like ontologies.

The relations $r \in R$ can be ontology or layer relations (like "is related to" or "is co-Author"), thus in general we can say every formal structure E_i which is part of the data model is a subgraph of G indicating $O \subseteq G$. In addition, we allow inter-structure relations between two nodes e_1, e_2 with $e_1 \in E_1$, $e_2 \in E_2$ and $O_1 \neq E_2$. In more general terms, we define $R = \{R_1, \dots, R_n\}$ as a list of either inter-structure or inner-structure relations. Both E as well as R are finite discrete spaces. See figure 3 for an example.

Every entity $e \in E$ may have some additional meta information which needs to be defined with respect to the application of the knowledge graph. For instance, there may be several node sets (some ontologies, some document spaces (patents, research data, ...), author sets, journal sets, ...) E_1, \dots, E_n so that $E_i \subset E$ and $E = \cup_{i=1, \dots, n} E_i$. The same holds for R when several context relations come together such as "is cited by", "has annotation", "has author", "is published in", etc.

The basis for generating our large-scale Knowledge Graph representation is biomedical literature (e.g. from PubMed and PMC). We also integrated bibliographic data and metadata from DBLP, monthly snapshot release of December 2019, see <https://dblp.uni-trier.de/> and [22]. Since the basic data coming from SCAIView is already annotated with different biomedical ontologies, we decided to use the CSO classifier (see [23]) to annotate CSO to DBLP data.

We enriched our graph with data from the EU Open Data Portal (CORDIS - EU research projects under Horizon 2020, see <https://data.europa.eu/euodp/en/data/dataset/cordisH2020projects>). This data set is free to reuse for both commercial or non-commercial purpose. Here, we integrated projects, their status, affiliations, persons and authors of publications mentioned in their data set.

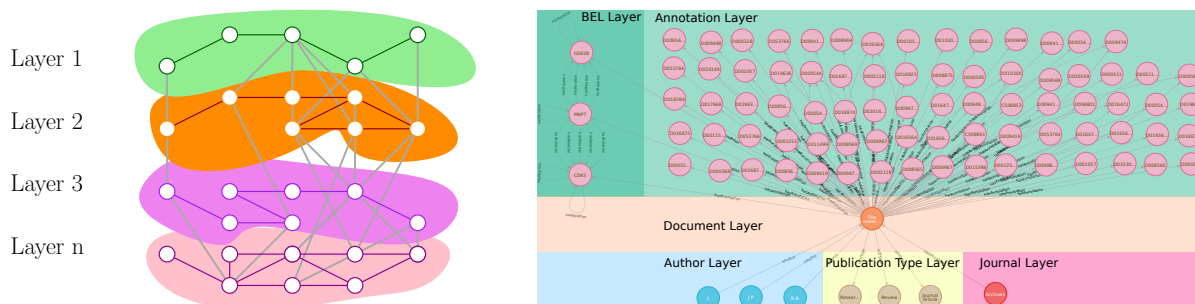


Fig. 2: (left) Illustration of the *knowledge graph embedding* between different layers. Here, every layer corresponds to a context defining of new contexts on several other layers. Thus layers and contexts are flexible and can be defined in a feasible way for every application. Data within the Knowledge Graph can be ordered according to context and information to data layers (e.g. a molecular or mechanism layer). This helps to examine novel causal connections and context. Layer 1 defines *Macro-Context* as Information Highway. The ordering of layers is based on the questions asked. It may also be used to allow an easy and FAIR access to the data and benefit from semantic graph-queries. Date integration, adding more data will increase the Knowledge-Foundation and gives a more precise view on the micro-context and helps to unveil new context and insights. This is a method from top to bottom, the other direction is dedicated to Data Mining. (right) Example illustration of different layers obtained by document *The molecular bases of Alzheimer's disease and other neurodegenerative disorders* (PMID:11578751).

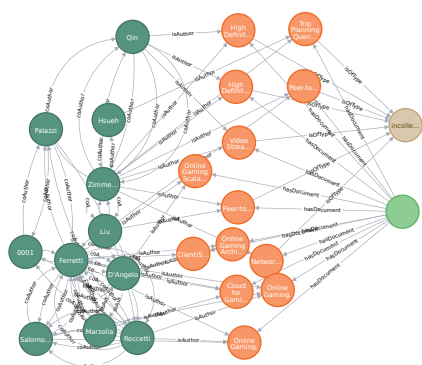


Fig. 3: Illustrations of inter-structure and inner-structure relations. Here, E_1 (left) is a structure containing authors, having an inner relation indicating co-authors. E_2 (orange) is a structure containing documents with an inner relation indicating for example citations. We can see inter-structure relations between both structures indicating authorship.

The articles or abstracts are the source for biological relations. In addition, meta information like authors, journals, keywords, etc. are available. Ontologies can be used to contextualize entities in the knowledge graph providing biological or medical relations. Every ontology will form another knowledge (sub-)graph. Using methods of natural language processing (NLP) and text mining, we can combine and link these knowledge graphs to a giant and very dense new knowledge graph. This will meet a very general definition of context. We can see every knowledge (sub-)graph as context to another. Biological expressions are context of the corresponding literature, authors are context of a text, named entities from ontologies found in a text are context to it or to

the corresponding biological expression.

Several ontologies and terminologies were added to the knowledge graph, for example Computer Science Ontology (CSO, see <http://cso.kmi.open.ac.uk/home>), HUGO Gene Nomenclature Committee (HGNC, see [24]), Gene Ontology (GO, see [25] and [26]) or Disease Ontology (DO, see [27]). These ontologies can be used to annotate context with methods from text mining to data entities within the graph, see [6].

B. Semantic Graph Embeddings

Semantic graph embeddings are closely related to the concept of context. Here, we use a quite general definition of context data. We assume that every information entity can also be a context information for other entities. For example a document can also be a context for other documents (e.g. by citing or referring to the other publication). An author is both a meta information to a document, but also itself context (by other publications, affiliations, co-author networks, ...). Other data is more obvious a context: named entities, topic maps, keywords, etc. extracted with text mining from documents. But already relations extracted from a text may stand for themselves, occurring in multiple documents and still valuable without the original textual information.

Definition III.2. (*Context*) We define context C as a set with context subsets $C = \{c_1, \dots, c_m\}$. This is a finite, discrete set. Every node $v \in G$ and every edge $r \in R$ may have one or more contexts $c \in C$ denoted by $con(v) \subset G$ or $con(r) \subset G$.

It is also possible to set $con(v) = \emptyset$. Thus we have a mapping $con : E \cup R \rightarrow \mathcal{P}(C)$. If we use a quite general approach towards context, we may set $C = E$. Therefore, every inter-ontology relation defines context of two entities, but also the relations within an ontology can be seen as context,

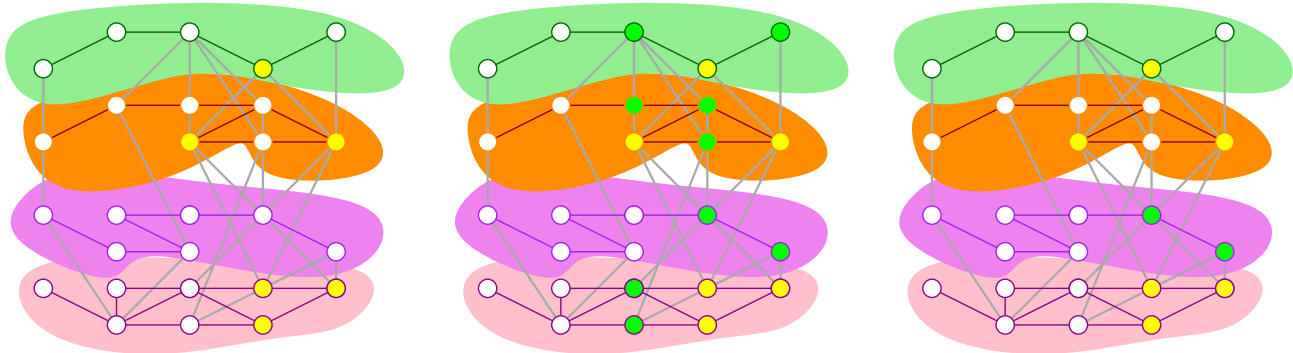


Fig. 4: Illustration of the steps for generating a semantic graph embedding \mathcal{E}_L where N is given by the yellow nodes and L is given by the pink layer (1), see example III.5. Subfigure (2) depicts the output of algorithm 1, $\mathcal{E}(N) = N \cup \text{con}(N)$. Limiting this to L returns $\mathcal{E}_L = \mathcal{E}(N) \cap L$, see subfigure (3).

With the neighborhood $N(E_i)$ every node set $E_i \in \{E_1, \dots, E_n\}$ induces a subgraph $G[E_i] \subset G$:

Definition III.3. (Semantic Graph Embeddings) With $G^c[E_i] = G[E_i] \cup N(E_i)$ we denote the extended context subgraph or semantic graph embedding which also contains the neighbors of each node in G , which is context of that node. With $G_L^c[E_i] = G^c[E_i] \cap L$ we denote the graph embedding on layer $L \subset G$.

To make the notation easier, we set $\mathcal{E}(N) = G^c[N]$ and $\mathcal{E}_L(N) = G_L^c[N]$.

For a graph drawing perspective, if $G^c[E_i]$ defines a proper surface, we can think about a graph embedding of another subgraph $G^c[E_j]$ on $G^c[E_i]$. This concept was introduced in [1]. Here, semantic knowledge graph embeddings were displayed between different layers. Every layer (for example: molecular layer, document layer, mechanism layer) corresponds to another context defining new contexts on other layers.

Example III.4. Consider the illustration in figure 2: Here we can see, that every subgraph L' of a layer L_1, \dots, L_n has an extended context subgraph $G^c[L'] = G[L'] \cup N(L')$ in multiple layers. In addition, if we have a set of nodes L'' in multiple layers L_i, L_j the same holds. Thus to see the embedding on just one layer L_i we can limit this set using $G_{L_i}^c[L'] = G^c[L'] \cap L_i$.

If the mapping con is well defined for the domain set, then Graph H can be generated in polynomial time. Since this is generally not the case, this step usually contains data or text mining task to generate other contexts from free texts or knowledge graph entities. With respect to the notation described in [2] this problem p can be formulated as

$$p = \mathbb{D} | R | \mathbf{f} : \mathbb{D} \rightarrow \mathbb{X} | \text{err} | \emptyset \quad (1)$$

Here, the domain set \mathbb{D} is explicitly given by $\mathbb{D} = G$ or – if additional full-texts \hat{D} supporting the knowledge Graph G exist – $\mathbb{D} = \{G, \hat{D}\}$, which in our case is the domain subset $R = \mathbb{D}$. Therefore, we need to find a description function

$f : \mathbb{D} \rightarrow \mathbb{X}$ with a description set $\mathbb{X} = C$ which holds all contexts. To find relevant contexts, we also need to measure the error as defined by $\text{err} : \mathbb{D} \rightarrow [0, 1]$.

C. Heuristic

To solve the knowledge graph embedding problem, we will use an extended version of algorithm 1 introduced in [3] within the field of document set cover. In our case, the input documents $\{d_1, \dots, d_n\} \subset \mathbb{D}$ can be seen as any elements or nodes $\{n_1, \dots, n_n\} \subset V$. The descriptive elements $f(d_i) = \{x_1, \dots, x_m\} \subset \mathbb{X}$ are now given by the context $\text{con}(n_i) = \{c_1, \dots, c_m\} \subset V$. See algorithm 1 for pseudocode.

Algorithm 1 s-GRAPH-EMBEDDING

Require: $N = \{n_1, \dots, n_n\} \subset V$ and descriptive elements $\text{con}(n_i) = \{c_1, \dots, c_m\} \subset V$, maxiter as maximum of iterations, s as sensitivity

Ensure: A semantic graph embedding $\mathcal{E}(N) = (V', E')$ of N with elements in V .

```

con' = con
2: for every v ∈ N do
   while iteration < maxiter AND con'(v) > (s · con(d))
   do
4:   remove c ∈ con'(v) with maximum weight
   end while
6: end for
return  $\mathcal{E}(N) = (\{c, \forall c \in \text{con}'(n)\} \cup \{n, \forall c \in \text{con}'(n) \forall n \in N\}, \{(c, n), \forall c \in \text{con}'(n) \forall n \in N\})$ 

```

Example III.5. See the example in figure 4. Here, we use algorithm 1 to compute a semantic graph embedding \mathcal{E}_L where N is given by the yellow nodes and L is given by the pink layer. We set $s = 1$ and $\text{maxiter} = 1$. The context in this example is defined as neighborhood in the graph, thus $\text{con}(v) = N(v)$. Algorithm 1 outputs both yellow and green nodes, which is $N \cup \text{con}(N)$. In this simplified example algorithm 1 returns $\mathcal{E}(N) = N \cup \text{con}(N)$. Limiting the graph embedding to the pink layer leads to $\mathcal{E}_L = \mathcal{E}(N) \cap L$.

If we use documents for the input N and only keywords as descriptive elements, algorithm 1 works exactly the same as described in [3]. Thus, our approach is a generalization of the initial algorithm to all descriptive elements found in any descriptive layer in a knowledge graph. Again we can argue, that that – while not limiting to a distinct layer – the algorithm outputs at least the initial nodes given in N . If the sensitivity is decreased to $s < 1$ we can see that less and less descriptive elements are chosen. In the next section, we will explain how to use this semantic graph embedding to knowledge discovery within the knowledge graph.

IV. APPLICATION

The initial research question was how to apply a general context added to biomedical knowledge graphs to answer several generic questions dedicated to knowledge discovery. As described above we have integrated several sources of publication data (PubMed, DBLP, H2020), several ontologies like GO, HGNC and mappings, BEL networks from Parkinson’s and Alzheimer’s disease as well as other structured data.

A. Search Query Finding and Knowledge Discovery

In [2] we proposed a very generic definition of search engines and search queries. Here, we will show, how this generic approach can be used to create real world search queries. A search engine is a function $q : \mathbb{X} \rightarrow \mathbb{D}$ which outputs a set of documents or any other content of the domain set if the input is a subset of a description set \mathbb{X} which we call search query. With this, it follows that the problem of finding a search query is given by

$$p = \mathbb{D}|R|\mathbb{X}|err|R$$

Given a knowledge graph $G = (V, E)$ with layers L_1, \dots, L_n . We denote L_D with the document layer. Let $D' \subset L_D$ be an initial set of documents, and let $\mathcal{E}_{L_D}(N) = D'$ be the semantic graph embedding on L_D . Thus, $\mathcal{E}(N) \cap L_D$ holds all descriptive elements of all documents in D' in other layers. If all layers can be used to search for documents, this returns a search query for D' . In [3] we proved this concept for one single layer containing keywords.

In order to get a feasible search query, we need to modify algorithm 1. In algorithm 2 we propose a generic approach not limited to a distinct layer returning a logical concatenation of nodes that are related to the semantic graph embedding. We call this a *semantic graph description* of D' .

Changing the value of s makes the search query more or less precise which helps with respect to knowledge discovery. For example, given a set of documents we may use them as seed to discover more related documents. Here, choosing the right description layers is quite important.

B. Generating and optimisation of Cluster Labels

In [2] we proposed a very generic approach towards cluster labeling. Given a knowledge graph $G = (V, E)$ finding cluster labels for clusters C_1, \dots, C_n is the task of assigning a subset of a description set \mathbb{X} , in our case on or more layers, with the

Algorithm 2 s -GRAPH-DESCRIPTION

Require: $N = \{n_1, \dots, n_n\} \subset L$ and descriptive elements $con(n_i) = \{c_1, \dots, c_m\} \subset V$, maxiter as maximum of iterations, s as sensitivity
Ensure: A semantic graph description $\mathcal{E}(N) = (V', E')$ of N with elements in $V \cap L$.
 $con' = con$
2: **for** every $v \in N$ **do**
 while iteration < maxiter AND $con'(v) > (s \cdot con(v))$
 do
4: remove $c \in con'(v)$ with maximum weight
 end while
6: **end for**
return $Z = \bigvee_{v \in N} (\bigwedge_{x \in con'(v)})$

description function $f : \mathbb{V} \rightarrow \mathbb{X}$ to a cluster $C \in \{C_1, \dots, C_n\}$. Thus, this problem is given by

$$p = \mathbb{D}|C|\mathbb{X}|err|R$$

where the resulting label set is the image $f(C) \subset X$. Depending on the choices of different layers to be included in \mathbb{X} this either leads to a set of metadata, terms from ontologies, sentences or any subset of natural language.

Once again we can apply the modified algorithm 2. As input, we use a set of nodes forming a cluster $C \subset G$. The return value needs to be filtered according to our choice of \mathbb{X} . As suggested in [3] we can either transform the logical operators to language (term x and term y or term z) or use a very low threshold which will lead to very small return value and return a ranked list of terms.

C. Document or Data Clustering

Document or data clustering is a specific application of text or data mining and a sub-problem of cluster analyses. Without any clusters pre-defined the goal is to cluster documents or data points to clusters sharing common features. Limiting the layers to documents will result in document clustering. If the knowledge graph layers contain any data points, this will result in data clustering. The application of clustering is a wide and open field and in terms of complexity it is still under heavy research, see for example [28] and [29].

Clustering is usually not perceived as a graph problem, although several attempts have been made (e.g. [30]) and here we will show how to generalize it on knowledge graphs. Usually the problem can be formulated in the following way: Given a similarity function for the document or data space D as $sim : D \times D \rightarrow \mathbb{R}^+$ and an $\epsilon \in \mathbb{R}^+$. We search for a minimal number of clusters, so that every two documents x, y in one cluster have $sim(x, y) \geq \epsilon$. For technical terms we refer to [8].

One common problem is to find sim . Here, the inverse problem helps: Given two data points d_1, d_2 they can be interpreted as an embedding of different layers. Thus by

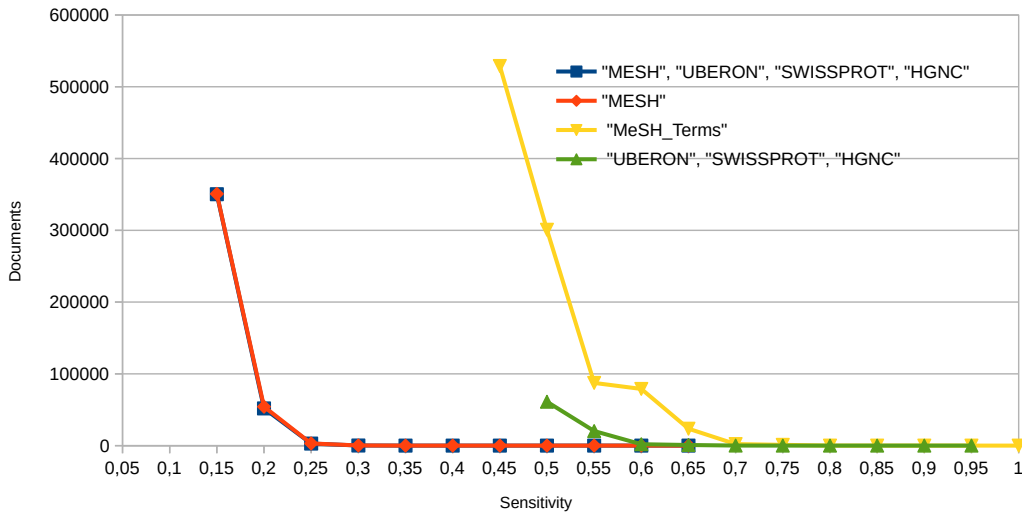


Fig. 5: Example outputs of heuristic for Corpus "Alzheimer Disease" with different layers. We used "MeSH_Terms" (manually annotated keywords from MeSH), MeSH (NER using terms in MeSH), SWISSPROT, HGNC and UBERON (NER). As we can see, the precision varies and depends on which layers are used. The text mining based MeSH has a great impact on the results, whereas the manually annotated expert knowledge from "MeSH_Terms" lead to a totally different result. For knowledge discovery, it is very important to choose the right value for s and to choose the correct layers.

changing algorithm 2 we can compute the distance between any two reverse embeddings or descriptions, see algorithm 3.

Algorithm 3 s -GRAPH-DISTANCE

Require: $d_1, d_2 \subset L$ and descriptive elements $con(d_i) = \{c_1, \dots, c_m\} \subset V$, maxiter as maximum of iterations, s as sensitivity

Ensure: A semantic graph distance $sim(d_1, d_2)$ of d_1, d_2 with elements in $V \cap L$.

```

1:  $con1 = con(d_1)$ 
2: while iteration < maxiter AND  $con1 > (s \cdot con(d_2))$  do
   remove  $c \in con1$  with maximum weight
3: end while
4:  $con1 = con1$ 
5:  $con2 = con(d_2)$ 
6: while iteration < maxiter AND  $con2 > (s \cdot con(d_1))$  do
   remove  $c \in con2$  with maximum weight
7: end while
8:  $con2 = con2$ 
9: return  $\frac{|con1 \cap con2|}{|con1 \cup con2|}$ 

```

In line 11 we compute the Jaccard similarity but any other distance measure is also possible. This describes two benefits of the knowledge graph approach: First, data clustering is a generalization of document clustering. Second, the similarity measures can be computed by using any other data layers and can be setup to fit the applications needs.

D. Knowledge Discovery on custom Layers

Combining both algorithm 1 and a custom layer in the knowledge graph we can use this for quite general knowledge

discovery. Given a knowledge graph $G = (V, E)$ with layers L_1, \dots, L_n . Let N be a set of nodes which form a subgraph $N \subset G$ of the knowledge graph G . These nodes can be seen as input data. If we generate a new custom layer L' which consists of data from different layers we can use algorithm 1 to embed the input data in the new layer.

We can generate several examples from NLP and text mining for this. For example, we can use this for text classification. If N contains only textual data (e.g. scientific literature from DBLP or PubMed) we can use several subsets of connected data to obtain the classes of any document. For text recognition we may also use subsets of layers which are not directly connected to documents. Given figure 1 we may use H2020 programmes or affiliations to recognize or classify whether a text belongs to a class or not.

V. EXPERIMENTAL RESULTS

The validity and correctness of the proposed algorithm in general was shown in [3]. Here, we will present some experimental results to show the correctness of the proposed algorithms on a multi-layer knowledge graph comprising multiple terminologies and the results of one specific knowledge discovery on custom layers within the context of dementia research.

A. Search Query Finding and Knowledge Discovery

Here, we will describe some results using algorithm 2. By design, the heuristic returns the original set of documents and a set of novel documents. Thus, the precision starting with a large value of sensitivity is in general 1.

The testing was done on a set of small literature corpora collected by scientists. Here, we present results using a corpus

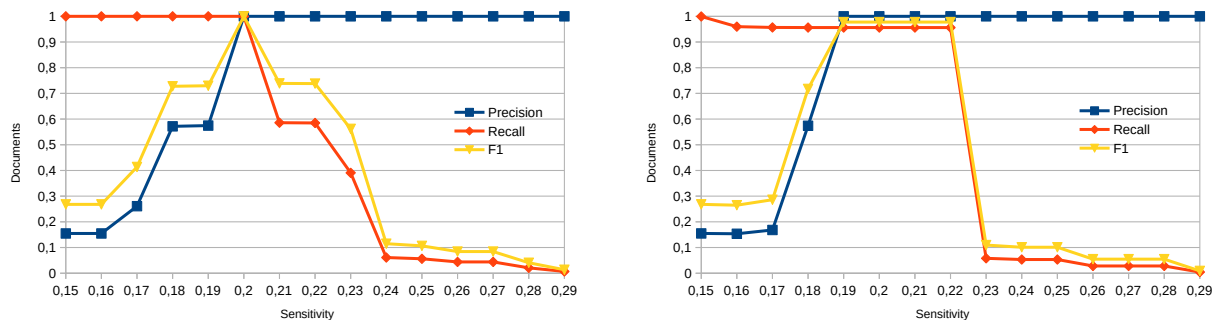


Fig. 6: Curves describing both precision and recall as well as the F_1 score for the "Alzheimer Disease" corpus and a gold standard containing 54251 documents. The results were computed using "MESH" and "HGNC" layers (left) and "MESH", "UBERON", "SWISSPROT" and "HGNC" layers (right) in the knowledge graph. It is obvious, that the gold standard was generated using MeSH-Terms. Changing layers has a great impact on the results.

of documents dedicated to alzheimers disease. First of all, we tested the algorithms with a layer of manually annotated keywords, the so called MeSH terms obtained from PubMed. We repeated the testing with several sensitivity values, see figure 5. Starting with the initial 8 documents, the amount of documents increases to 52 when using a sensitivity of 0.95 and rapidly increases to 1078 documents at 0.75. Using MeSH as a terminology used by named entity recognition the number of documents increases to 15 when the sensitivity is less than 0.45. Using all terminologies ("MESH", "UBERON", "SWISSPROT", "HGNC") the result only changes by a few documents, whereas the only usage of "UBERON", "SWISSPROT", "HGNC" changes the picture very much. We can see that different layers in the knowledge graph give a different view on the document layer and return different results.

To analyse the results, we used a manually generated gold standard for Alzheimers disease containing 54251 documents which was generated using the MeSH-Terms. We computed results using "MESH" and "HGNC" layers and "MESH", "UBERON", "SWISSPROT" and "HGNC" layers, see figure 6. We computed both precision, recall and F_1 score which is the harmonic mean of both precision and recall. With true positives (TP) in the gold standard, false positives (FP), false negatives (FN) and true negatives (TN) the precision is given by $p = |TP|/(|TP| + |FP|)$ and recall by $r = |TP|/(|TP| + |FN|)$. With this we can compute F_1 -score as $F_1 = \frac{2pr}{p+r}$.

The results in figure 6 show that the quality of results are related to the layers used and whether they were used to manually generate a gold standard. They indicate that the returned documents and their relevance relies on both the used knowledge graph layers as well as the sensitivity used. Thus, the evaluation of the proposed methods needs to consider the use case. Do we need to retrieve just a few more documents closely related to a set of documents or do we want to find all documents within a corpus. Together with the results in figure 5 we would need to discuss how the best value for sensitivity can be found.

B. Knowledge Discovery on custom Layers

We have tested the custom layer approach on a biomedical use case in the field of neurodegeneration. Alzheimer's disease (AD), also referred to simply as Alzheimer's, is a chronic neurodegenerative disease that usually starts slowly and gradually worsens over time. It is the cause of 60–70% of cases of dementia. The cause of Alzheimer's disease is poorly understood. There are no medications or supplements that have been shown to decrease risk of acquiring AD and there are no treatments stop or reverse AD progression. The human brain pharmacome project focuses on the design and construction of a dedicated knowledge base for human brain pharmacology. We used the approach discussed in this paper to create this pharmacology knowledge base, referred to as the Human Brain Pharmacome (HBP) as a unique and comprehensive resource that aggregates data and knowledge around current drug treatments that are available for major brain and neurodegenerative disorders. The HBP knowledge base provides data at a single place for building models and supporting hypotheses. Because knowledge-driven approaches to model the relevant biology and chemistry are inherently limited by the completeness and correctness of their associated knowledge assemblies, natural language processing and relation extraction are used to continuously extract biomedical relations from the recent biomedical literature and prioritize for semi-automated curation and update. One application for the HBP is Drug repositioning (also called drug repurposing). It involves the investigation of existing drugs for new therapeutic purposes. One of the main advantages of drug repositioning lies in the reduced number of required clinical trial steps and this could potentially reduce the time and costs for the medicine to reach market

We used our knowledge graph to search for interesting targets, how these targets are linked to AD and what drugs are known to interact with these targets. As can be seen in figure 8, AD can be linked to the gene CD33 which is altered in some patients suffering from the disease. The gene is coding for a protein also named CD33 which is involved

in several biological processes. Microglial activation is one of these processes that can be linked to phagocytosis. In a multicellular organism's immune system, phagocytosis is a major mechanism used to remove pathogens and cell debris. The ingested material is then digested in the phagosome. Phagocytosis is one of the main mechanisms of the innate immune defense. It is one of the first processes responding to infection, and is also one of the initiating branches of an adaptive immune response.

We have integrated H2020 data from EU Open Data Portal which contains several data fields. Persons, affiliations and documents can also be found in DBLP or PubMed data. Thus we get an linked data knowledge graphs combining H2020 data with text mining on documents from other sources.

Carefully considering the H2020 data we found for all projects, their meta data, research institutes, researchers and publications. Not all publications and persons are described. For example only 6 researchers are affiliated with Fraunhofer in this data set. Thus using H2020 as provenance, we get a fare more sparse dataset for Fraunhofer, whilst DBLP or PubMed lists all past and present affiliations in the context of publications. In addition, not all documents are listed. Querying PubMed with project acronyms usually returns more results.

In our knowledge graph the H2020 funded project PHAGO is linked to the topic of phagocytosis. In figure 9 we present a subset of the PHAGO project graph as seen by H2020. Within this project several papers to the role of CD33 and TREM2 in the process of phagocytosis and its context to AD have been published. We can directly identify experts working in the field and the organizations they are working in by switching the context. We can make several observations. First of all, the authors involved in the publications do not intersect with the researchers which are affiliated with the institutes. This is due to the fact that usually only a few researchers are mentioned in projects, thus the researchers illustrated are found in a different project scope. Thus, for knowledge discovery we can use project, documents and authors. Figure 7 illustrates the different layers.

Our goal is to understand the embedding of a H2020 project called Phago in the context of scientific literature and drug databases. Phago is related to Alzheimer's disease and studies TREM2, CD33 and related pathways in this field. Thus, we are interested in overlaps between the knowledge graph embedding towards other Alzheimer's networks, for example [31], and in drug networks, for example [32]. Thus as custom target layers we use PubMed documents, BEL networks and NE coming from the Alzheimer's network, Substances from PubChem, PharmGKB¹ and DrugBank.

Applying the method proposed in section IV-D we obtain a graph containing 126 documents, all from PubMed. We receive 29 substances and descriptive elements from MeSH and MeSH-Terms. In addition, we were able to find biomedical relations from different networks containing more than 133

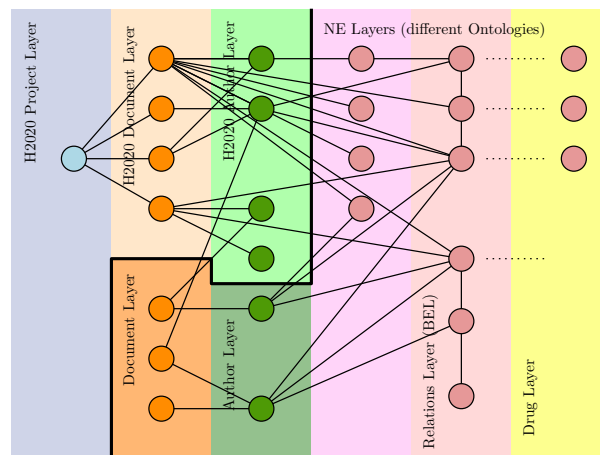


Fig. 7: An illustration of the different layers involved in exploring H2020 data. The first layer – H2020 projects – is just contained in H2020 data. Documents and Authors both contain data from H2020 and other sources. All other layers contain data from different ontologies and terminologies. They are connected using NLP and text mining technologies and also contain intra-ontology relations like biological or cause-and-effect relations.

entites from MeSH, 25 proteins and more than 66 genes, see figure 8 for a subset network illustration.



Fig. 8: Biomedical relation subnetwork linked with document PMID: 30037848 entitled "Mycobacterial PknG Targets the Rab711 Signaling Pathway To Inhibit Phagosome-Lysosome Fusion".

VI. CONCLUSION AND OUTLOOK

Big Data approaches using NLP technologies on natural language are an emerging topic in all data-driven fields. More and more extensive data is being collected, e.g. in medicine, engineering and also in the humanities (so-called "digital humanities"). To evaluate this data, new methods from the fields of artificial intelligence (AI), big data and high performance computing must be developed. For example, in medical research and digital health the massive data available build the basis for a multitude of predictive medicine Machine Learning (ML) and AI approaches. This includes also the organization of this data (knowledge management) in order to achieve reproducible research and to benchmark and evaluate these methods since both training and validation data are required.

Knowledge graphs play a central role in tackling these challenges. They address central ethical standards of science: reproducibility, transparency and a fair and – if possible –

¹See <https://www.pharmgkb.org/>.

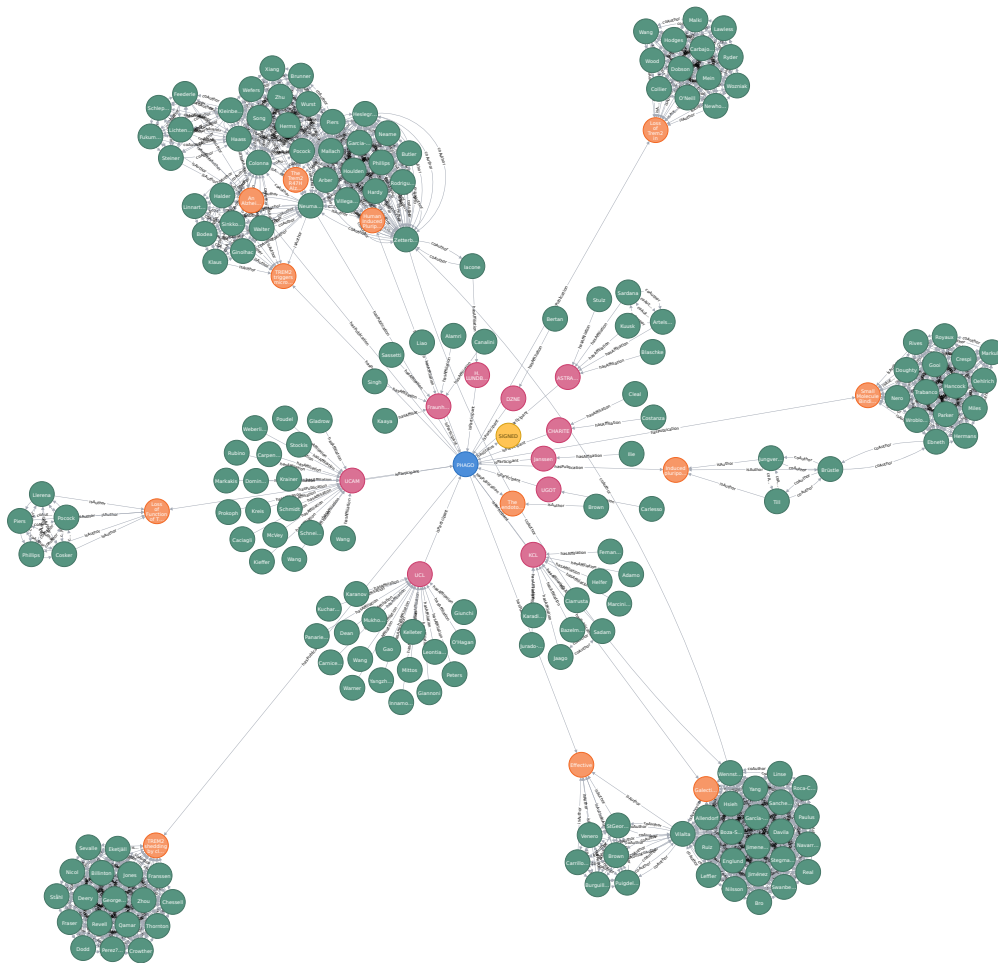


Fig. 9: A subset of the PHAGO project graph as seen by H2020. Blue nodes refer to H2020 projects, red nodes to research institutes, green nodes to persons and orange nodes to documents. Persons, affiliations and documents can be found in DBLP or PubMed data. Thus we get an linked data knowledge graphs combining H2020 data with text mining on documents from other sources.

open, handling of data. These can be summarized with the "FAIR Data" principle, which was published in 2016 by Wilkinson et al. [33]. FAIR as an acronym refers to Findable, Accessible, Interoperable and Re-usable. A central component of FAIR Data is the semantic preparation of knowledge in a format that allows not only the search and retrieval of (meta-)data, but also interoperability and reusability. This provides the central data for the application of AI methods since knowledge graph aim at comparing research data records from different sources as well as the selection of relevant data sets using graph-theoretical algorithms. Making data interoperable and accessible is necessary to develop next-generation services in NLP and text mining.

Here we presented a novel semantic approach towards a context enriched biomedical knowledge graph utilizing data (PubMed, DBLP, H2020, biomedical network) integration with linked data and text mining (NER, relation extraction) which is based on a recent approach that annotates research data with

context information. The result is a knowledge graph representation of data, the context graph. It contains computable statement representation (e.g. RDF or BEL). This graph allows to compare research data records from different sources as well as the selection of relevant data sets using graph-theoretical algorithms. It can be used as a reference system for question-answering-processes and it can be a dedicated tool that assists and guides knowledge discovery.

We showed, that this graph concept can be used for graph embedding applied in the described different approaches, e.g. with focus on topic detection and knowledge discovery. We discussed several algorithmic approaches to tackle these challenges and show results for three applications: search query finding, generating cluster labels and knowledge discovery. The presented remarkable approaches lead to valuable results on large knowledge graphs. We faced several issues with data integration and missing data, for example because the input data had a bad quality. In addition we have not yet worked on

the problem of author and affiliation disambiguation.

We compared the results of different knowledge graph layers on a text corpus. We could show that the graph embeddings itself is only valuable for different use cases when choosing the right layers and sensitivity. Although we have proven that this approach is valid, we might need to evaluate more methods to compute or estimate values for s and the knowledge graph layers. This has thrown up many questions in need of further investigation.

VII. ACKNOWLEDGMENTS

We thank Martin Hofmann-Apitius and Vanessa Lage-Rupprecht for valuable suggestions, Bruce Schultz for his technical help and Vanessa Lage-Rupprecht for carefully revising the manuscript.

This manuscript has been supported by Fraunhofer Society under the MAVO Project; Human Brain Pharmacome.

REFERENCES

- [1] J. Dörpinghaus and M. Jacobs, "Semantic knowledge graph embeddings for biomedical research: Data integration using linked open data," *Posters and Demo Track of the 15th International Conference on Semantic Systems. (Poster and Demo Track at SEMANTICS 2019)*, no. 2451, pp. 46–50, 2019. [Online]. Available: <http://ceur-ws.org/Vol-2451/#paper-10>
- [2] J. Dörpinghaus, J. Darms, and M. Jacobs, "What was the question? a systematization of information retrieval and nlp problems." in *2018 Federated Conference on Computer Science and Information Systems (FedCSIS)*. IEEE, 2018.
- [3] J. Dörpinghaus, C. Düing, and V. Weil, "A minimum set-cover problem with several constraints," in *2019 Federated Conference on Computer Science and Information Systems (FedCSIS)*, Sep. 2019, pp. 115–122.
- [4] J. Dörpinghaus, A. Stefan, B. Schultz, and M. Jacobs, "Towards context in large scale biomedical knowledge graphs," *arXiv preprint arXiv:2001.08392*, 2020.
- [5] V. Gligorijević and N. Pržulj, "Methods for biological data integration: perspectives and challenges," *Journal of the Royal Society Interface*, vol. 12, no. 112, p. 20150571, 2015.
- [6] J. Dörpinghaus and A. Stefan, "Knowledge extraction and applications utilizing context data in knowledge graphs," in *2019 Federated Conference on Computer Science and Information Systems (FedCSIS)*. IEEE, 2019, pp. 265–272.
- [7] J. Dörpinghaus, A. Stefan, B. Schultz, and M. Jacobs. (2020) Towards context in large scale biomedical knowledge graphs. [Online]. Available: <http://arxiv.org/abs/2001.08392>
- [8] C. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge University Press, 2008.
- [9] A. Clark, C. Fox, and S. Lappin, *The handbook of computational linguistics and natural language processing*. John Wiley & Sons, 2013.
- [10] H. Mirisae, E. Gaussier, C. Lagnier, and A. Guerraz, "Terminology-based text embedding for computing document similarities on technical content," *arXiv preprint arXiv:1906.01874*, 2019.
- [11] N. Yarushkina, A. Filippov, and M. Grigorieva, "Using of linguistic analysis of search query for improving the quality of information retrieval," in *International Conference on Information Technologies*. Springer, 2019, pp. 215–226.
- [12] C. S. Burns, R. M. Shapiro, T. Nix, J. T. Huber *et al.*, "Examining medline search query reproducibility and resulting variation in search results," *iConference 2019 Proceedings*, 2019.
- [13] J. Lin and W. J. Wilbur, "Pubmed related articles: a probabilistic topic-based model for content similarity," *BMC bioinformatics*, vol. 8, no. 1, p. 423, 2007.
- [14] D. Newman, S. Karimi, and L. Cavedon, "Using topic models to interpret medline's medical subject headings," in *Australasian Joint Conference on Artificial Intelligence*. Springer, 2009, pp. 270–279.
- [15] D. Trieschnigg, P. Pezik, V. Lee, F. De Jong, W. Kraaij, and D. Rebholz-Schuhmann, "Mesh up: effective mesh text classification for improved document retrieval," *Bioinformatics*, vol. 25, no. 11, pp. 1412–1418, 2009.
- [16] Z. Lu, W. J. Wilbur, J. R. McEntyre, A. Iskhakov, and L. Szilagy, "Finding query suggestions for pubmed," in *AMIA Annual Symposium Proceedings*, vol. 2009. American Medical Informatics Association, 2009, p. 396.
- [17] M. Hagen, M. Michel, and B. Stein, "What was the query? generating queries for document sets with applications in cluster labeling," in *International Conference on Applications of Natural Language to Information Systems*. Springer, 2015, pp. 124–133.
- [18] Y. Yan, X.-C. Yin, C. Yang, S. Li, and B.-W. Zhang, "Biomedical literature classification with a cnns-based hybrid learning network," *PLoS one*, vol. 13, no. 7, p. e0197933, 2018.
- [19] A. Varghese, M. Cawley, and T. Hong, "Supervised clustering for automated document classification and prioritization: a case study using toxicological abstracts," *Environment Systems and Decisions*, vol. 38, no. 3, pp. 398–414, 2018.
- [20] D. Fensel, U. Şimşek, K. Angele, E. Huaman, E. Kärle, O. Panasiuk, I. Toma, J. Umbrich, and A. Wahler, *Introduction: What Is a Knowledge Graph?* Cham: Springer International Publishing, 2020, pp. 1–10. [Online]. Available: https://doi.org/10.1007/978-3-030-37439-6_1
- [21] L. Ehlringer and W. Wöb, "Towards a definition of knowledge graphs," *SEMANTICS (Posters, Demos, SuCESS)*, vol. 48, 2016.
- [22] M. Ley, "Dblp: some lessons learned," *Proceedings of the VLDB Endowment*, vol. 2, no. 2, pp. 1493–1500, 2009.
- [23] A. A. Salatino, F. Osborne, T. Thanapalasingam, and E. Motta, "The cso classifier: Ontology-driven detection of research topics in scholarly articles," in *International Conference on Theory and Practice of Digital Libraries*. Springer, 2019, pp. 296–311.
- [24] B. Yates, B. Braschi, K. A. Gray, R. L. Seal, S. Tweedie, and E. A. Bruford, "Genenames.org: the HGNC and VGNC resources in 2017," *Nucleic Acids Research*, vol. 45, no. D1, pp. D619–D625, 10 2016. [Online]. Available: <https://doi.org/10.1093/nar/gkw1033>
- [25] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig *et al.*, "Gene ontology: tool for the unification of biology," *Nature genetics*, vol. 25, no. 1, pp. 25–29, 2000.
- [26] G. O. Consortium, "The gene ontology resource: 20 years and still going strong," *Nucleic acids research*, vol. 47, no. D1, pp. D330–D338, 2019.
- [27] L. M. Schriml, E. Mittra, J. Munro, B. Tauber, M. Schor, L. Nickle, V. Felix, L. Jeng, C. Bearer, R. Lichenstein *et al.*, "Human disease ontology 2018 update: classification, content and workflow expansion," *Nucleic acids research*, vol. 47, no. D1, pp. D955–D962, 2019.
- [28] R. Feldman and J. Sanger, *The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data*. Cambridge University Press, 2007.
- [29] F. França and A. de Souza, *Intelligent Text Categorization and Clustering*, ser. Studies in Computational Intelligence. Springer Berlin Heidelberg, 2008.
- [30] J. Dörpinghaus, S. Schaaf, and M. Jacobs, "Soft document clustering using a novel graph covering approach," *BioData mining*, vol. 11, no. 1, p. 11, 2018.
- [31] A. T. Kodamullil, E. Younesi, M. Naz, S. Bagewadi, and M. Hofmann-Apitius, "Computable cause-and-effect models of healthy and alzheimer's disease states and their mechanistic differential analysis," *Alzheimer's & Dementia*, vol. 11, no. 11, pp. 1329–1339, 2015.
- [32] D. S. Wishart, Y. D. Feunang, A. S. Guo, E. J. Lo, A. Marcu, J. R. Grant, T. Sajed, D. Johnson, C. Li, Z. Sayeeda *et al.*, "Drugbank 5.0: a major update to the drugbank database for 2018," *Nucleic acids research*, vol. 46, no. D1, pp. D1074–D1082, 2017.
- [33] M. D. Wilkinson, M. Dumontier, I. J. Aalbersberg, G. Appleton, M. Axton, A. Baak, N. Blomberg, J.-W. Boiten, L. B. da Silva Santos, P. E. Bourne *et al.*, "The fair guiding principles for scientific data management and stewardship," *Scientific data*, vol. 3, 2016.