Intelligent Systems with Applications 17 (2023) 200161

Contents lists available at ScienceDirect

Intelligent Systems with Applications

journal homepage: www.journals.elsevier.com/intelligent-systems-with-applications

Improving graph embeddings via entity linking: A case study on Italian clinical notes

Daniela D'Auria [a], Vincenzo Moscato [b], Marco Postiglione [b], Giuseppe Romito [b], Giancarlo Sperlí [b,*]

[a] Faculty of Computer Science, Free University of Bozen-Bolzano, Bozen-Bolzano, 39100, Italy
[b] University of Naples "Federico II", Dept. of Electrical Engineering and Information Technology (DIETI), Via Claudio 21, 80125, Italy

ARTICLE INFO

Keywords:
Entity linking
Graph embedding
Link prediction
Health analytics
Healthcare

ABSTRACT

The ever-increasing availability of Electronic Health Records (EHRs) is the key enabling factor of *precision medicine*, which aims to provide therapies and diagnoses based not only on medical literature, but also on clinical experience and individual information of patients (e.g. genomics, lifestyle, health history). The unstructured nature of EHRs has posed several challenges on their effective analysis, and heterogeneous graphs are the most suitable solution to handle the heterogeneity of information contained in EHRs. However, while EHRs are an extremely valuable data source, information from current medical literature has yet to be considered in clinical decision support systems. In this work, we build an heterogeneous graph from Italian EHRs provided by the Hospital of Naples Federico II, and we define a methodological workflow allowing us to predict the presence of a link between patients and diagnosed diseases. We empirically demonstrate that linking concepts to biomedical ontologies (e.g. UMLS, DBpedia) — which allow us to extract entities and relationships from medical literature — is significantly beneficial to our link-prediction workflow in terms of Area Under the ROC curve (AUC) and Mean Reciprocal Rank (MRR).

1. Introduction

The top 10 drugs in the United States, under the best of circumstances, are effective on 1 patient out of 4 (1 out of 25 in the worst-case scenario) (Schork, 2015). We could report a plethora of similar examples to highlight the need for innovative ways to provide diagnoses and therapies so that they can be tailored according to each individual patient, with his lifestyle, genomics, comorbidities, and clinical history (Abul-Husn & Kenny, 2019, Kraljevic et al., 2021). This is the main objective of *precision medicine*, which has been enabled by the ever-increasing availability of digitized medical documents, often in the form of Electronic Health Records (EHRs) (Abul-Husn & Kenny, 2019, Rajkomar et al., 2018). These documents are filled out by nurses and physicians in hospital wards and contain details about hospital admissions (e.g. anamneses, diagnoses, treatments, ICU admissions) with an unstructured or semi-structured approach (Kormilitzin et al., 2021, Negro-Calduch et al., 2021, Gao et al., 2021).

The lack of a unified approach/framework to compile and handle clinical notes, the shortage of medical staff time — which is usually implied in much more important activities — inevitably result in two major problems (Yoon et al., 2019): (1) the heterogeneity of data coming from different sources (i.e. hospitals, wards or even health professionals) and (2) the abundance of abbreviations and ortographic errors which, along with the peculiarities of biomedical text characterized by polysemous words and alternate spellings, make it extremely difficult to automatically retrieve information from such documents.

Nowadays, heterogeneous graphs are the most suitable solution to the former problem (Liu et al., 2020): information from heterogeneous sources can be integrated in a unified data structure consisting of medical entities (e.g. hospital admissions, symptoms, diseases, drugs) and their relations represented as nodes and edges, respectively. Not only do these data structures enable the possibility to easily and intuitively explore EHRs, but they can be also used as the input of many downstream tasks, such as community detection (Moscato & Sperlì, 2022), drug repurposing (Zhang et al., 2021), question answering (Park et al.,

* Corresponding author.
E-mail addresses: daniela.dauria@unibz.it (D. D'Auria), vincenzo.moscato@unina.it (V. Moscato), marco.postiglione@unina.it (M. Postiglione), gius.romito@studenti.unina.it (G. Romito), giancarlo.sperli@unina.it (G. Sperlí).

https://doi.org/10.1016/j.iswa.2022.200161
Received 1 July 2022; Received in revised form 15 October 2022; Accepted 27 November 2022
Available online 1 December 2022

2021), and so on. For example, hospital admissions — with information about the anamnesis, diagnosis and tested symptoms — can be stored in an integrated heterogeneous graph which allows physicians to explore the medical history of patients and AI systems to analyze the hidden value in all the entities and relationships.

The difficulty in processing the unstructured information of EHRs, and thus in retrieving entities and relation from such data, is the main hurdle for an effective use of heterogeneous graphs. Xiao et al. (2019) have shown the importance of integrating knowledge graph data with ontologies not only to easily leverage data from heterogeneous data sources, but also to enrich data with domain knowledge. In light of this, to improve the quality heterogeneous graphs, information from EHRs can be enriched with external knowledge from existing medical ontologies, such as UMLS (Bodenreider, 2004) and DBpedia (Auer et al., 2007a), which allow to add new nodes and new relationships to the existing graph based on the existing medical literature.

In this work, we build a knowledge graph of hospital admissions in cardiology departments with data provided by the *Hospital of Naples Federico II*. We define a methodological workflow which — by leveraging state-of-the-art techniques for entity linking and heterogeneous graph embeddings — allows us to predict links between the patient and possible diagnosed diseases based on its medical history and relations with other patients histories. Above all, we empirically demonstrate the beneficial effects of entity linking which, allowing us to semantically enrich the information embedded in the heterogeneous graph, guarantees a significant improvement in link prediction performance. Relying just on the medical history of the patient and its current symptoms, our framework could not only be successfully applied by physicians to explore possible diagnoses and complications, but also to reduce the load of First Aids, which are often swamped by people who do not actually need first aid — predictions of links to disorders may help patients understand whether they are in an emergency state or not. In this perspective, to ease the burden of physicians, our approach could be also integrated with tools automatically analyzing electrocardiograms (Persia et al., 2021) by integrating their information within the graph data structure.

The remainder of this work is structured as follows: in Section 2 we provide the theoretical background of this work and concurrently examine related work; in Section 3 we describe the Italian corpus provided by the Hospital of Naples Federico II, while in Section 4 we provide an in-depth description of the methodological flow characterizing this work. Finally, experimental results are shown in Section 5.

2. Background and related work

In this section we examine related work and provide a theoretical background of this paper. In particular, in Section 2.1, an overview of heterogeneous graphs, with definitions (Section 2.1.1) and an overview of *representation learning* (Section 2.1.2 and *link prediction* downstream task (Section 2.1.3) is given. An in-depth examination of Entity Linking is reported in Section 2.2. In Section 2.3 an overview of current biomedical ontologies is provided, with a focus on those used in this paper. Finally in Section 2.4 we discuss about our contribution, highlighting the novelties of the proposed approach.

2.1. Heterogeneous graphs

In general, a graph $\mathcal{G}$ is defined as a data structure consisting in a set of nodes $v \in \mathcal{V}$ linked between them with edges $e \in \mathcal{E}$. However, biomedical scenarios are plenty of inter-related entities with different types (e.g. patients, diseases, treatments, symptoms). Thus, characterizing nodes with their *type* may be extremely informative and useful for downstream tasks. Graphs with two or more types of nodes and/or relationships are called *heterogeneous graphs*.

Heterogeneous graphs were born with the purpose of incorporating human knowledge into intelligent systems. This knowledge is represented by a graph-based structure whose nodes represent real-world entities, while edges define different relationships between those entities. In particular, in the biomedical field, the biomedical heterogeneous graphs play a central role in big data integration. Bringing unstructured text into a structured and comparable format is one of the key assets. As cause and effect models, heterogeneous graphs can potentially analyze interactions and relations between patients, diseases, genes, drugs, proteins and more, or divide patients into communities, or facilitate clinical decision making or help to drive research towards precision medicine. The use of graphs in the biomedical domain has been widely explored: Ma et al. (2018) and Choi et al. (2017) use external data sources and Graph Neural Networks to learn embeddings from them, which are then used for downstream tasks (e.g. sequential diagnoses prediction, heart failures prediction). Choi et al. (2020) and Choi et al. (2018) assume that different kinds of medical codes in several EHRs have latent causal relations which can be exploited to perform downstream predictions. Despite their proven effectiveness, these methods focus on homogeneous graphs without taking into account the heterogeneity of the information embedded in EHRs. To fill this gap, Liu et al. (2020) propose a Similarity Graph Neural Network (HSGNN) to organize information of EHRs in multiple homogeneous graphs, which are then combined into one graph to make diagnosis predictions. In this work, we analyze the effects of exploiting external ontologies to insert structural information (e.g. known relations between diseases and treatments) into an existing heterogeneous graph which models EHRs data.

2.1.1. Definitions

Definition 1 *(Heterogeneous graph)*. Formally, a heterogeneous graph $H = \{V, E, X, R, \phi, \psi\}$ is a network with multiple types of nodes and links. Particularly, within H, each node v_i is associated with a node type $\phi(v_i)$, and each link $e_{ij} \in E$ is associated with a link type $\psi(e_{ij})$.

Definition 2 *(Meta-path)*. A meta-path is a path defined on the network schema denoted in the form of $o_1 \xrightarrow{l_1} o_2 \xrightarrow{l_2} \ldots \xrightarrow{l_m} o_{m+1}$ where o and l are node types and link types, respectively.

Each meta-path captures the proximity among the nodes on its two ends from a particular semantic perspective.

Definition 3 *(Heterogeneous graph embedding)*. For a given heterogeneous graph H, a heterogeneous network embedding is a set of mapping functions $\{\phi_k : V_k \mapsto \mathbb{R}^{|V_k| \times d}\}_{k=1}^{K}$ where K is the number of node types, $\forall v_i \in V_k, \phi(v_i) = k, d \ll |V|$. Each mapping ϕ_k defines the latent representation (a.k.a. embedding) of all nodes of type k, which captures the network topological information regarding the heterogeneous links in E.

2.1.2. Representation learning

Representation Learning is the process that maps structured and unstructured data to *embedding vectors*, so as they can be directly executable by various downstream machine learning algorithms.

It is possible to categorize the different approaches of HNE existing in the literature into 3 different groups, based on their common objectives (Yang et al., 2020):

- **Proximity-Preserving Methods** (Fu et al., 2017, Zhang et al., 2022). By preserving different types of proximity among nodes it is possible to capture the topological information of the network. There are two major categories of proximity-preserving methods in HNE: random walk approaches and first/second-order proximity based ones. Both types of proximity-preserving methods are considered as shallow network embedding, due to their essential single-layer decomposition of certain affinity matrices.

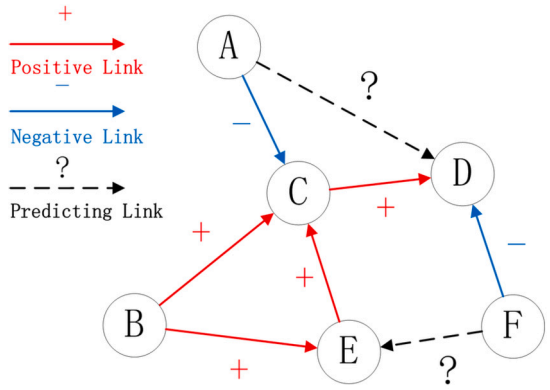

Fig. 1. Link prediction example.

- **Message-Passing Methods** (Schlichtkrull et al., 2018, Hu et al., 2020). Each node in a network can have an attribute information represented as a feature vector x_u. Message-passing methods aim to learn node embeddings e_u based on x_u, by aggregating information from u's neighbors. Graph Neural Networks (GNNs) are widely used for the message-passing process. Therefore, unlike proximity based HNE methods, message-passing methods are often considered as deep network embedding, due to their multiple layers of learnable projection functions.
- **Relation-Learning Methods** (Yang et al., 2015, Dettmers et al., 2018). Unlike previous methods that modeled the relation among edge types via parametric algebraic operations, relation-learning methods rely on introducing a triple-based scoring function instead of considering meta-paths or meta-graphs. Each edge in a knowledge graph can be seen as a triple (u,l,v) where u,v are two nodes of the graph defining an edge, whose type is l. The goal of these methods is to learn a scoring function $s_l(u,v)$ that evaluates each triple and returns a scalar to measure the acceptability of each.

2.1.3. Link prediction

Link prediction is the task of predicting the existence of a link between two nodes in a network. There are several examples of link prediction like: predicting friendship links among users in a social network, predicting co-authorship links in a citation network, predicting interactions between genes and proteins in a biological network, and, as in our case, predict if there is a link between a patient and a disease in a biomedical network. Link prediction can also have a inductive aspect, where, given a snapshot of the set of links at time t, the goal is to predict the links at time $t+1$ (see Fig. 1).

Problem definition Given a network $G=(V,E)$, where V represents the entity nodes in the network and E represents the set of true links across entities in the network, we consider the set of entities V and a subset of true links known as *observed links* (training set), with the goal of predicting true links not considered in observed links, known as *unobserved links*. In the inductive formulation of link prediction, the observed links correspond to true links at a time t, and the goal is to infer the set of true links at time t+1. Usually, we are also given a subset of unobserved links called potential links E' (test set), and we need to identify true links among these potential links.

In the binary classification formulation of the link prediction task, the potential links are classified as either true links or false links. Link prediction approaches for this setting learn a classifier that maps links in E' to positive and negative labels. In the probability estimation formulation, potential links are associated with existence probabilities. Link prediction approaches for this setting learn a model that maps links in E' to a probability.

Approaches Based on the type of information used to predict links, approaches can be categorized as:

- *Topology-based methods* (Behrouzi et al., 2020). Nodes with similar network structure are more likely to form a link. In this regard, different methods can be used to calculate the similarity between two nodes based on their common neighbors, as: Common neighbors, Jaccard measure, Adamic–Adar measure, Katz measure.
- *Content-based approaches* (Gao et al., 2011). These approaches predict the existence of a link based on the similarity of the node attributes, calculated with cosine similarity on euclidean distance.
- *Mixed methods* (Zhang et al., 2020). They combine attribute and topology based methods. One of the most used mixed methods is *graph embeddings*, which learn an embedding space in which neighboring nodes are represented by vectors so that vector similarity measures, such as dot product similarity, or euclidean distance, hold in the embedding space. These similarities are functions of both topological features and attribute-based similarity. One can then use other machine learning techniques (e.g., SVM) to predict edges on the basis of vector similarity.

2.2. Entity linking

Entity Linking (EL) consists in mapping named entity mentions to their corresponding concepts in a knowledge base. Named entity mentions are token sequences in text which refer to an entity type of interest (e.g. person, disease, location). The general architecture to solve this task consists of 3 different steps (Shen et al., 2015):

- *Generation of candidate entities.* (Shen et al., 2013, Deorowicz & Ciura, 2005, Zhang et al., 2010) For each named entity mention m, the EL system filters out irrelevant entities in the knowledge base and returns a subset of candidate entities which m may refer to.
- *Candidate ranking.* (Cucerzan, 2007, Chen et al., 2010, Shen et al., 2012, Chen & Ji, 2011) Candidate entities are ranked based on supervised or unsupervised methods to retrieve the most appropriate candidates to be linked to the named entity mention.
- *Unlinkable mentions prediction.* In some cases, named entity mentions cannot be linked with any of the concepts in the knowledge base.

In the last few years, several approaches have been proposed to improve the overall performance by jointly performing both the named entity recognition and disambiguation tasks (*end-to-end* entity linking) (Broscheit, 2019, Wiatrak & Iso-Sipila, 2020).

Another effective approach to entity linking consists in leveraging domain knowledge: for example, in the biomedical field, knowledge bases and thesauri with biomedical concepts which may be linked to named entities exist (see Section 2.3). Bhowmik et al. (2021) follow the classic *"retrieve and rerank"* paradigm to solve the entity linking task using an architecture based on Bi-Encoder that uses the information on the context of the mention to link it to the most similar candidate entity in the considered knowledge base. Zhu et al. (2020) and Onoe and Durrett (2020) add further information to improve the performance obtained, that is, the prediction of the type associated with each candidate entity and each mention, based on the idea that knowing the types of entities simplifies disambiguation.

Since it is not always possible to rely on rich information about a specific domain (i.e. annotated data, large knowledge bases), domain-independent approaches based on distant learning for the annotation of unlabeled data (Le & Titov, 2019) and zero-shot approaches to train the model on one domain and predict on other different domains (Wu et al., 2020, Yao et al., 2020) have become widespread.

Finally, in recent years, an attempt has been made to improve the vector representations obtained for each entity in the knowledge base considered by exploiting, not only information based on unstructured text, but also information based on graph embeddings, exploiting the relationships between the different entities in the basis of knowledge in

Table 1
Comparison with related work.

Ref.	Year	Nodes	Task	Method	Ontology
Yang and Yang (2015)	2015	user, drug, disease, adverse drug reaction (ADR)	drug-drug interaction	logistic regression	×
Chen et al. (2016)	2016	record, physical test, mental assessment, profile	risk prediction	optimization	×
Yang and Yang (2016)	2016	user, drug, disease, ADR	drug-drug interaction	decision tree, k-NN, MLP, SVM, RBF	×
Zhao and Yang (2016)	2016	drug, disease, ADR	drug repositioning	path mining	×
Pham et al. (2018)	2018	kidney conditions, hepatitis, diabetes, blood pressure and cholesterol, heart disease, respiratory disease, profile and others	risk prediction	optimization	×
Sun et al. (2019)	2019	patient, diagnosis	disease progression mining	community detection, rule mining	×
Wanyan et al. (2020)	2020	patient, lab, diagnosis	disease prediction	heterogeneous graph embeddings	×
Liu et al. (2020)	2020	patient, visit, diagnosis, medication	disease prediction	heterogeneous graph embeddings	×
Wang et al. (2021)	2021	patient, symptom, disease	disease prediction	heterogeneous graph embeddings	×
Wanyan et al. (2021)	2021	patient, lab, diagnosis	mortality prediction	heterogeneous graph embeddings	×
Ours	2022	patient, disease, symptom, drug	disease prediction	heterogeneous graph embeddings	✓

order to collectively disambiguate different mentions within the same document (Bhowmik et al., 2021, Parravicini et al., 2019).

2.3. Biomedical knowledge bases

To solve the entity linking problem, it is necessary to consider as input not only the mentions, but also a knowledge base that allows to link each mention to the corresponding entity in the considered knowledge base. The idea behind biomedical entity linking is that known relations between medical concepts may help in prediction downstream tasks. The vast amount of publicly available biomedical databases provide a rich resource for factual knowledge to be added in heterogeneous graphs. For example, the National Center for Biomedical Ontology (NCBO) BioPortal[1] adds about 75 new ontologies each year. In this work, we will consider two different knowledge bases, known as:

- **MeSH**.[2] The Medical Subject Headings (MeSH) thesaurus is a controlled and hierarchically-organized vocabulary produced by the National Library of Medicine. It is used for indexing, cataloging, and searching of biomedical and health-related information. MeSH includes the subject headings appearing in MEDLINE/PubMed, the NLM Catalog, and other NLM databases.
- **DBpedia** (Auer et al., 2007b). The DBpedia community project extracts structured and multilingual knowledge from Wikipedia, then makes it freely available on the Web using Semantic Web and Linked Data technologies. The project extracts knowledge from 111 different language editions of Wikipedia. The largest DBpedia knowledge base which is extracted from the English edition of Wikipedia consists of over 400 million facts that describe 3.7 million things. The DBpedia knowledge bases that are extracted from the other 110 Wikipedia editions together consist of 1.46 billion facts and describe 10 million additional things. The DBpedia project maps Wikipedia infoboxes from 27 different language editions to a single shared ontology consisting of 320 classes and 1,650 properties.

[1] https://bioportal.bioontology.org.
[2] http://www.nlm.nih.gov/mesh/.

2.4. Our contribution

In our work, after building a biomedical heterogeneous knowledge graph, we see how the entity linking task can improve the previously created knowledge graph by normalizing the entities within it, with entities present in the DBPedia knowledge base and adding from the latter further information. In this way, after calculating the node embeddings, it will be possible to obtain better performances for the task of link prediction and well-distributed patient communities through cluster analysis.

Unlike the approach used in Parravicini et al. (2019), who used a technique based on Deep Walk (Perozzi et al., 2014) to obtain vector representations of each vertex of the knowledge graph, in our work state-of-the-art heterogeneous network representation learning techniques were used (Message-passing methods, Relation-learning methods formalized in Chapter 1).

Inspired by the recent approaches used to improve the performance for the entity linking task through graph embeddings (Parravicini et al., 2019), the reverse process was carried out: an entity linking model based on Dual Encoder and Cross Encoder (Wu et al., 2020) was used to improve the quality of a heterogeneous knowledge graph, normalizing the entities inside it with the entities present in DBpedia and adding further information extracted from the normalized entities.

Table 1 summarizes existing approaches applying heterogeneous graph analysis to healthcare. We can observe that heterogeneous graphs have been used for many downstream tasks: *drug-drug interaction* prediction is a significant health safety challenge due to the wide presence of adverse drug reactions, which should be preventively detected; *drug repositioning* aims to identify known drugs which could be applied for new indications; *health risk* prediction aims to identify patients at risk based on their current and past EHRs, where the risk is intended as an unwanted outcome, such as mortality or morbidity; *disease prediction* identifies possible diagnoses for a patient based on his EHR or medical history (a.k.a. disease progression mining). In our work, we will focus on disease prediction, which will be handled as a *link prediction* task, i.e. we will identify links between *Patient* and *Disease* nodes based on their EHRs. As in recent works, we analyze the heterogeneous graph structure by means of heterogeneous graph embeddings, and compare performance obtained with different state-of-the-art methods. To the best of our knowledge, entities and relations obtained by integrating EHR-based heterogeneous graphs to existing biomedical knowledge bases have been never investigated, despite having a high-potential value

Table 2
Dataset characterization.

Statistic	Value
Number of admissions	404
Number of patients	239
Admissions date range	from 2012-02-28 to 2021-01-20
Admissions per patient (mean, std)	1.690, 1.646
Maximum number of admissions	11
Top 5 diseases with number of occurrences	(Hypertensive disorder, 291), (Atrial fibrillation, 151), (Coronary artery disorder, 121), (Dyslipoproteinemias, 110), (Diabetes mellitus type 1, 101)

Table 3
Graph characterization.

Statistic	Value
Total number of nodes	1764
Total number of edges	3351
Number of disease nodes	204
Number of anamnesis disease nodes	668
Number of diagnosis disease nodes	455
Number of symptom nodes	143
Number of patient nodes	239
Mean number of diseases for patient	4.71
Mean number of symptoms for patient	2.29

thanks to the public medical knowledge about entities they provide. To fill this gap, we build two different heterogeneous graphs (*before* and *after* the entity linking based enrichment) and study the performance of link prediction models based on heterogeneous graph embeddings. In our experiments, we confirm the added value of ontologies to EHR-based heterogeneous graphs, which allow us to obtain significant and consistent performance improvements with all the embedding methods experimented.

3. Material

In this work we used a corpus of clinical records, which we will refer to with the name *Wincare*, whose characterization has shown in Table 2, regarding hospital admissions in cardiology departments provided by *Hospital of Naples Federico II*.

A team of students in biomedical engineering annotated a total of 1000 disease mentions linked to unique concepts obtained from thesauri, such as DBpedia and Medical Subject Heading (MeSH). For the labeling necessary for the entity linking task, it was sufficient to assign an alphanumeric label corresponding to a MeshID code to each mention labeled as Disease. The corpus characteristics are the following: 700 disease mentions in the training set, 150 in the development set and 150 in the test set.

Starting from the dataset labeled for the Name Entity Recognition task, in which each token was associated with a label between: B-Disease, I-Disease, B-Symptom, I-Symptom, B-Drug, I-Drug, O. Then each token was associated with a further label representative of the specific biomedical entity, represented by the considered token. These labels were obtained through external knowledge bases such as DBpedia (Auer et al., 2007a) and MeSH (Lipscomb, 2000) (Medical Subject Headings), by extracting the corresponding Mesh ID code for each biomedical entity. In particular, through the appropriate search engines of DBpedia and MeSH, called respectively OpenLink Virtuoso and MeSH Browser, it is possible to search, for each token of the dataset provided, the corresponding entity in the knowledge base and extract from the latter the mesh ID code to be used as the token label. In this way, it was possible to label 1000 different entity mentions that could be used for the fine-tuning of entity linking models, based on transformers such as BERT. Specifically, 85.8% of these 1000 total mentions have the corresponding biomedical entity in the DBpedia knowledge base, while only the remaining 14.2% do not. For this 14.2% the label corresponding to each entity mention was extracted from the MeSH knowledge base.

For the sake of completeness, we report in Table 3 statistics of our heterogeneous graph resulting from the methodological workflow.

4. Methods

The main purpose of our study concerns the analysis of entity linking approaches on Italian clinical notes, with the aim to improve the link prediction task through graph embeddings strategy. In particular, we extract information from medical records of patients treated in Italian hospitals for building and improving biomedical knowledge graphs. Starting from the constructed knowledge graph, state-of-the-art techniques of heterogeneous graph representation learning have been used to represent each node of the graph in a low-dimensional space, thus neighboring nodes in the graph have similar vector representations (*homophily*). These representations have been then used to visually analyze the results obtained through dimensionality reduction techniques (PCA and t-SNE) and to deal with the link prediction task.

The overall methodological workflow is summarized in Fig. 2. First, we pre-process the available dataset so as to collect the information of interest from the original dataset. Clinical notes concerning the anamneses, tested symptoms and diagnoses of hospital admissions are thus retrieved and organized with a graph data structure. A NER step is performed so as to connect these unstructured notes to medical concepts, i.e. diseases, symptoms and drugs. Then, an Entity Linking step is performed not only to disambiguate entity mentions but also to retrieve new relations between medical concepts. By computing heterogeneous graph embeddings, we can eventually perform link prediction to predict the links from diagnoses to disorders, thus allowing us to help physicians in the identification of possible medical problems.

We will now deeply describe each step in the methodology and we will provide a running example alongside to facilitate understanding.

Running Example 1. Let us consider a *patient* with the following anamnesis clinical note:

> "Ex fumatore, iperteso, displidemico, nega familiarità. IMA nel 2006. PCi primaria su IVA e successiva PCI su Cx. Nega angor, riferisce dispnea da sforzi non abituali"

Starting from this unstructured text comment, our aim is to extract medical concepts and predict associated disorders.

4.1. Preprocessing

In our dataset, each patient can make one or more visits, that are associated with a single medical history (anamnesis), a single diagnosis and a single information on the symptoms tested by the doctor. In turn, each medical history can have zero or more symptoms, diseases and drugs, while each diagnosis can only have information on drugs to be taken after the visit and diseases diagnosed.

Starting from the raw data provided, it was necessary to analyze this data to extract the information we needed to build the knowledge graph. In particular, for each patient the following information was obtained:

- *Visits*: numeric identification code, date and time.
- *Anamnesis*: the collection and critical study of the symptoms and facts of medical interest reported by the patient or his family. This investigation is carried out with the aim of enriching picture of information useful for a correct diagnosis of current morbid state.
- *Diagnosis*: the identification of the disease, affection or injury, of its location and its nature. Identification is achieved through the

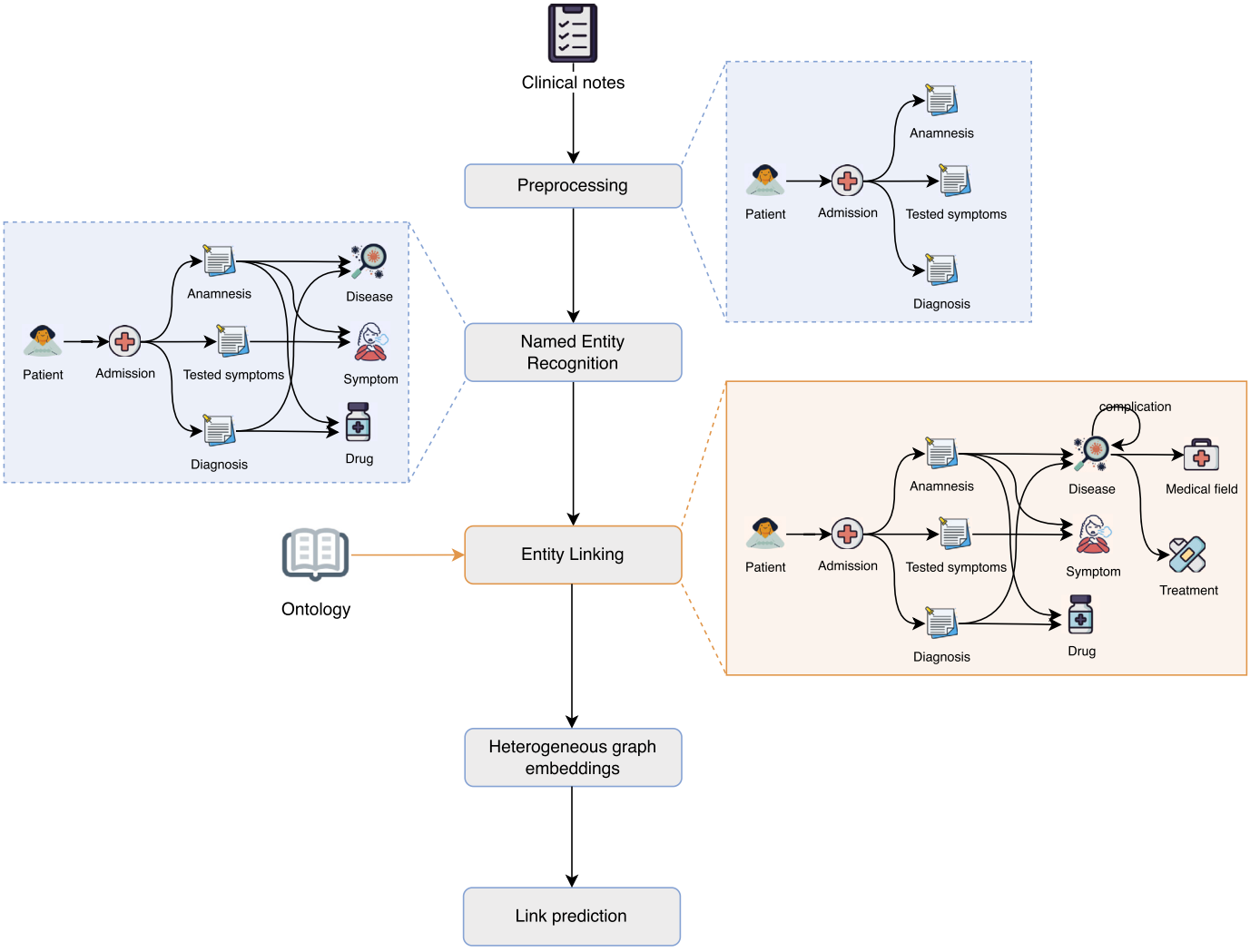

Fig. 2. Overall methodological workflow in our analysis.

evaluation of the various symptoms presented by the patient, based on analog reasoning.
- Symptoms tested by doctors.
- Symptoms and Disease in the Anamnesis phase.
- Diagnosed diseases.
- Drugs taken before and after the visit.

4.2. Named entity recognition

The information on Symptoms and Diseases are extracted from the textual fields relating to the anamnesis and diagnosis present in the medical records, provided through a previous phase of labeling according to the IOB2 (Inside – outside – beginning) format (Ramshaw & Marcus, 1999), which is a common tagging format for tagging tokens in a chunking task in computational linguistics. The I- prefix before a tag indicates that the tag is inside a chunk (in our case I-Disease or I-Symptom). An O tag indicates that a token belongs to no chunk. The B- prefix before a tag indicates that the tag is the beginning of a chunk (in our case B-Disease or I-Disease). After splitting the text into tokens, for each token contained in the text relating to the anamnesis field and the diagnosis field, a label is assigned between: B-Disease, I-Disease, B-Symptom, I-Symptom, B-Drug, I-Drug, O.

Running Example 1 *(continuing from p. 5)*. The NER step allows us to individuate and extract medical concepts from the clinical note. We show retrieved concepts from clinical note under analysis, where disease and symptom concepts are highlighted:

> "Ex fumatore, iperteso , displidemico , nega familiarità. IMA nel 2006. PCi primaria su IVA e successiva PCI su Cx. Nega angor, riferisce dispnea da sforzi non abituali"

4.3. Entity linking

In general, an entity linking model needs two different inputs: the examples to be trained, validated and tested on (dataset), and an external knowledge base to link each example to the corresponding entity in the knowledge base itself. In particular, the information on the mentions and their contexts are extracted from the dataset, while information on the entities (i.e., title and description of each entity) is inferred from the knowledge base used.

We used BLINK (Wu et al., 2020) for entity linking due to its proven effectiveness in few-shot scenarios, where we do not have the availability of big training sets, as in our case. In particular, to link each mention to the corresponding entity in the considered knowledge base we used the union of two different models:

- **Dual-Encoder model**. It uses two independent BERT transformers model to encode context/mention and entity into dense vectors, and each entity candidate is scored as the dot product of these

vectors. Candidates retrieved by the dual-encoder are then passed to cross-encoder for ranking.
- **Cross-Encoder model**. It encodes context/mention and entity in one transformer, and applies an additional linear layer to compute the final score for each pair. The input is concatenation of the input context, mention representation and entity representation. This allows the model to have deep cross attention between the context and entity descriptions.

One of the main advantages of this entity linking model is the ability to correctly predict, in the test phase, even mentions never seen in the training phase, exploiting the information on the title and description of the candidate entities.

4.3.1. Entity normalization

In this phase, the results obtained from the entity linking model are used to normalize the entities of Disease type, contained in the previously created knowledge graph. In particular, the model will predict for each mention within the dataset, the corresponding correct entity within the knowledge base. After storing this information in a file, it is possible to create a script that executes several cypher queries on the graph in order to modify the disease type nodes (corresponding to the mentions of our dataset) with new disease type nodes, corresponding to the predicted entities belonging to the DBpedia knowledge base.

4.3.2. Enriching the information stored in the knowledge graph via DBpedia

After the entity normalization step, we will have a set of Disease type nodes within the graph that corresponds to the entities stored in the DBpedia knowledge base, used to normalize the entity mentions present in the Wincare dataset. As a result, once these DBpedia entities were obtained, it was possible to enrich the knowledge graph by extracting additional information for each entity from DBpedia. In particular, the information extracted from DBpedia are:

- *Field*: the branch of medicine which a specific disease belongs to (e.g. cardiology, pulmonology, oncology, etc.);
- *Complications*: an unfavorable evolution or consequence of a disease. Complications generally involve a worsening in severity of disease or the development of new signs, symptoms, or pathological changes which may become widespread throughout the body and affect other organ systems. Thus, complications may lead to the development of new diseases resulting from a previously existing disease. Complications may also arise as a result of various treatments.
- *Treatment*: the attempted cure of a health problem, usually following a medical diagnosis. As a rule, each therapy has indications and contraindications. There are many different types of therapy. Not all therapies are effective. Many therapies can produce unwanted adverse effects.

In this way, it is possible to derive further relationships between the different diseases diagnosed for each patient. For example, two different diseases can be related to each other if they belong to the same branch of medicine (e.g., "sharing" *"Fibrillazione atriale"* and *"Ipertensione arteriosa"* both belong to the branch known as *Cardiology*).

Running Example 1 *(continuing from p. 6)*. Linked concepts for each entity in the clinical note under analysis are shown as follows:

- iperteso → Hypertension (https://dbpedia.org/page/Hypertension)
- displidemico → Dysplidemia (https://dbpedia.org/page/Dyslipidemia)
- IMA → Myocardial Infarction (https://dbpedia.org/page/Myocardial_infarction)
- dispnea → Shortness of breath (https://dbpedia.org/page/Shortness_of_breath)

Furthermore, relations between these medical concepts and with other concepts in the ontology are extracted. For example, "hypertension" can be associated to the concept "Coronary artery disease" by the relation "complication".

4.4. Heterogeneous graph embeddings

In this section, we provide an overview of the state-of-the-art techniques for heterogeneous graph embedding used in this work. The goal of this step is to use these heterogeneous graph embedding techniques to obtain vector representations of each node of the linked knowledge graph previously obtained. In this regard, 4 different state-of-the-art techniques will be used to obtain node embeddings. Subsequently, they will be compared both through visual analysis and through qualitative analysis evaluating their performance on the link prediction task, thus finding, in our case, the best state-of-the-art technique to obtain graph embeddings.

4.4.1. HIN2Vec (proximity-preserving method)

HIN2Vec (Fu et al., 2017) is a neural network model designed to capture the rich semantics embedded in heterogeneous information networks (HINs) by exploiting different types of relationships among nodes. The idea of HIN2Vec is to learn node vectors jointly learning a model for multiple prediction tasks, one for each meta-path. Hence, the goal of the model will be to predict a set of target relationships (specified by meta-paths and the number of hops) between each pair of input nodes. The model used is a single-hidden-layer feedforward neural network that takes a pair of nodes $x, y \in V$ as the input to predict the probabilities $P(r_i|x, y)$ $(i = 1..|R|)$ of relationships between x and y in the target relationship set R as the output.

In summary, this conceptual model can be seen as a multi-label classifier, and the three matrices, W_X, W_Y and W_R, collect the feature vectors of input node pairs and their relationships. This conceptual model faces excessive overhead in both training data preparation and model learning processes, for this reason the authors of HIN2Vec proposed a better design.

4.4.2. R-GCN (message-passing method)

R-GCN (Schlichtkrull et al., 2018) is an extension of GCNs (Graph Convolutional Networks) which introduces relation-specific transformations (i.e. which depends on the type and direction of the edge) and has K convolutional layers.

Recall that in GCN, the hidden representation for each node i at $(l + 1)^{th}$ layer is computed by:

$$h_i^{l+1} = \sigma\left(\sum_{j \in N_i} \frac{1}{c_i} W^{(l)} h_j^{(l)}\right) \tag{1}$$

where c_i is a normalization constant, N_i is the set of neighbor indices of node i.

The key difference between R-GCN and GCN is that in R-GCN, edges can represent different relations. In GCN, weight $W^{(l)}$ in equation (1.1) is shared by all edges in layer l. In contrast, in R-GCN, different edge types use different weights and only edges of the same relation type r are associated with the same projection weight $W_r^{(l)}$. So the hidden representation of entities in $(l + 1)^{th}$ layer in R-GCN can be formulated as the following equation:

$$h_i^{l+1} = \sigma\left(W_0^l h_i^l + \sum_{r \in R} \sum_{j \in N_i^r} \frac{1}{c_{i,r}} W_r^{(l)} h_j^{(l)}\right) \tag{2}$$

where N_i^r denotes the set of neighbor indices of node i under relation $r \in R$ and $c_{i,r}$ is a normalization constant. The problem of applying the

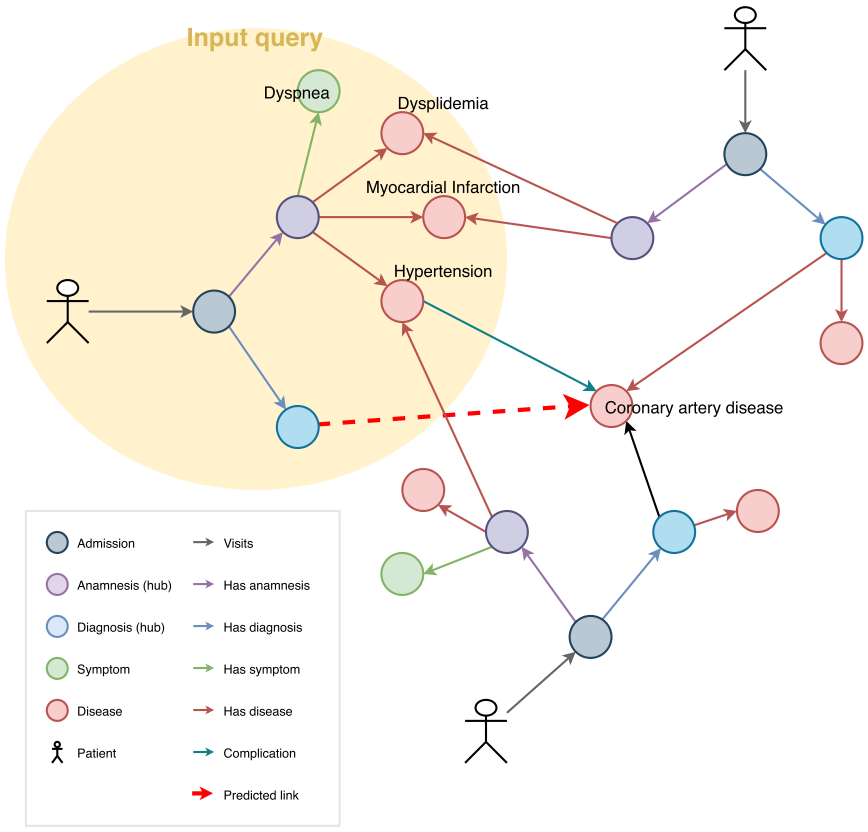

Fig. 3. Example of link prediction given an input query.

above equation directly is the rapid growth of the number of parameters, especially with highly multi-relational data. In order to reduce model parameter size and prevent overfitting, the authors of the paper propose to use basis decomposition.

$$W_r^{(l)} = \sum_{b=1}^{B} a_{rb}^{(l)} V_b^{(l)} \tag{3}$$

Therefore, the weight $W_r^{(l)}$ is a linear combination of basis transformation $V_b^{(l)}$ with coefficients $a_{rb}^{(l)}$. The number of bases B is much smaller than the number of relations in the knowledge base. Finally, the node embedding of node i is the output of the K-th layer $h_i^{(K)}$.

4.4.3. DistMult (relation-learning method)

Given a triplet (subject entity (alias head), relation and object entity (alias tail)), DistMult (Yang et al., 2015) uses a diagonal matrix only, instead of using multiple matrics (as approaches such as RESCAL (Nickel & Tresp, 2011)), to represent the relations among entities. The similarity based scoring function $s_l(u, v)$ is defined using a bilinear function:

$$s_l(u, v) = e_u^T A_l e_v \tag{4}$$

where $A_l = diag(e_{l1}, ..., e_{ld})$ is the diagonal matrix, while e_u and e_v are node embeddings of u and v.

A great advantage of this approach, compared to previous approaches such as RESCAL (Nickel & Tresp, 2011), is the number of parameters used in the training phase, which is significantly lower.

4.4.4. ConvE (relation-learning method)

ConvE (Dettmers et al., 2018) goes beyond simple distance or similarity functions and proposes deep neural models to score a triplet. In ConvE the interactions between input entities and relationships are modeled by convolutional and fully-connected layers. The main characteristic of this model is that the score is defined by a convolution over 2D shaped embeddings. Formally:

$$s_l(u, v) = \sigma(vec(\sigma([E_u; E_r] * w))W)e_v \tag{5}$$

where E_u and E_r denote the 2D reshaping matrices of node embedding and relation embedding, respectively; vec is the vectorization operator that maps a m by n matrix to a mn-dimensional vector; "$*$" is the convolution operator.

4.5. Link prediction

Following the approach from Yang et al. (2020), we use the Hadamard function to construct feature vectors for node pairs, train a two-class LinearSVC on the 80% training links and evaluate towards the 20% held out links. We repeat the process for standard five-fold cross validation and compute the average scores regarding AUC (area under the ROC curve) and MRR (mean reciprocal rank). AUC is a standard measure for binary classification problems, while MRR is a standard measure for ranking (link prediction can be considered as a link retrieval problem).

Running Example 1 *(continuing from p. 7).* By leveraging the heterogeneous graph built in the previous steps and the embeddings obtained with state-of-the-art methods, we are able to perform link prediction between diagnoses and disorders. An example for the patient under analysis is shown in Fig. 3. Given the concepts extracted from the clinical note and their connections to the rest of the heterogeneous graph, the link prediction framework picks "Coronary artery disease" as the most likely disorder to be diagnosed. This is supported not only by the similarities to other patients, but also by the *complication* link between hypertension and the diagnosed disorder.

5. Experiments

In this section, we discuss the obtained results in each step of the designed methodology. In particular:

- Datasets, knowledge bases, metrics and training parameters used for the entity linking step will be described.
- Comparison between obtained results, paying attention to the difference between before and after.
- Discussions analyzing the results, about the advantages that each step entails.

5.1. Experimental protocol

In our case study, we evaluate the performance of an entity linking model on 3 different biomedical datasets (2 English and 1 Italian) and on 2 different external knowledge bases (DBpedia and MeSH). Among all the results, we focus on carrying out the normalization of the entities in the graph, and the subsequent enrichment. Those are obtained by considering the Wincare dataset and the DBpedia external knowledge base in Italian as the input of the entity linking model.

All the experiments were carried out through Google Colaboratory,[3] a platform-as-a-service that provides a virtual machine equipped by the TESLA T4 GPU with 16 GB of RAM. The entire used code is available at the following link, https://colab.research.google.com/.

5.2. Dataset

Three different biomedical datasets were used for biomedical entity linking task. In particular, we will first describe the two existing biomedical datasets known as *NCBI-Disease* (Doğan et al., 2014) and *BC5CDR* (Li et al., 2016). Then we describe the biomedical dataset specifically created for the biomedical entity linking task on Italian data, called *Wincare*. The two existing datasets share the same format, known as PubTator format: a representative index of the document, textual content of the document, a set of entity mentions that appear in the document with the corresponding initial and final offsets, and an alphanumeric label (Mesh ID code).

1. **NCBI-Disease** is a corpus belonging to the biomedical domain annotated starting from textual data present in 793 PubMed abstracts, for a total of 6,892 disease mentions linked to 790 unique disease concepts obtained from thesauri such as: Medical Subject Headings (MeSH) and Online Mendelian Inheritance in Man (OMIM). The corpus was developed by a team of 12 annotators (two people per annotation) and covers all sentences in a PubMed abstract. Disease mentions are categorized into Specific Disease, Disease Class, Composite Mention and Modifier categories. The corpus characteristics are the following: 5148 disease mentions in the training set, 791 in the development set and 961 in the test set.
2. **BC5CDR** is a corpus that consists of 1500 PubMed articles with 4409 annotated chemicals, 5818 diseases and 3116 chemical-disease interactions. Each entity annotation includes both the mention text spans and normalized concept identifiers, using MeSH as the controlled vocabulary. To ensure accuracy, the entities were first captured independently by two annotators followed by a consensus annotation. The corpus characteristics are the following: 9385 between disease and chemical mentions in the training set, 9591 in the development set and 9807 in the test set.
3. **Wincare** is a corpus belonging to the biomedical domain provided by the *Hospital of Naples Federico II* annotated starting from clinical records provided by the Italian hospital, for a total of 1000 disease mentions linked to unique concepts obtained from thesauri such as: DBpedia and Medical Subject Heading (MeSH). This dataset was not built from scratch, but was already labeled for the Name Entity Recognition (NER) task by identifying each token as Disease, Symptom, or neither (O). For the labeling necessary for the entity linking task, it was sufficient to assign an alphanumeric label corresponding to a MeshID code to each mention labeled as Disease. The corpus characteristics are the following: 700 disease mentions in the training set, 150 in the development set and 150 in the test set.

[3] https://colab.research.google.com

5.3. Dataset and knowledge base preprocessing

Given the Pubtator format, for the fine-tuning of the entity linking model shown in Section 4.3, the input dataset must be transformed into a suitable format. Specifically, the information that the model needs to be trained are:

- *Context left*: tokens that appear on the left of the term considered as entity mention.
- *Context right*: tokens that appear on the right of the term considered as entity mention.
- *Mention*: the token(s) corresponding to the entity mention.
- *Label*: description of the mapping entity.
- *Label ID*: numerical identifier of the mapping entity considered for that entity mention.
- *Label title*: title of the mapping entity (of the external knowledge base) for that entity mention considered.

To extract the information on each mention and the corresponding context, a first preprocessing step was necessary in order to subsequently obtain the information on *mention*, *context*, and *label*.

For the entities extracted from the two considered external knowledge bases (DBpedia and MeSH), starting from the JSON file produced as a result of the sparql queries, a pre-processing step is carried out to obtain a numeric identifier for each entity in the knowledge base. In particular, four different json files are produced:

- *idx to cui*; associates the corresponding mesh ID code to each identifier;
- *cui to idx*; associates the corresponding identifier to each meshID code;
- *cui to cano*; associates the corresponding entity title to each meshID code;
- *cui to def*; associates the corresponding description of the entity to each meshID code.

In this way, it is possible to exploit both the information obtained through pre-processing of the dataset, and from the external knowledge base, retrieved through a sparql query. We show all the information needed to train the entity linking model as follows:

```
{"context_left": "Ketoconazole induced",
 "context_right": "Without concomitant use of QT
 interval-prolonging drug."
 "mention": "torsades de pointes",
 "label": "A malignant form of polymorphic
 ventricular tachycardia that is characterized by
 HEART RATE between 200 and 250 beats per minute,
 and QRS complexes with changing amplitude"
 "label_id:" "26039",
 "label_title": "Torsades de Pointes"}
```

5.4. Training parameters

For the entity normalization and linking step, the considered entity linking model is made up of two different components. For this rea-

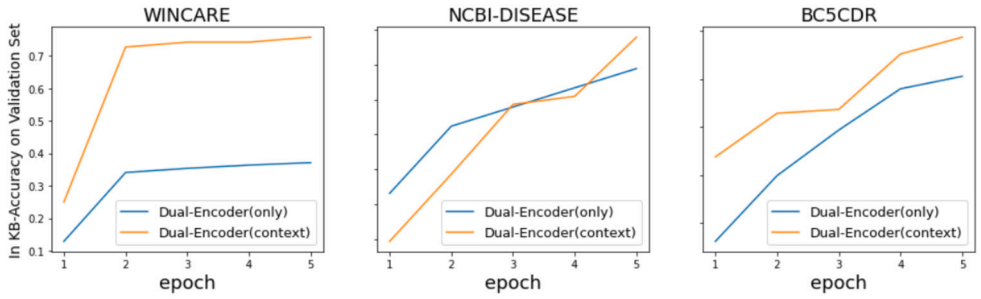

Fig. 4. Comparison of Dual-encoder results with (orange) and without (blue) contextual information.

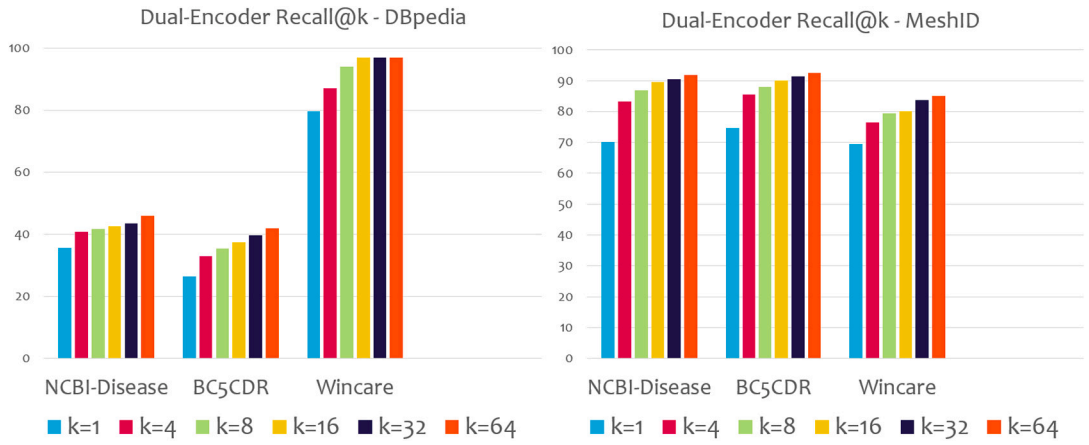

Fig. 5. Comparison between DBpedia and MeSH for the Candidate Generation step (Dual-Encoder).

son, we separately describe the training parameters used for the two components:

- Dual-Encoder parameters; the hyperparameter configuration for best model is: learning rate = 2e-5, train batch size = 16, max sequence length = 256 (max context length = 128, max candidate length = 128), epochs = 5, warmup proportion = 0.1, gradient accumulation steps = 1.
- Cross-Encoder parameters; the hyperparameter configuration for best model is: learning rate = 1e-5, train batch size = 2, max sequence length = 256 (max context length = 128, max candidate length = 128), epochs = 2.

For the heterogeneous graph representation learning, four different state of the art techniques were trained:

- *HIN2Vec*: parameters settings in HIN2Vec affect the node representation learning and the application performance (link prediction). In particular, the first parameter is the embedding dimension. Generally speaking, a small size is not sufficient to capture the information embedded in relationships between nodes, but a large size may lead to noises and cause overfitting. For this reason, a good value is size = 50. The performance does not change much when the number of negative sampling per positive sample "negative" is set at between 3 to 7. Thus, setting "negative" to 5 is a good choice. The initial learning rate is set to alpha = 0.025, the length of each random walk is set to 100, the number of random walks starting for each node is set to 10.
- *R-GCN*: the embedding dimension is set to size = 50, the number of negative sample is set to 5, the initial learning rate is set to lr = 0.01, dropout = 0.2, regularization weight = 0.01, to avoid the explosion of the gradient the grad clipping parameter is set to 1.0, the number of epochs is set to 1000.
- *DistMult*: training was implemented using mini-batch stochastic gradient descent with AdaGrad. The embedding dimension of each entity vector is set to size = 50, the batch size is set to 128 and the number of training epochs T = 300. The learning rate was initially set to 0.003 and then adapted during training by AdaGrad (learning rate decay = 0.0995).
- *ConvE*: the embedding dimension is set to size = 50, the number of epochs is set to 300, with learning rate lr = 0.002, learning rate decay = 0.0995 and batch size = 128.

6. Results

6.1. Candidate entity generation

6.1.1. Effects of contextual information

The first step for solving the entity linking task consists in the generation of the candidate entities for each entity mention, through the previously defined Dual-Encoder. In Fig. 4 it is possible to compare the results obtained by two different Dual-Encoders that take as input two different representations: the first does not consider context of the mention and description of the candidate entities, whereas the second does it.

Through the obtained results, it is possible to note how the contextual information both for the mention and for the candidate entities significantly improves the performance of the Dual-Encoder, especially for the Italian dataset "Wincare".

6.1.2. Effects of DBpedia and MeSH knowledge bases

Afterwards, we compare two different external knowledge bases: DBpedia and MeSH. In Fig. 5 it is possible to see the recall@K results obtained by the Dual-Encoder on the 3 considered datasets, as the parameter k changes. The results obtained show, as k varies, the probability that among the first k returned candidate entities, there is the correct mapping entity associated with each entity mention.

Table 4
Overall results of the Entity Linking task.

Method	Wincare	NCBI-Disease	BC5CDR
BM25	23.48	13.21	9.27
BERT similarity	50.76	29.71	60.02
Dual-Encoder	25.76	66.02	74.90
Dual-Encoder (context)	79.55	70.15	74.78
Cross-Encoder	**97.58**	**92.53**	**95.49**

Specifically, for the Italian Wincare dataset the best results are obtained with the DBpedia knowledge base, as it contains biomedical entities translated into Italian with the corresponding description. MeSH knowledge base, instead, is strictly in English and for this reason the performances obtained are worse. For the two English datasets considered, the external MeSH knowledge base allows to obtain better results for the generation of the candidate entities. This is due to the knowledge base of biomedical domain that contains a number of entities significantly higher than the base of DBpedia knowledge considered (only biomedical entities within it).

For this reason, the DBpedia knowledge base (in the Italian version) was used for the Wincare dataset for the following steps, while the MeSH knowledge base was used for the two other datasets.

6.2. Reranking entities

Since the in-KB accuracy results provided by the Dual-Encoder are not satisfactory, a second step was necessary for the resolution of the entity linking task. It consists in the re-ranking of the candidate entities provided by the Bi-Encoder, through a model called Cross-Encoder. By evaluating the performance on the test sets corresponding to each considered dataset, it is possible to note how the addition of the Cross-Encoder allows to obtain excellent results for the prediction of the mapping entity for each mention. In particular, Table 4 shows how the results have been improved using increasingly complex models.

Specifically, the approaches used are:

- **BM25** (Robertson & Zaragoza, 2009). Okapi BM25 is a ranking function used by search engines to estimate the relevance of documents to a given search query. Considering each mention as a query and the set of entities in the considered knowledge base (with related descriptions) as documentary collection, the mapping entity was obtained by considering the entity in the documentary collection to which the highest score is assigned. This approach is an evolution of the classic TF-IDF approach, based on counting how many times the query repeats itself in each document and in how many different documents it appears. The results obtained are bad, as the documentary collection considered is only a set of descriptions of each entity and not "real" documents.
- **BERT similarity** (Devlin et al., 2018). BERT-base-uncased used for cosine similarity uses a neural network without fine-tuning it, but simply to obtain a vector representation of each mention and each candidate entity, without considering contextual information. Once these representations are obtained, the cosine similarity is calculated and the mention-candidate entity pair that obtains the highest similarity score is returned.
- **Dual-Encoder** (Wu et al., 2020). A pre-trained BERT-based model is fine-tuned, considering as input only the name of the mention and the title of the entity in the target knowledge base, contextual information is not considered.
- **Dual-Encoder (context)** (Wu et al., 2020). Same model as the previous point, with the difference that in this case it also considers contextual information as input: specifically, context of the mention, and description of the entities.
- **Cross-Encoder** (Wu et al., 2020). It uses the Dual-Encoder to generate the top-8 candidate entities for each mention and a Cross-Encoder to assign a new score to each pair (mention, candidate entity) and obtain as a result the pair with the highest score, for each mention.

Table 5
Link prediction results, before and after enrichment.

Method	Before EL		After EL	
	AUC	*MRR*	*AUC*	*MRR*
HIN2Vec	50.00	78.01	50.00	78.40
R-GCN	53.63	78.91	58.72	82.38
ConvE	68.884	86.30	69.53	88.10
DistMult	78.86	91.73	79.69	93.19

An important aspect is the choice of the number of candidate entities returned by the Dual-Encoder for subsequent re-ranking via Cross-Encoder. Obviously, the greater the number of candidate entities, the better the results obtained. At the same time, the greater the number of candidate entities, the greater the amount of RAM memory required for Cross-Encoder training. For this reason, it is necessary to find a trade-off between performance and computational costs. From the results obtained in Fig. 5, it is possible to notice how choosing 8 as the number of candidate entities, the recall results obtained are similar to the case with $k = 32$ or $k = 64$. However, the amount of RAM required with $k = 8$ will not exceed 16 GB, while for higher values of k (i.e. 16,32,64) 16 GB was not enough.

6.3. Link prediction

After obtaining the node embeddings using state-of-the-art techniques of heterogeneous graph representation learning, the performance on the link prediction task was evaluated predicting whether or not a link is present between two Diagnosis-Disease type nodes. In particular, the results obtained before and after enrichment of the knowledge graph through DBpedia were compared.

From the obtained results, as shown in Table 5, we can draw two important remarks:

1. Among the four state-of-the-art techniques for generating node embeddings on heterogeneous graphs, the one that allows to obtain better vector representations is certainly DistMult. It manages to obtain better performances than AUC and MRR on the link prediction task, both on the starting knowledge graph, and on the one obtained downstream of the enrichment through DBpedia.
2. It is possible to notice how the addition of nodes and links to the normalized knowledge graph allows to improve the performance of all the state-of-the-art techniques used, confirming however the claim that DistMult is the best technique for the generation of node embeddings.

After the entity normalization step, within the created biomedical knowledge graph, we will have a series of nodes corresponding to the ones present in the DBpedia external knowledge base. Starting from these, it is possible to extract further knowledge from DBpedia. Fig. 6 shows an example of how the knowledge graph changes after having enriched it. Different disease nodes will be connected to each other, by exploiting the information on specialization (HAS SPECIALIZATION) of each disease (e.g., arterial hypertension and atrial fibrillation both have cardiology as their specialization). In addition, as shown in Fig. 6, two different diseases can be linked to each other through the relationship "HAS COMPLICATION", i.e., one disease can degenerate into the other due to complications.

The results obtained from the considered four state-of-the-art heterogeneous graph embedding techniques are visually compared by using t-SNE to reduce the dimensionality of embeddings. We report the comparison in Fig. 7. In particular, the HIN2Vec and R-GCN techniques do not allow to discriminate the type of entity (e.g., Disease, Symptom,

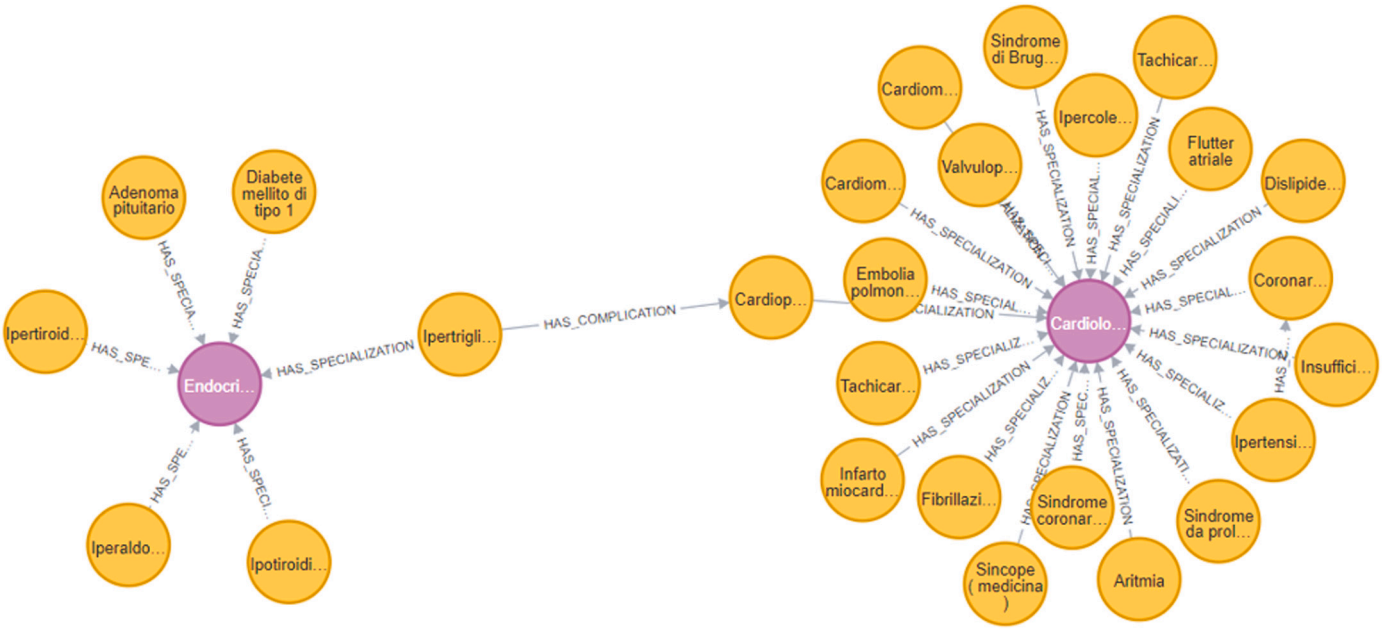

Fig. 6. Example of adding nodes in the graph.

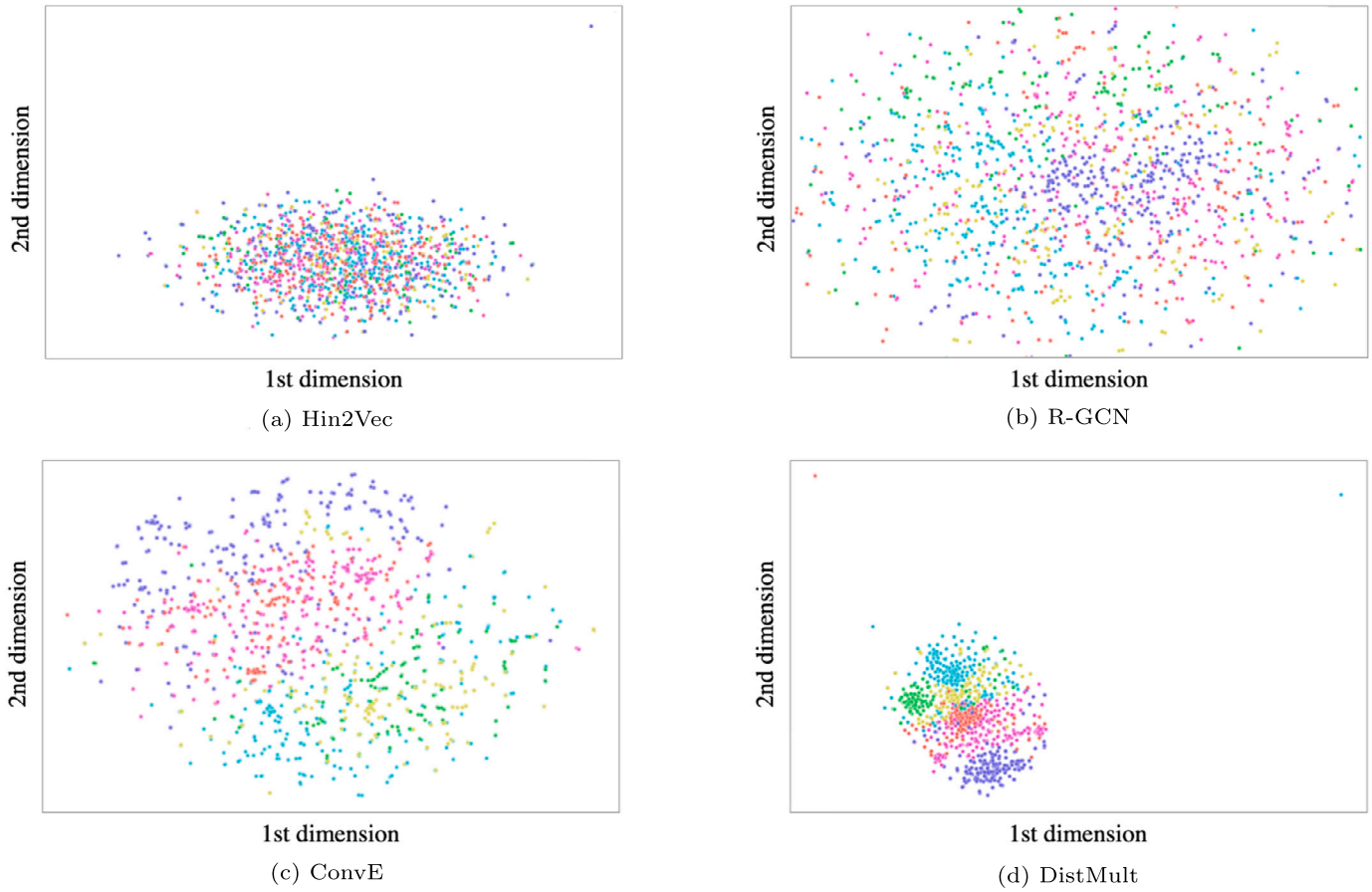

Fig. 7. Visual comparison of embedding methods. Plots are obtained by reducing embeddings dimensionality with t-SNE. Colors refer to node types.

Patient, Visit, etc.) in a two-dimensional space, as entities of different types appear in the same portion of the graph, while with more advanced techniques — i.e., based on relation learning methods, such as DistMult and ConvE through a simple visual analysis — it is possible to see how entities of different types in the knowledge graph will be represented in a different portion of the two-dimensional space.

7. Conclusions & future work

In this paper we have shown the value of entity linking in the construction and prospective analysis of healthcare knowledge graphs. Specifically, we can summarize our work as follows.

- With biomedical entity linking experiments based on pre-trained transformers and carried out on two well-known English biomedical datasets (i.e. NCBI-Disease, BC5CDR), we have shown that excellent results can be obtained with a two-phases approach consisting in (1) the generation of candidates by means of a dual-encoder, and (2) the re-ranking of candidate entities through a cross-encoder.
- Results from the dual-encoder have been used to compare DBpedia and MeSH knowledge bases. We observed that a domain-specific knowledge base (i.e., MeSH) leads to better performance w.r.t. a generic knowledge base (i.e., DBpedia) in generating candidate entities.
- However, the Dual-Encoder alone did not allow to obtain good results on test sets. For this reason, we have added a Cross-Encoder, which has significantly improved the results on both datasets.
- We have tested the applicability of the proposed entity linking model on Italian data from cardiology departments of the *Hospital of Naples Federico II*. Given the language of this dataset, it was not possible to use MeSH, but we have considered Italian concepts from DBpedia. From the results obtained from the fine-tuning of the BERT transformers inside the model, it emerged the ability to correctly predict mentions of the test set never seen in the training phase.
- Taking a cue from previous works that used graph embeddings to improve the performance of entity linking task, a reverse approach was carried out: exploiting results of the entity linking task to improve graph embeddings (obtained through state-of-the-art techniques) of the biomedical knowledge graph previously created. In particular, from results obtained for the link prediction task, it has been shown how the addition of information to the graph through DBpedia, and the results of the entity linking task allow to improve the graph embeddings obtained by all the considered state-of-the-art techniques.

Future work will be devoted to improvements of the information extraction steps in our pipeline. The extraction of entity mentions and, thus, the linking procedure is indeed challenging, due to the low-resource scenario imposed by the Italian language. There is room for improvements in this direction by leveraging few-shot learning techniques (e.g., data augmentation, distant supervision, meta-learning). Furthermore, another key aspect to improve the performance of the current framework consists in taking temporal information into consideration: in fact, the analysis of the dynamic of past events has recently shown promising results in several domains (Zhu et al., 2021, Li et al., 2021), and consequently healthcare could benefit from such works too.

CRediT authorship contribution statement

Conception and design of study: D. D'Auria, V. Moscato, M. Postiglione, G. Romito, G. Sperlì; Acquisition of data: D. D'Auria, V. Moscato, M. Postiglione, G. Romito, G. Sperlì; Analysis and/or interpretation of data: D. D'Auria, V. Moscato, M. Postiglione, G. Romito, G. Sperlì. Drafting the manuscript: D. D'Auria, V. Moscato, M. Postiglione, G. Romito, G. Sperlì; Revising the manuscript critically for important intellectual content: D. D'Auria, V. Moscato, M. Postiglione, G. Romito, G. Sperlì.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data that has been used is confidential.

References

Abul-Husn, N. S., & Kenny, E. E. (2019). Personalized medicine and the power of electronic health records. *Cell*, *177*, 58–69.

Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., & Ives, Z. (2007a). Dbpedia: A nucleus for a web of open data. In *The semantic web* (pp. 722–735). Springer.

Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., & Ives, Z. G. (2007b). Dbpedia: A nucleus for a web of open data. In *ISWC/ASWC*.

Behrouzi, S., Shafaeipour Sarmoor, Z., Hajsadeghi, K., & Kavousi, K. (2020). Predicting scientific research trends based on link prediction in keyword networks. *Journal of Informetrics*, *14*, Article 101079. https://doi.org/10.1016/j.joi.2020.101079.

Bhowmik, R., Stratos, K., & de Melo, G. (2021). Fast and effective biomedical entity linking using a dual encoder. In *Proceedings of the 12th international workshop on health text mining and information analysis* (pp. 28–37). Association for Computational Linguistics. Online.

Bodenreider, O. (2004). The unified medical language system (umls): Integrating biomedical terminology. *Nucleic Acids Research*, *32*, D267–D270.

Broscheit, S. (2019). Investigating entity knowledge in BERT with simple neural end-to-end entity linking. In *Proceedings of the 23rd conference on computational natural language learning (CoNLL)* (pp. 677–685). Hong Kong, China: Association for Computational Linguistics. https://aclanthology.org/K19-1063.

Chen, L., Li, X., Sheng, Q. Z., Peng, W., Bennett, J., Hu, H., & Huang, N. (2016). Mining health examination records - a graph-based approach. *IEEE Transactions on Knowledge and Data Engineering*, *28*, 2423–2437. https://doi.org/10.1109/TKDE.2016.2561278.

Chen, Z., & Ji, H. (2011). Collaborative ranking: A case study on entity linking. In *Proceedings of the 2011 conference on empirical methods in natural language processing* (pp. 771–781). Edinburgh, Scotland, UK: Association for Computational Linguistics. https://aclanthology.org/D11-1071.

Chen, Z., Tamang, S. R., Lee, A., Li, X., Lin, W.-P., Snover, M. G., Artiles, J., Passantino, M., & Ji, H. (2010). Cuny-blender tac-kbp2010 entity linking and slot filling system description. *Theory and Applications of Categories*.

Choi, E., Bahadori, M. T., Song, L., Stewart, W. F., & Sun, J. (2017). Gram: Graph-based attention model for healthcare representation learning. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining KDD '17* (pp. 787–795). New York, NY, USA: Association for Computing Machinery.

Choi, E., Xiao, C., Stewart, W. F., & Sun, J. (2018). Mime: Multilevel medical embedding of electronic health records for predictive healthcare. In *Proceedings of the 32nd international conference on neural information processing systems* (pp. 4552–4562). Red Hook, NY, USA: Curran Associates Inc.

Choi, E., Xu, Z., Li, Y., Dusenberry, M. W., Flores, G., Xue, Y., & Dai, A. M. (2020). Learning the graphical structure of electronic health records with graph convolutional transformer. In *AAAI*.

Cucerzan, S. (2007). Large-scale named entity disambiguation based on Wikipedia data. In *Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)* (pp. 708–716). Prague, Czech Republic: Association for Computational Linguistics. https://aclanthology.org/D07-1074.

Deorowicz, S., & Ciura, M. (2005). Correcting spelling errors by modelling their causes. *International Journal of Applied Mathematics and Computer Science*, *15*, 275–285.

Dettmers, T., Minervini, P., Stenetorp, P., & Riedel, S. (2018). Convolutional 2d knowledge graph embeddings. In *Thirty-second AAAI conference on artificial intelligence*.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint, arXiv:1810.04805.

Doğan, R. I., Leaman, R., & Lu, Z. (2014). Ncbi disease corpus: A resource for disease name recognition and concept normalization. *Journal of Biomedical Informatics*, *47*, 1–10. https://doi.org/10.1016/j.jbi.2013.12.006.

Fu, T-y., Lee, W.-C., & Lei, Z. (2017). Hin2vec: Explore meta-paths in heterogeneous information networks for representation learning. In *Proceedings of the 2017 ACM on conference on information and knowledge management* (pp. 1797–1806).

Gao, S., Alawad, M., Young, M. T., Gounley, J., Schaefferkoetter, N., Yoon, H. J., Wu, X.-C., Durbin, E. B., Doherty, J., Stroup, A., Coyle, L., & Tourassi, G. (2021). Limitations of transformers on clinical text classification. *IEEE Journal of Biomedical and Health Informatics*, *25*, 3596–3607. https://doi.org/10.1109/JBHI.2021.3062322.

Gao, S., Denoyer, L., & Gallinari, P. (2011). Temporal link prediction by integrating content and structure information. In *Proceedings of the 20th ACM international conference on information and knowledge management* (pp. 1169–1174). New York, NY, USA: Association for Computing Machinery.

Hu, Z., Dong, Y., Wang, K., & Sun, Y. (2020). Heterogeneous graph transformer. In *Proceedings of the web conference 2020* (pp. 2704–2710). New York, NY, USA: Association for Computing Machinery.

Kormilitzin, A., Vaci, N., Liu, Q., & Nevado-Holgado, A. (2021). Med7: A transferable clinical natural language processing model for electronic health records. *Artificial Intelligence in Medicine*, *118*, Article 102086. https://doi.org/10.1016/j.artmed.2021.102086.

Kraljevic, Z., Searle, T., Shek, A., Roguski, L., Noor, K., Bean, D., Mascio, A., Zhu, L., Folarin, A. A., Roberts, A., et al. (2021). Multi-domain clinical natural language processing with medcat: The medical concept annotation toolkit. *Artificial Intelligence in Medicine*, *117*, Article 102083.

Le, P., & Titov, I. (2019). Distant learning for entity linking with automatic noise detection. In *Proceedings of the 57th annual meeting of the association for computational linguistics* (pp. 4081–4090). Florence, Italy: Association for Computational Linguistics.

Li, J., Sun, Y., Johnson, R. J., Sciaky, D., Wei, C.-H., Leaman, R., Davis, A. P., Mattingly, C. J., Wiegers, T. C., & Lu, Z. (2016). Biocreative v cdr task corpus: a resource for chemical disease relation extraction. *Database*, *2016*. https://doi.org/10.1093/database/baw068.

Li, Z., Jin, X., Li, W., Guan, S., Guo, J., Shen, H., Wang, Y., & Cheng, X. (2021). Temporal knowledge graph reasoning based on evolutional representation learning. In F. Diaz, C. Shah, T. Suel, P. Castells, R. Jones, & T. Sakai (Eds.), *SIGIR '21: the 44th international ACM SIGIR conference on research and development in information retrieval, virtual event* (pp. 408–417). ACM.

Lipscomb, C. E. (2000). Medical subject headings (mesh). *Bulletin of the Medical Library Association*, *88*, 265.

Liu, Z., Li, X., Peng, H., He, L., & Yu, P. S. (2020). Heterogeneous similarity graph neural network on electronic health records. In *2020 IEEE international conference on big data (Big Data)* (pp. 1196–1205). IEEE.

Ma, F., You, Q., Xiao, H., Chitta, R., Zhou, J., & Gao, J. (2018). Kame: Knowledge-based attention model for diagnosis prediction in healthcare. In *Proceedings of the 27th ACM international conference on information and knowledge management* (pp. 743–752). New York, NY, USA: Association for Computing Machinery.

Moscato, V., & Sperlì, G. (2022). Community detection over feature-rich information networks: An ehealth case study. *Information Systems*, *109*, Article 102092. https://doi.org/10.1016/j.is.2022.102092.

Negro-Calduch, E., Azzopardi-Muscat, N., Krishnamurthy, R. S., & Novillo-Ortiz, D. (2021). Technological progress in electronic health record system optimization: Systematic review of systematic literature reviews. *International Journal of Medical Informatics*, *152*, Article 104507. https://doi.org/10.1016/j.ijmedinf.2021.104507.

Nickel, M., & Tresp, V. (2011). A three-way model for collective learning on multi-relational data.

Onoe, Y., & Durrett, G. (2020). Fine-grained entity typing for domain independent entity linking. In *Proceedings of the AAAI conference on artificial intelligence: Vol. 34* (pp. 8576–8583).

Park, J., Cho, Y., Lee, H., Choo, J., & Choi, E. (2021). Knowledge graph-based question answering with electronic health records. In *Machine learning for healthcare conference* (pp. 36–53). PMLR.

Parravicini, A., Patra, R., Bartolini, D. B., & Santambrogio, M. D. (2019). Fast and accurate entity linking via graph embedding. In *Proceedings of the 2nd joint international workshop on graph data management experiences & systems (GRADES) and network data analytics (NDA)*. New York, NY, USA: Association for Computing Machinery.

Perozzi, B., Al-Rfou, R., & Skiena, S. (2014). Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 701–710). New York, NY, USA: Association for Computing Machinery.

Persia, F., Costantini, S., Ferri, C., Lauretis, L. D., & D'Auria, D. (2021). A smart framework for automatically analyzing electrocardiograms. In *2021 third international conference on transdisciplinary AI (TransAI)* (pp. 64–67). https://doi.org/10.1109/TransAI51903.2021.00019.

Pham, T., Tao, X., Zhan, J., Yong, J., Zhang, W., & Cai, Y. (2018). Mining heterogeneous information graph for health status classification. In *5th international conference on behavioral, economic, and socio-cultural computing* (pp. 73–78). IEEE.

Rajkomar, A., Oren, E., Chen, K., Dai, A. M., Hajaj, N., Hardt, M., Liu, P. J., Liu, X., Marcus, J., Sun, M., Sundberg, P., Yee, H., Zhang, K., Zhang, Y., Flores, G., Duggan, G. E., Irvine, J., Le, Q. V., Litsch, K., … Dean, J. (2018). Scalable and accurate deep learning with electronic health records. *npj Digital Medicine*, *1*.

Ramshaw, L. A., & Marcus, M. P. (1999). Text chunking using transformation-based learning. In *Natural language processing using very large corpora* (pp. 157–176). Springer.

Robertson, S., & Zaragoza, H. (2009). The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends in Information Retrieval*, *3*, 333–389. https://doi.org/10.1561/1500000019.

Schlichtkrull, M., Kipf, T. N., Bloem, P., Van Den Berg, R., Titov, I., & Welling, M. (2018). Modeling relational data with graph convolutional networks. In *European semantic web conference* (pp. 593–607). Springer.

Schork, N. J. (2015). Personalized medicine: Time for one-person trials. *Nature*, *520*, 609–611.

Shen, W., Wang, J., & Han, J. (2015). Entity linking with a knowledge base: Issues, techniques, and solutions. *IEEE Transactions on Knowledge and Data Engineering*, *27*, 443–460. https://doi.org/10.1109/TKDE.2014.2327028.

Shen, W., Wang, J., Luo, P., & Wang, M. (2012). Liege: Link entities in web lists with knowledge base. In *Proceedings of the 18th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 1424–1432). New York, NY, USA: Association for Computing Machinery.

Shen, W., Wang, J., Luo, P., & Wang, M. (2013). Linking named entities in tweets with knowledge base via user interest modeling. In *Proceedings of the 19th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 68–76). New York, NY, USA: Association for Computing Machinery.

Sun, C., Li, Q., Cui, L., Li, H., & Shi, Y. (2019). Heterogeneous network-based chronic disease progression mining. *Big Data Mining and Analytics*, *2*, 25–34. https://doi.org/10.26599/BDMA.2018.9020009.

Wang, Z., Wen, R., Chen, X., Cao, S., Huang, S., Qian, B., & Zheng, Y. (2021). Online disease diagnosis with inductive heterogeneous graph convolutional networks. In J. Leskovec, M. Grobelnik, M. Najork, J. Tang, & L. Zia (Eds.), *WWW '21: The web conference 2021, virtual event* (pp. 3349–3358). ACM/IW3C2.

Wanyan, T., Honarvar, H., Azad, A., Ding, Y., & Glicksberg, B. S. (2021). Deep learning with heterogeneous graph embeddings for mortality prediction from electronic health records. *Data Intelligence*, *3*, 329–339. https://doi.org/10.1162/dint_a_00097.

Wanyan, T., Kang, M., Badgeley, M. A., Johnson, K. W., Freitas, J. K. D., Chaudhry, F. F., Vaid, A., Zhao, S., Miotto, R., Nadkarni, G. N., Wang, F., Rousseau, J. F., Azad, A., Ding, Y., & Glicksberg, B. S. (2020). Heterogeneous graph embeddings of electronic health records improve critical care disease predictions. In M. Michalowski, & R. Moskovitch (Eds.), *Artificial intelligence in medicine - 18th international conference on artificial intelligence in medicine, AIME 2020, Minneapolis, MN, USA, August 25-28, 2020, proceedings. Lecture notes in computer science: Vol. 12299* (pp. 14–25). Springer.

Wiatrak, M., & Iso-Sipila, J. (2020). Simple hierarchical multi-task neural end-to-end entity linking for biomedical text. In *Proceedings of the 11th international workshop on health text mining and information analysis* (pp. 12–17).

Wu, L., Petroni, F., Josifoski, M., Riedel, S., & Zettlemoyer, L. (2020). Scalable zero-shot entity linking with dense entity retrieval. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)* (pp. 6397–6407). Association for Computational Linguistics. Online. https://aclanthology.org/2020.emnlp-main.519.

Xiao, G., Ding, L., Cogrel, B., & Calvanese, D. (2019). Virtual knowledge graphs: An overview of systems and use cases. *Data Intelligence*, *1*, 201–223. https://doi.org/10.1162/dint_a_00011.

Yang, B., Yih, S. W.-t., He, X., Gao, J., & Deng, L. (2015). Embedding entities and relations for learning and inference in knowledge bases. In *Proceedings of the international conference on learning representations (ICLR) 2015*.

Yang, C., Xiao, Y., Zhang, Y., Sun, Y., & Han, J. (2020). Heterogeneous network representation learning: A unified framework with survey and benchmark. *IEEE Transactions on Knowledge and Data Engineering*, 1. https://doi.org/10.1109/TKDE.2020.3045924.

Yang, H., & Yang, C. C. (2015). Mining a weighted heterogeneous network extracted from healthcare-specific social media for identifying interactions between drugs. In *2015 IEEE international conference on data mining workshop (ICDMW)* (pp. 196–203).

Yang, H., & Yang, C. C. (2016). Discovering drug-drug interactions and associated adverse drug reactions with triad prediction in heterogeneous healthcare networks. In *2016 IEEE international conference on healthcare informatics* (pp. 244–254). IEEE Computer Society.

Yao, Z., Cao, L., & Pan, H. (2020). Zero-shot entity linking with efficient long range sequence modeling. In *Findings of the association for computational linguistics: EMNLP 2020*. Association for Computational Linguistics. Online.

Yoon, W., So, C. H., Lee, J., & Kang, J. (2019). Collabonet: Collaboration of deep neural networks for biomedical named entity recognition. *BMC Bioinformatics*, *20*.

Zhang, R., Hristovski, D., Schutte, D., Kastrin, A., Fiszman, M., & Kilicoglu, H. (2021). Drug repurposing for Covid-19 via knowledge graph completion. *Journal of Biomedical Informatics*, *115*, Article 103696. https://doi.org/10.1016/j.jbi.2021.103696.

Zhang, W., Fang, Y., Liu, Z., Wu, M., & Zhang, X. (2022). mg2vec: Learning relationship-preserving heterogeneous graph representations via metagraph embedding. *IEEE Transactions on Knowledge and Data Engineering*, *34*, 1317–1329. https://doi.org/10.1109/TKDE.2020.2992500.

Zhang, W., Su, J., Tan, C. L., & Wang, W. T. (2010). Entity linking leveraging automatically generated annotation. In *Proceedings of the 23rd international conference on computational linguistics (Coling 2010)* (pp. 1290–1298). Beijing, China: Coling 2010 Organizing Committee. https://aclanthology.org/C10-1145.

Zhang, Z., Cai, J., Zhang, Y., & Wang, J. (2020). Learning hierarchy-aware knowledge graph embeddings for link prediction. In *Proceedings of the AAAI conference on artificial intelligence: Vol. 34* (pp. 3065–3072).

Zhao, M., & Yang, C. C. (2016). Mining online heterogeneous healthcare networks for drug repositioning. In *2016 IEEE international conference on healthcare informatics* (pp. 106–112). IEEE Computer Society.

Zhu, C., Chen, M., Fan, C., Cheng, G., & Zhang, Y. (2021). Learning from history: Modeling temporal knowledge graphs with sequential copy-generation networks. In *Thirty-fifth AAAI conference on artificial intelligence, AAAI 2021, thirty-third conference on innovative applications of artificial intelligence, IAAI 2021, the eleventh symposium on educational advances in artificial intelligence, EAAI 2021, virtual event* (pp. 4732–4740). AAAI Press. https://ojs.aaai.org/index.php/AAAI/article/view/16604.

Zhu, M., Celikkaya, B., Bhatia, P., & Reddy, C. K. (2020). Latte: Latent type modeling for biomedical entity linking. In *Proceedings of the AAAI conference on artificial intelligence: Vol. 34* (pp. 9757–9764).