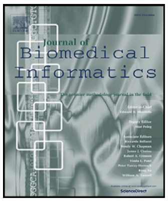
Contents lists available at ScienceDirect

Journal of Biomedical Informatics

journal homepage: www.elsevier.com/locate/yjbin



Original Research

NILINKER: Attention-based approach to NIL Entity Linking

Pedro Ruas [*], Francisco M. Couto

LASIGE, Faculdade de Ciencias, Universidade de Lisboa, Lisbon, 1749-016, Portugal

ARTICLE INFO

Keywords:
Biomedical text
Named Entity Linking
Knowledge Bases
Natural language processing
Neural networks
Text mining

ABSTRACT

The existence of unlinkable (NIL) entities is a major hurdle affecting the performance of Named Entity Linking approaches, and, consequently, the performance of downstream models that depend on them. Existing approaches to deal with NIL entities focus mainly on clustering and prediction and are limited to general entities. However, other domains, such as the biomedical sciences, are also prone to the existence of NIL entities, given the growing nature of scientific literature. We propose NILINKER, a model that includes a candidate retrieval module for biomedical NIL entities and a neural network that leverages the attention mechanism to find the top-k relevant concepts from target Knowledge Bases (MEDIC, CTD-Chemicals, ChEBI, HP, CTD-Anatomy and Gene Ontology-Biological Process) that may partially represent a given NIL entity. We also make available a new evaluation dataset designated by EvaNIL, suitable for training and evaluating models focusing on the NIL entity linking task. This dataset contains 846,165 documents (abstracts and full-text biomedical articles), including 1,071,776 annotations, distributed by six different partitions: EvaNIL-MEDIC, EvaNIL-CTD-Chemicals, EvaNIL-ChEBI, EvaNIL-HP, EvaNIL-CTD-Anatomy and EvaNIL-Gene Ontology-Biological Process. NILINKER was integrated into a graph-based Named Entity Linking model (REEL) and the results of the experiments show that this approach is able to increase the performance of the Named Entity Linking model.

1. Introduction

Named Entity Linking (NEL), also known as disambiguation or normalization, is the task of mapping entities present in free text to entries in a target Knowledge Base (KB) that describe their meaning [1]. NEL tools bridge the gap between natural language and computer-friendly structures, like KBs, which provide a semantic representation simultaneously readable by humans and machines. These tools play a crucial role in the automatic population and curation of KBs [2,3] and, with the growing amount of text present in the Web, a large part of it expressed in natural language, the importance of these applications is only increasing. At the same time, more data means that keeping KBs updated becomes harder and their coverage decreases, so the probability that a given entity has a correspondent KB concept will also decrease. Besides, these tools also directly influence the performance of many other systems and tasks, such as search engines [4], question answering systems [5], electronic health records-based phenotyping [6], pharmacovigilance [7], among others.

In many cases, it is possible to link a recognized entity to a KB concept that is able to express its meaning in the context of the text. This type of entity is designated by in-KB. However, for other entities, there are simply no KB concepts that accurately convey their original meaning, which are then designated by NIL, out-of-KB, unlinkable or CUI-less entities [8]. For example, if we want to link the entity mention “bioinformatician” to the DBpedia ontology,[1] at the moment of writing there is no concept that truly captures its meaning. The closest concept would be “Scientist”, which is not specific enough but represents a substantial portion of the semantics of the entity, and its child concepts, “Biologist”, “Entomologist”, “Medician” and “Professor”, related concepts that do not entirely represent the meaning of the entity. Dredze et al. [3] consider that the absence of KB entries able to represent entities in the text is one of the main challenges that NEL models face. First, it limits their recall, because it defeats the main purpose of the NEL task, which is precisely to link all entity mentions in the text to KB entries. Since this type of system plays an important role in text mining pipelines, the performance of downstream tools that depend upon correct linked entities will also be hindered. Additionally, state-of-the-art tools are injecting knowledge in deep learning models, but if KBs are incomplete that process gets hindered.

Existing approaches to deal with NIL entities focus mainly on predicting them among the entire set of entities to link [9] or in their clustering [10]. For instance, a common approach considers a given

[*] Corresponding author.
E-mail address: psruas@fc.ul.pt (P. Ruas).

[1] https://wiki.dbpedia.org/services-resources/ontology.

https://doi.org/10.1016/j.jbi.2022.104137
Received 25 January 2022; Received in revised form 29 June 2022; Accepted 5 July 2022
Available online 8 July 2022

entity as NIL if the score of the top candidate in the respective candidates' list is below a predefined threshold value. For instance Chen and coworkers [11] proposed a model that learns a threshold value from the training set or instead sets it to zero if there are no NIL entities in the dataset. These approaches limit the semantic information that is possible to obtain from the text, since they just mark those entities as NIL which does not provide semantic value for other text mining tasks. Another approach passes by detecting emergent entities, which, according with the definition by Farber et al. [12], correspond to entities that are *trending* and *notable for the first time* in the context of tasks like semantic media monitoring or automatic speech recognition. Emerging entities are usually associated with real-time events, in which their number of mentions suddenly increase in news articles. Emerging entities correspond to NIL entities. The authors proposed a machine learning-based model to identify emerging entities in news articles among a set including entities that are linkable to Wikipedia and also unlinkable entities. The detected emergent entities with higher confidence score, consisting mainly on living and dead people and politicians, are then suggested for inclusion in Wikipedia. Other approaches [13] use linkable Wikipedia entities to help in the semantic typing of NIL entities. However, approaches focusing on biomedical NIL entities that attempt to leverage NIL entity linking to improve NEL are still scarce. One example is DILBERT [14], which is a neural-based model for medical entity linking that includes a module for out-of-KB prediction.

Similarly to what happens with news articles, scientific articles about diverse topics are constantly being published on large volumes due to the ever changing nature of the scientific landscape. Bornmann and Mutz [15] estimated an increase of 8%–9% in the number of new scientific articles from the end of the Second World War up until 2010. For instance, PubMed added new 952,919 citations only in 2020.[2] There are events that trigger a sudden increase in the number of publications, such as the COVID-19 pandemic. This creates a challenge since many of the entities mentioned in the newly published scientific literature are not yet part of biomedical KBs due to their novelty and due to the effort of manually curating a large volume of the text, so the performance of NEL models decrease. Besides, biomedical KBs miss Wikipedia-specific features. For example Farber et al. [12] leverage Wikipedia page views to detect emergent entities, but this is not possible to replicate in the biomedical domain. At last, there is a lack of evaluation datasets which limits the training and the benchmarking of approaches developed for NIL entity linking.

In the present work, we address the following research questions:

1. How to develop an approach to link NIL entities to biomedical KB concepts?
2. How to improve the performance of biomedical NEL approaches by linking NIL entities to KB concepts?

To tackle the aforementioned challenges, this work proposes an approach to link NIL entities to target biomedical KBs designated by NILINKER, more concretely, by generating KB candidates for the input NIL entity through a word-concept dictionary and then by applying a neural network using the attention model to determine the relevance of each KB candidate. In the last step, the entity is associated with the highest-scoring candidates. For example, this approach would theoretically link the entity mention "bioinformatician" above to the KB concept "Scientist", which would allow the expansion of the target KB through the creation of a new subconcept of "Scientist". Besides, as the existence of NIL entities hinders the performance in the NEL task, NILINKER was integrated jointly with a NEL model and the impact of this integration was assessed on several evaluation datasets. The main contributions of this work are thus:

- The novel approach called **NILINKER** that, instead of detecting and clustering NIL entities, attempts to link them to available KB concepts. The model associates the NIL entity with the top-k relevant KB concepts.
- The dataset **EvaNIL** to train and benchmark NIL entity linking models.
- The model **REEL-NILINKER**, that performs biomedical NEL leveraging NILINKER to deal with NIL entities.

The main novelties of our work are the explicit focus on developing a model to link the NIL entities and not just predict them and the proposed approach to train such model: taking entity mentions in NEL datasets and converting them to "artificial NIL entities", forcing the model to link them to the direct ancestor of the original target KB concept. This way, every entity mention appearing in a given document is leveraged and the available semantic information increases.

The code associated with the models and the performed experiments is fully available.[3] Relevant prior work are described in the next section.

2. Related work

2.1. Biomedical named entity linking

Besides NIL entities, there are other challenges in biomedical NEL, such as entity name variations (e.g. the symbols "DIF", "TNFA", "TNFSF2", "TNLG1F", or "TNF-alpha" refer to the same gene, "tumour necrosis factor") and ambiguity (e.g. "iris" can be both an anatomical structure of the eye or a genus of plants).

Proposed approaches for the NEL task can be broadly divided into local or global. Local models attempt to link each entity mention to the most appropriate concept in the target KB using local evidences (e.g. lexical similarity between mention and candidates), independently of the global context (i.e. the other entity mentions appearing in the same document) [16,17]. On the other hand, global models attempt to maximize the coherence of several linking decisions, i.e., the linking decision of a given entity mention is influenced by the linking of the other entities present in the same document and vice-versa [11,18].

Besides this division, it is possible to further classify the models into rule-based, graph-based or machine learning/deep learning-based. One example of rule-based model was proposed by [17], which includes a set of ten consecutive rules to link a given entity mention (exact match with a concept in the target KB, abbreviation expansion, replacement with synonyms, among others). Graph-based models are global models that build a graph usually including the entity mentions and respective KB candidate concepts and then score each candidate according to their role in the graph (for example, candidates associated with nodes with higher degree receive higher score) [18]. In terms of machine learning, there is a focus on models based on neural networks and, more recently, on large pre-trained language models based on the Transformers architecture. Models like BioBERT [19], PubMedBERT [20], SciBERT [21] are pre-trained on large biomedical corpora through unsupervised tasks and then fine-tuned for specific tasks, including NEL. Examples of these deep learning approaches include Ji et al. [22], BioSyn [6], BERN2 [23] (partially based on BioSyn), or DILBERT [14]. These models achieved state-of-the-art performance in several NEL benchmarks. These categories are not strict, since there are approaches that combine aspects of different categories. For instance, BioSyn [6] applies rules for abbreviation expansion, composite mention resolution, lowercasing and punctuation removal before generating the representation for the entity mentions with a deep learning model.

[2] https://www.nlm.nih.gov/bsd/medline_pubmed_production_stats.html.

[3] https://github.com/lasigeBioTM/NILINKER.

2.2. Predicting, clustering and typing NIL entities

Some recent studies proposed NEL models that include modules for the prediction of NIL entities among the entire set of entity mentions (also called out-of-KB prediction). Dredze et al. [3] proposed a model to link entities of several types (organizations, geopolitical and person entities) to entries of a KB derived from Wikipedia, which also included a module to predict NIL entities. The model applied the Ranking SVM algorithm to rank the candidates for each mention according to a set of features (based on lexical, Wikipedia, popularity and document properties) and then adapted the SVM ranker to predict which entities were NIL according to predefined features. The ranker considered an entity as NIL if, for example, there was low string similarity between the entity and the respective KB candidates and if the KB candidates were not associated with any highly-ranked Wikitology pages (an ontology derived from the Wikipedia category system). DILBERT [14] integrates a module to predict out-of-KB entity mentions in clinical trials by applying three strategies based on the distance of false-positive instances that differ on the threshold value: threshold value equivalent to the minimum distance of false positives, threshold value equivalent to the maximum distance and threshold value equivalent to the average of the two previous values.

The Knowledge Base Population track at the "Text Analysis Conference" (TAC) 2011 [2] introduced the requirement to cluster NIL entities (or "NIL queries") after the NEL step. We et al. [24] divided the approaches to NIL entity clustering into three categories: string matching, hierarchical agglomerative clustering and graph-based. String matching based methods calculate a similarity measure between the surface forms of the predicted NIL entities and then cluster the entities sharing high similarity. The hierarchical agglomerative clustering algorithm assigns the NIL entities to random clusters and iterates until the calculated distances between clusters are lower than a predefined threshold. Graph-based methods attempt to factor in the semantic relations between NIL entities. These methods usually create a semantic graph of NIL entities and then apply similarity measures or the hierarchical agglomerative clustering algorithm to create the clusters. More recently, Blissett et al. [10] proposed a recurrent neural network (RNN) architecture to perform cross-lingual NIL entity clustering.

Another approach consists of classifying unlinkable entities according to pre-defined types, which is designated by typing. Lin et al. [13] focus on the unlinkable noun phrase problem: if a noun phrase cannot be linked to any Wikipedia article or entity, the goal is first to determine if it is an entity and, if so, then classify it according to 1339 Freebase semantic types. The authors used noun phrases extracted from the ClueWeb09Web corpus that are involved in relations described in the text, linked the noun phrases to Wikipedia, obtained a dataset with linked and unlinkable entities, and developed a classifier to distinguish between the two types (entities, non-entities) based on the usage patterns across time. Then they propagated the semantic types of the linked entities to the unlinkable entities by developing an algorithm that leverages, among other features, the relations described in text: the main idea is that two entities, one linkable and the other unlinkable, that appear in the same type of relations must also share the respective semantic types.

The task of emerging entity recognition or detection is a closely related one. The third edition of the *Workshop on Noisy User-generated Text* (WNUT2017) focused precisely on this task [25]. The evaluation dataset consisted of documents including mainly emergent or rare entities belonging to several types, such as person, location, corporation, product, creative-work and group.

There are differences between our work and the models previously described. They focus exclusively on NIL entities present in general text, whereas our focus was to develop an approach to deal with NIL entities in the biomedical domain. Biomedical KBs have fewer available features compared to Wikipedia, thus specific approaches are required. For instance, the approach proposed by Lin et al. [13] explores Wikipedia-specific features, such as Freebase semantic types associated with an article, and leverages entities that are linked to Wikipedia to propagate their semantic types to unlinkable entities through relations described in text, whilst our approach does the opposite, i.e, uses NIL entity linking to improve NEL. Farber et al. [12] also focus on Wikipedia entities, but only on emergent entities, which are mostly of the type person. None directly determines the impact that the task of linking NIL entities can have on the broader NEL task. So, our modelling of the problem is different, as well as training and evaluation methodologies, as it will be further described.

2.3. Attention models

The review by Galassi et al. [26] provides a comprehensive overview of attention mechanisms used in neural networks in the context of natural language processing and define a "core attention model", which contains elements that, according to the authors, are present in most attention architectures. In essence, an attention function maps an input sequence of keys (such as word or character embeddings) to a distribution of attention weights. A compatibility function returns the energy scores calculated based on the relevance of the keys with respect to an input query, and then the distribution function computes the attention weights for the keys according to the energy scores.

The seminal paper by Vaswan et al. [27] introduced the Transformer architecture, which represented an improvement over state-of-the-art approaches in language modelling, until then based on recurrent neural networks. Transformer-based models are able to compute representations for input and output data using a self-attention mechanism. A recent trend in natural language processing consists of pre-training language representation models on large corpora and then fine-tuning them to specific downstream tasks. It is the case of BERT ("Bidirectional Encoder Representations from Transformers") [28], which, as the name suggests, leverages a multi-layer bidirectional Transformer encoder and then it performs two unsupervised tasks (Masked Language Model and Next Sentence Prediction) on large corpora as training objectives. Other BERT-based models have also been trained on domain-specific corpora [19,21,29]. The self-attention mechanism in BERT allows the fine-tuning of the pre-trained model to several downstream tasks, including NEL [22,30,31]. Besides BERT-related models, other attention-based models for NEL have been proposed [32].

For the present work, the "Semantic compositionality with Mutual Sememe Attention" (SCMSA) model proposed by Qi et al. for modelling semantic compositionality of multi-word expressions [33] is of great relevance. The principle of Semantic Compositionality asserts that the meaning of a syntactically complex expression can be expressed as a function of the meaning of its syntactic parts [34]. Qi et al. [33] retrieved sememes, indivisible semantic units of meaning in human languages, for each word from an external KB, HowNet [35]. This is a common-sense KB including Chinese and English words and a limited list of sememes, which can be combined in different ways to express the meaning of all words belonging to he KB vocabulary. The main goal of the authors in the referred article was to obtain meaningful vector representations for multi-word expressions, more concretely, expressions that contain two different words. To accomplish that, they proposed an attention-based that assigns different weights for the sememes associated with the constituent words of a multi-word expression in order to build an embedding representation that reflects the relative importance of each sememe. This reasoning was the motivation to build NILINKER, as will be further described.

3. Methodology

3.1. Problem definition

The problem of linking a NIL entity to a KB can be viewed as the selection of the most relevant KB concept from the respective candidates' list, assuming that a perfect concept to describe the entity does not exist in the KB. NIL entity linking is thus formulated in the following way: for each entity defined as NIL $NIL_e \in E$, being E the set of named entities mentioned in a given document, there is a candidates' list $CL(NIL_e) = \{c_1, \ldots, c_i\}$ retrieved from the KB, and the goal is to find the candidate $c_i \in CL$ that partially disambiguates NIL_e. In the present work we will model the problem of NIL entity linking as a multi-class classification, since each instance will be assigned the most relevant concept or class from a list including multiple classes, i.e., all the concepts belonging to the considered KB. However, in some cases, it may be helpful to obtain a set of several probable candidates instead of a single one, so the model needs to have the flexibility to return a variable number of KB candidates. One of the criteria to define a given entity as NIL is the fact that common methods for candidate retrieval from a KB return an empty list or below an acceptable confidence threshold, so it is necessary to find an alternative approach to build the candidates' list for NIL entities.

3.2. Candidate retrieval for NIL entities

In the context of the present work, NIL entities are the equivalent to the multi-word expressions considered in Qi et al. [33]. The authors resorted to HowNet to build the candidate sememes list for the words in the expression, however, this is not possible to replicate for NIL entities for two reasons. First, the goal is to find the relative importance of candidate KB concepts for NIL entities, and not of sememes for multi-word expressions, so HowNet is not useful. Second, HowNet only includes common English words, missing domain-specific words that appear in biomedical text, the focus of this work.

To overcome this, it is proposed the construction of a KB-derived word-concept dictionary: each word appearing in concept names and synonyms fields is associated with the respective concept ids. The approach includes the following steps:

1. Tokenization of the text present in *Name* and *Synonyms* fields for all concepts in the KB, exclusion of stop words which are not relevant (e.g. "and", "or, "ith"), and generation of a list with the remaining words.
2. Lemmatization and normalization of words present in the list. For example, the adjective "Parkinsonian" would be converted into the normalized lemma "parkinson".
3. Addition of the resulting words (keys) and the respective concept ids (values) where they appear to the KB word-concept dictionary.

For the case of a NIL entity constituted by two words, w_1 and w_2, the respective candidates' list is expressed by:

$$CL(NIL_e) = CL(w_1) + CL(w_2) \tag{1}$$

where $CL(w_1) = \{c^1_{w_1}, \ldots, c^i_{w_1}\}$ and $CL(w_2) = \{c^1_{w_2}, \ldots, c^i_{w_2}\}$.

An input NIL entity is tokenized and, for each word of its words, it is retrieved the respective candidates' list from the word-concept dictionary. Then, it is necessary to generate vectorized representations for both words and candidates to allow their input to the attention-based disambiguation model.

3.3. Representation of words and KB concepts with embeddings

3.3.1. Word embeddings

Word embeddings are dense, low-dimensional, distributed representations of words in a vector space, in which similar words are grouped. The Word2vec [36] method is based on Skip-gram, which is an unsupervised technique to learn embeddings from large amounts of unstructured text. Word2vec learns one vector per word, without considering the internal structure of words, which may decrease the quality of the representations in domains where it is common the existence of out-of-vocabulary and rare words, such as the biomedical one. BioWordVec [37] attempts to overcome these limitations, by using a subword embedding model to learn word representations from Medical Subject Headings (MeSH)[4] terms and biomedical literature text. In the present work, we used pre-trained Word2vec and BioWord2Vec embeddings through Gensim[5] Python library. Considering an example of a NIL entity with two different words, the embeddings relative to word 1 and to word 2 are denoted as $\mathbf{w1}, \mathbf{w2} \in \mathbb{R}^d$, being d the dimension of embeddings. In cases where the NIL entity has only one constituent word, it is assumed that the NIL entity has two repeated words, i.e., the constituent word is simultaneously word 1 and word 2, and the procedure is the same. If the entity has more than 2 words, the models truncates it and only considers the first two words.

3.3.2. Concept embeddings

node2vec [38] is an algorithm to build feature vector representations for the nodes and edges of a network. A KB is a network, where its concepts are the nodes and the relations between nodes are the edges. We used the code provided by the authors[6] to build embeddings for the concepts belonging to several KBs. Relations between concepts were in the input file with the format $(child\_concept) \rightarrow (parent\_concept)$, one relation per line. The output file contains a 200-dimensioned vector for each KB concept. The parameters values are: 'dimensions' = 200 (128), 'directed' ('undirected'), default values for the rest of the parameters. Considering the same example of a NIL entity with two different words, the set of embeddings relative to the KB candidates associated with word 1 are defined as $w'_1 = \{\mathbf{c^1_{w_1}}, \ldots, \mathbf{c^i_{w_1}}\}$, with $c^i_{w_1} \in CL(w_1)$ and $\mathbf{c^i_{w_1}} \in R^d$. Conversely, the set of embeddings relative to the KB candidates associated with word 2 are defined as $w'_2 = \{\mathbf{c^1_{w_2}}, \ldots, \mathbf{c^i_{w_2}}\}$, with $c^i_{w_2} \in CL(w_2)$ and $\mathbf{c^i_{w_2}} \in R^d$.

3.4. NILINKER: disambiguation of NIL entities

Qi et al. [33] proposed an approach to model Semantic Compositionality using sememes and have applied their model in a downstream sememe prediction task, which, as its name suggests, consists in the discovery of the most relevant sememes to represent the meaning of a multi-word expression. Similarly, the idea is that NIL entities can be viewed as multi-word expressions, and their meaning can be expressed through the most relevant KB concepts. In the context of this work, the KB-derived word-concept dictionary is the equivalent of HowNet.

Following the definition by Galassi et al. [26], the attention mechanism learns the relevance of the different parts of the input, so the development of an attention-based model contributes to the aforementioned goal. The core attention model defined by Galassi et al. [26] includes the following elements: *K*, keys or vectors which are the target of the attention weights, *e*, the energy scores, *f*, the compatibility function, and *g*, the distribution function, being the output a set of attention weights.

Considering the case of a NIL entity including word 1 and word 2, the attention weight for a KB candidate associated with word 1 is obtained through the following distribution function:

$$a_{2,i} = \frac{exp(\mathbf{c^i_{w_1}} \cdot \mathbf{e_1})}{\sum_{\mathbf{c_j} \in w'_2} exp(\mathbf{c^j_{w_2}} \cdot \mathbf{e_1})} \tag{2}$$

where $\mathbf{c^i_{w_1}}$ is the vector of candidate i for word 1, $\mathbf{c^j_{w_2}}$ is the vector of candidate j for word 2, and $\mathbf{e_1}$ is the vector of energy scores for word 1. This is obtained through the compatibility function:

$$e_1 = tanh(\mathbf{W_a w_1} + \mathbf{b_a}) \tag{3}$$

[4] https://meshb.nlm.nih.gov/search.

[5] https://radimrehurek.com/gensim/.

[6] https://github.com/aditya-grover/node2vec.

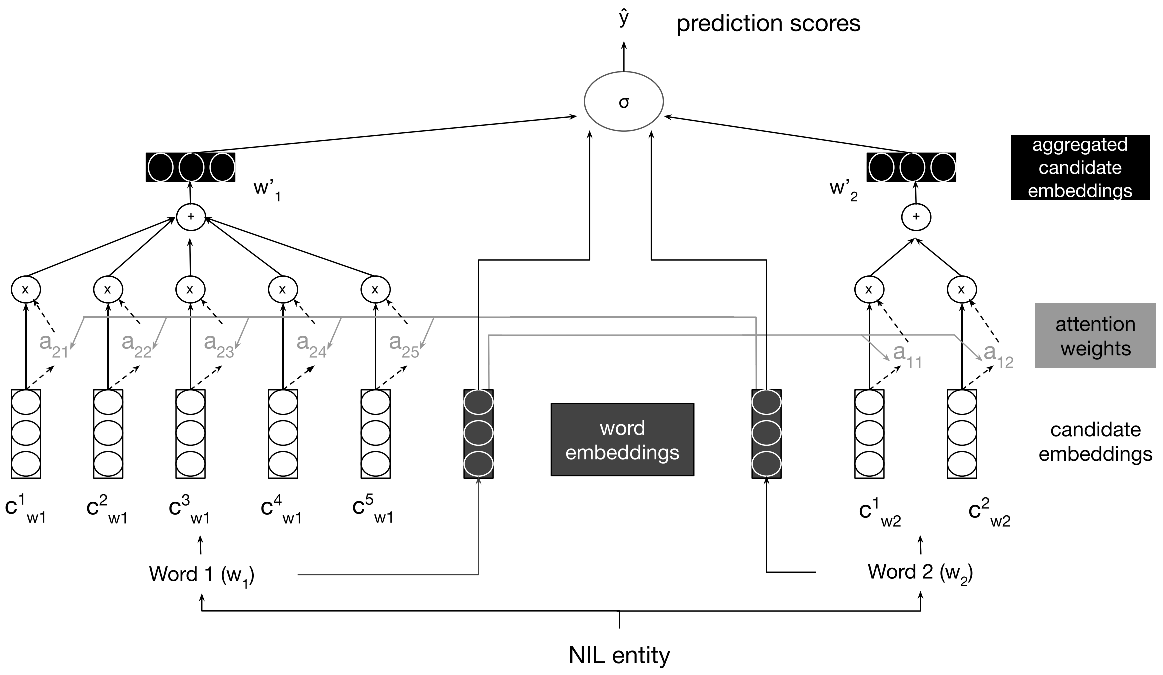


Fig. 1. Architecture of the proposed NILINKER model to find the most relevant KB concept for a NIL entity.

where $\mathbf{W_a} \in R^{d \times d}$ is the weight matrix, $\mathbf{w_1} \in R^d$ is the vector of word 1, and $\mathbf{b_a} \in R^d$ is the bias vector. Conversely, it is possible to calculate $a_{1,j}$ for each $j \in CL(w_2)$, as well e_2.

The aggregated embeddings for word 1 candidates are denoted by w_1' and the aggregated embeddings for word 2 candidates are denoted by w_2'. In order to classify the KB candidates, it is applied a single-layer perceptron:

$$\hat{y}_p = \sigma(\mathbf{W}_{KB} \cdot \mathbf{p}) \tag{4}$$

In which σ is the softmax function, $\mathbf{W}_{KB} = w_1' + w_2'$, $\mathbf{p} = w_1 + w_2$, and $\hat{y}_p$ is the probability distribution for the KB concepts. The higher the score, the higher the probability that the respective candidate is the correct disambiguation, consequently, the highest scoring candidate disambiguates the NIL entity. The full overview of the proposed model is available in Fig. 1. The parameter `top_k` determines the number of top-k candidates according to their probabilities to be returned for a given input entity.

To calculate the training loss we use Softmax Cross Entropy Loss H:

$$H(y, \hat{y}_p) = -\sum_x y(x) log \hat{y}_p(x) \tag{5}$$

Where y is the true distribution of labels for given instance x (the true label corresponds to “1”, the remaining to “0”), $\hat{y}_p$ is the estimated probability distribution outputted by the model. Therefore, the equation returns the distance between expected and model prediction.

Additionally, we modified the loss function by adding class weighting to further reduce the impact of class imbalance using the scikit-learn implementation `compute_class_weight` [39]. The vector including the weight for each label is obtained by:

$$ClassWeights = \frac{n_{samples}}{n_{classes} * B} \tag{6}$$

where $n_{samples}$ corresponds to the total number of values present in y, $n_{classes}$ corresponds to the total number of classes/labels present in y, B corresponds to the set including the frequency values associated with each label present in y (e.g. the index 0 corresponds to the label 0 and the value associated with the index 0 corresponds to the number of times that the label 0 is present in y). This means that frequent classes/labels in the dataset are assigned smaller weights, consequently have less impact on the loss value, whereas rare classes are assigned higher weights and have more impact on the loss.

One of the challenges hindering the development and implementation of the proposed model was the lack of training datasets, which was solved through the generation of a new dataset, as it is further described.

3.5. Data

Data used in the context of the present work consisted of files associated with commonly used biomedical KBs and of annotated datasets belonging to the same domains.

3.5.1. KB files

The files related with the following KBs were used: MEDIC vocabulary (version: 2022-04-04) [40], CTD-Anatomy (CTD-Anat) (version: 2022-04-04) [40], CTD-Chemicals (CTD-Chem) vocabulary (version: 2022-04-04) [40], ChEBI ontology (version: 13-Oct-2021 14:23) [41], Gene Ontology (GO) (2021-09-01 release) [42], and Human Phenotype Ontology (HP) (2021-10-10 release) [43].

3.5.2. Datasets

We have used the gold standard NEL datasets that are described in Table 1. Based on the work by [44], for evaluation we used the original test set and, to remove the bias, a refined test set that excludes entity mentions that are present in the training and development sets. The source of the datasets BC5CDR-Disease, BC5CDR-Chemical and NCBI Disease was the repository https://github.com/insilicomedicine/Fair-Evaluation-BERT. For the remaining datasets we have generated the refined test set excepting GSC+: we used the whole dataset since it does not have separated sets.

In order to train the NILINKER model, it is necessary a specific dataset including NIL entities associated with KB concepts. The datasets used to build the EvaNIL dataset (see Section 3.5.3) were: PubMedDS [51] (silver standard, 13,197,430 PubMed abstracts with 57,943,354 biomedical annotations, MeSH, including MEDIC, CTD-Chemicals and CTD-Anatomy concepts), CRAFT corpus [52] (gold standard, 97

Table 1
Statistics for the evaluation NEL datasets. Docs: Documents, Annots: Annotations, 1 word: Entities with 1 word, 2 words: Entities with 2 words, 3 words: Entities with 3 words, 4 words: Entities with 4 words, 5 words: Entities with 5 words.

Dataset	Set	Docs	Annots	1 word	2 words	3 words	4 words	5 words
BC5CDR-Disease [45]	Test	500	4287	2226 (51.9244%)	1264 (29.4845%)	619 (14.4390%)	107 (2.4959%)	71 (1.6562%)
	Test-Refined	322	692	171 (24.7110%)	288 (41.6185%)	134 (19.3642%)	56 (8.0925%)	43 (6.2139%)
BC5CDR-Chemical [45]	Test	500	5015	4144 (82.6321%)	668 (13.3200%)	93 (1.8544%)	75 (1.4955%)	35 (0.6979%)
	Test-Refined	230	447	277 (61.9687%)	111 (24.8322%)	26 (5.8166%)	15 (3.3557%)	18 (4.0268%)
NCBI Disease [46]	Test	100	960	239 (24.8958%)	378 (39.3750%)	198 (20.6250%)	88 (9.1667%)	57 (5.9375%)
	Test-Refined	71	208	25 (12.0192%)	76 (36.5385%)	45 (21.6346%)	32 (15.3846%)	30 (14.4231%)
GSC+ [47,48]	All	228	2773	1079 (38.9109%)	1020 (36.7832%)	403 (14.5330%)	131 (4.7241%)	140 (5.0487%)
CHR [49]	Test	3611	30,556	27,235 (89.1314%)	3236 (10.5904%)	50 (0.1636%)	31 (0.1015%)	4 (0.0131%)
	Test-Refined	233	526	444 (84.4106%)	73 (13.8783%)	7 (1.3308%)	2 (0.3802%)	0 (0%)
PHAEDRA [50]	Test	115	1633	1527 (93.5089%)	85 (5.2051%)	18 (1.1023%)	2 (0.1225%)	1 (0.0612%)
	Test-Refined	37	177	150 (84.7458%)	16 (9.0395%)	9 (5.0847%)	1 (0.5650%)	1 (0.5650%)

Table 2
Statistics for the EvaNIL dataset. Docs: Documents, Annots: Annotations, 1 word: Entities with 1 word, 2 words: Entities with 2 words, 3 words: Entities with 3 words, 4 words: Entities with 4 words, 5 words: Entities with 5 words.

Partition	Subset	Docs	Annots	1 word	2 words	3 words	4 words	5 words
MEDIC	Train	294,926	379,041	242,978 (64.1034%)	115,961 (30.5933%)	18,176 (4.7953%)	1663 (0.4387%)	263 (0.0694%)
	Dev	62,000	80,143	1347 (64.0692%)	24,292 (30.3108%)	4122 (5.1433%)	330 (0.4118%)	52 (0.0649%)
	Test	62,000	80,050	51,734 (64.6271%)	24,175 (30.1999%)	3788 (4.7320%)	311 (0.3885%)	42 (0.0525%)
	Test-Refined	147	151	54 (35.7616%)	77 (50.9934%)	14 (9.2715%)	4 (2.6490%)	2 (1.3245%)
CTD-Chem	Train	10,749	15,755	12,701 (80.6157%)	2560 (16.2488%)	420 (2.6658%)	66 (0.4189%)	8 (0.0508%)
	Dev	4000	5655	4613 (81.5738%)	834 (14.7480%)	165 (2.9178%)	40 (0.7073%)	3 (0.0531%)
	Test	4000	5922	4800 (81.0537%)	1002 (16.9200%)	100 (1.6886%)	16 (0.2702%)	4 (0.0675%)
	Test-Refined	329	356	233 (65.4494%)	107 (30.0562%)	12 (3.3708%)	3 (0.8427%)	1 (0.2809%)
CTD-Anat	Train	282,153	343,262	223,452 (65.0966%)	109,917 (32.0213%)	9346 2.7227%)	113 (0.0329%)	434 (0.1264%)
	Dev	62,000	75,578	49,437 (65.4119%)	23,906 (31.6309%)	2130 (2.8183%)	17 (0.0225%)	88 (0.1164%)
	Test	62,000	75,925	49,871 (65.6846%)	24,002 (31.6128%)	1937 (2.5512%)	29 (0.0382%)	86 (0.1133%)
	Test-Refined	78	79	19 (24.0506%)	55 (69.6203%)	4 (5.0633%)	1 (1.2658%)	0 (0%)
ChEBI	Train	69	1080	904 (83.7037%)	151 (13.9815%)	21 (1.9444%)	2 (0.1852%)	2 (0.1852%)
	Dev	14	196	157 (80.1020%)	24 (12.2449%)	8 (4.0816%)	7 (3.5714%)	0 (0%)
	Test	14	204	162 (79.4118%)	37 (18.1373%)	5 (2.4510%)	0 (0%)	0 (0%)
	Test-Refined	13	61	38 (62.2951%)	19 (31.1475%)	4 (6.5574%)	0 (0%)	0 (0%)
HP	Train	1501	4052	2395 (59.1066%)	1144 (28.2330%)	387 (9.5508%)	101 (2.4926%)	25 (0.6170%)
	Dev	321	850	483 (56.8235%)	248 (29.1765%)	104 (12.2353%)	10 (1.1765%)	5 (0.5882%)
	Test	321	847	492 (58.0874%)	241 (28.4534%)	93 (10.9799%)	18 (2.1251%)	3 (0.3542%)
	Test-Refined	160	296	101 (34.1216%)	133 (44.9324%)	50 (16.8919%)	10 (3.3784%)	2 (0.6757%)
GO-BP	Train	69	2413	1400 (58.0191%)	525 (21.7571%)	206 (8.5371%)	176 (7.2938%)	106 (4.3929%)
	Dev	14	377	206 (54.6419%)	87 (23.0769%)	44 (11.6711%)	29 (7.6923%)	11 (2.9178%)
	Test	14	426	262 (61.5023%)	100 (23.4742%)	29 (6.8075%)	19 (4.4601%)	16 (3.7559%)
	Test-Refined	14	139	38 (27.3381%)	48 (34.5324%)	25 (17.9856%)	13 (9.3525%)	15 (10.7914%)

PubMed articles with chemical and biological process entities, including ChEBI and GO concepts), and MedMentions [53] (gold standard, 4392 PubMed abstracts with biomedical annotations, UMLS concept with cross-links to HP concepts).

3.5.3. Evanil: large silver standard for NIL entity linking evaluation

As there are no datasets for evaluation of the NIL entity linking task and the development of gold standard corpora from scratch is both time and effort-intensive, we generated a large dataset from existing annotated datasets for NEL evaluation. These datasets contain annotations with entities and the respective KB concepts that disambiguate them, with format (entity, concept ID), however, they usually do not include NIL annotations, which is not representative of a real application scenario where there are always unlinkable entities.

To leverage the “linkable” entities already present in the annotations of those corpora, we assumed that the associated concept is absent from the respective KB and, consequently, that the entity then should be linked to the direct ancestor of the annotated concept. This way, linkable entities were transformed into NIL entities. Applying this strategy, annotations present in several corpora (CRAFT corpus, MedMentions, and PubMedDS) were converted into the format (entity, direct ancestor concept ID). UMLS ids present in MedMentions corpus were mapped to HP ids since this is a freely available ontology.

The final dataset includes six different sets, which aggregate annotations containing concept ids from the following KBs: GO, ChEBI, MEDIC, CTD-Chem, HP, CTD-Anat. To reduce the bias, we also generated a test-refined set for each partition that excludes entity mentions appearing in the respective training and development sets. To reduce the class imbalance in the final dataset, repeated annotations in the same document were eliminated. The final dataset is available for download.[7] and its statistics are shown in Table 2.

3.6. Experimental setup

The experimental setup was divided in two phases: evaluation on the EvaNIL dataset and evaluation of the impact in the NEL task. An overview of the experimental setup is shown in Fig. 2.

3.6.1. Evaluation on the evanil dataset

The following models were evaluated on each partition of the EvaNIL dataset, more concretely on the test and on the test-refined sets:

[7] https://zenodo.org/record/5849231.

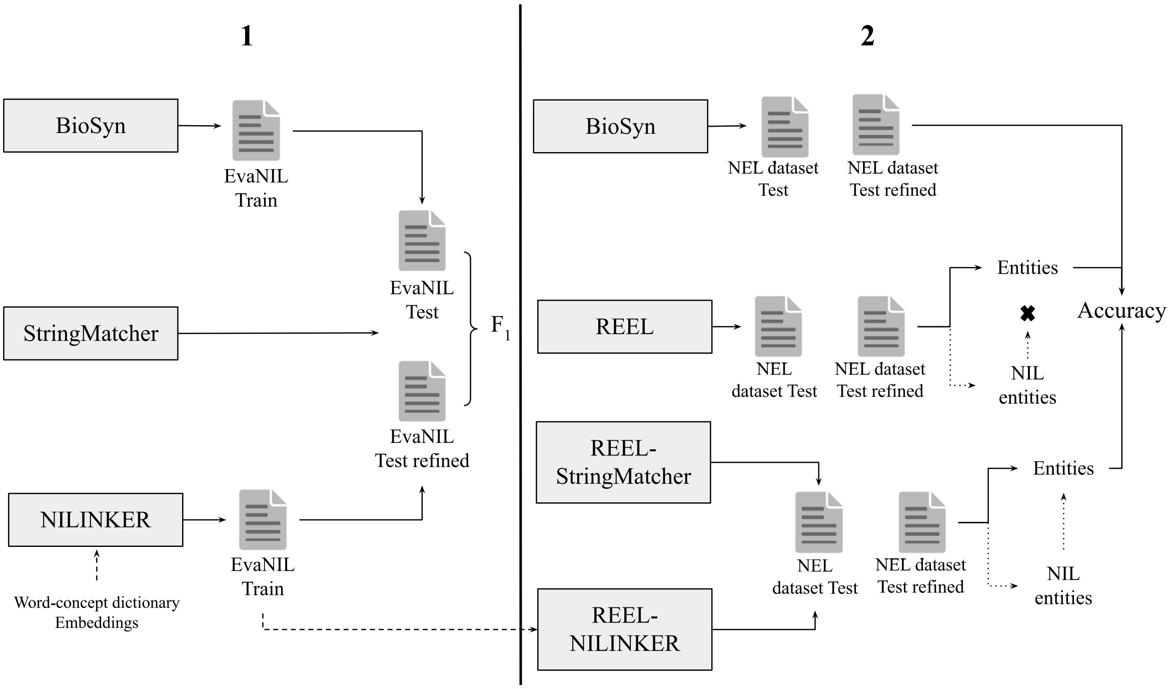


Fig. 2. Experimental setup: (1) evaluation on the EvaNIL dataset, (2) determination of the impact in the NEL task of integrating the trained NILINKER model with a NEL model.

- StringMatcher: a simple baseline that associates each NIL entity with the most similar concept in the respective target KB according to lexical distance.
- BioSyn: model with state-of-the-art performance in the NEL task [6] it learns representations for biomedical entity mentions using synonym marginalization and iterative candidate retrieval. We trained the original BioBERT-based implementation using the same parameters used by the authors.
- NILINKER

The models BioSyn and NILINKER were trained and validated on the training and development sets and evaluated on the test and test-refined sets of the respective EvaNIL partition. For the model BioSyn, we used the default values described by its authors [6]: number of top candidates k of 20, mini-batch size of 16, learning rate of $1e^{-5}$, dense ratio of 0.5, training epochs equal to 10.

The model StringMatcher is only evaluated on the test and test-refined sets. In terms of evaluation metrics, True positives (tp) refer to the number of entities correctly linked, false positives (fp) to the number of entities wrongly linked and false negatives (fn) to the number of entities that the model does not link. As the EvaNIL dataset is class imbalanced, the performance of each model was evaluated through $precision$, $recall$ and the micro-averaged F_1 $score$.

We tested for statistical significance between the baseline model (StringMatcher) and the alternative models (either NILINKER or BioSyn) using the paired Wilcoxon Signed-Rank test, following the recommendation of Dror and co-workers [54], and, in case of detected statistical significance effect size, we also determined its magnitude through the Wilcoxon effect size (r) [55]. Concerning the paired Wilcoxon Signed-Rank test, for each pair of models being tested, we considered the following null (H_0) and alternative (H_1) statistical hypotheses:

- H_0: There is no difference between the performance of the baseline model and the performance of the alternative model.
- H_1: The performance of the baseline approach is different than the performance of the alternative model.

A difference is statistically significant when the p-value p is smaller than the significance level α ($p < \alpha$), being $\alpha = 0.05$ in the context of this work. We developed a R script to perform the statistical analysis based on the functions `wilcox_test` and `wilcox_effsize` of the `rstatix` package.[8]

3.6.2. Impact in the NEL task

The goal of this phase was to evaluate the trained NILINKER models on downstream tasks. Given the impact that NIL entities have on the performance of NEL models, we studied the effect of integrating the NILINKER model with a NEL model. We used a model previously developed by our group, REEL [18], which links biomedical entities (chemicals and diseases), to several biomedical KBs, more concretely, ChEBI, MEDIC, and CTD-Chemicals.

REEL includes two components: candidate generation (retrieval from the KB of the most relevant candidates for a given entity) and candidate disambiguation (selection of the most relevant candidate). The candidates for the entities present in a given document form a disambiguation graph: each candidate is a node, the edges between the candidates are added according to existing relations in the structure of the KB and, if available, to relations extracted directly from the respective text. The candidate disambiguation component is based on the Personalized PageRank (PPR) algorithm and on Information Content. An entity was considered to be NIL (a) if the respective candidates' list was empty, (b) if the correct candidate for the entity is not present in the retrieved candidate list or (c) if there is no KB id associated with it in the respective corpus. As REEL is a graph-based approach that is based on the maximization of the coherence between the nodes/candidates we consider that it is useful to study the impact of adding nodes by the NIL entity linking models. This experiment evaluated the following models:

- BioSyn: model with state-of-the-art performance in the NEL task [6]. We used the SapBERT-based models already trained on

[8] https://cran.r-project.org/web/packages/rstatix/index.html.

the training and development sets of the BC5CDR-Disease,[9] the BC5CDR-Chemical[10] and the NCBI Disease[11] datasets.

- REEL: baseline approach, ignores NIL entities.
- REEL-StringMatcher: StringMatcher associates each NIL entity with the most similar concept in the respective target KB according to lexical distance.
- REEL-NILINKER: for each NIL entity, NILINKER returns the `top-k` most probable KB concepts. The parameter `top-k` is optimized for each evaluation dataset.

We evaluated the models in gold standard datasets focusing on disease entities, such as BC5CDR-Disease, GSC+ and NCBI Disease and on chemical entities, such as BC5CDR-Chemical, CHR, PHAEDRA. The model BioSyn was evaluated on the datasets BC5CDR-Disease, BC5CDR-Chemical and NCBI Disease since these are popular datasets, which allows the comparison with other SOTA approaches, and because they include two of the most important types of biomedical entities, chemicals and disease. The performance of each model is represented by the returned accuracy at 1 ($acc@1$) in each dataset, a common metric used in the NEL tasks that allows the comparison with state-of-the-art models:

$$acc@1 = \frac{tp}{tp + fp} \tag{7}$$

Corresponding *tp* to true positives (entities correctly identified by the model) and *fp* to false positives (entities incorrectly identified by the model). We performed a statistical analysis of the performance of the three models in the evaluation datasets. We applied the same testing for statistical significance, the Wilcoxon Signed-Rank test. The baseline approach was the REEL model and the alternative approach was either the REEL-StringMatcher, REEL-NILINKER or the BioSyn models.

3.7. Implementation

The NILINKER model was implemented using Tensorflow 2.5.1 and Python 3.8.6. The two phases of the experimental setup were executed on a Tesla M10 GPU. For each partition of the EvaNIL dataset, we performed a manual hyperparameter optimization before training, focusing on the parameters `learning_rate`, `optimizer`, `train_batch_size`, and `test_batch_size`. The training and evaluation times for the final versions of the NILINKER model were approximately 0.2, 1.4, 5.2, 10, 1.2 and 1.7 h for the partitions HP, ChEBI, MEDIC, CTD-CHEM, GO-BP and CTD-ANAT, respectively.

4. Results and discussion

4.1. Results

The results relative to the evaluation on the EvaNIL dataset are shown in Table 3. On the original test sets, the best result was achieved by NILINKER in the MEDIC partition (0.9401 F_1 *score*) and the worst result by the BioSyn model in the ChEBI partition (0.0194 F_1 *score*) . On the test-refined sets, the best result was achieved by NILINKER in the MEDIC partition (0.7149 F_1 *score*) and the worst result by the BioSyn model in the GO-BP partition (0.0284 F_1 *score*).

The results relative to the second evaluation, the integration of NILINKER with the REEL model, are shown in Table 4. The best model in all datasets was BioSyn. As expected, the results for the test-refined sets were worse than the results for the original Test sets. In three of the datasets (BC5CDR-Disease, NCBI-Disease and PHAEDTA) the performance of the REEL-NILINKER model was higher compared with the baseline model REEL, although the difference in the performance was only statistically significant in two of those datasets (BC5CDR-Disease and NCBI-Disease). The optimal values for the parameter `top_k` in the datasets BC5CDR-Disease, GSC+, NCBI Disease, BC5CDR-Chemical, CHR and PHAEDRA are 2, 6, 2, 5, 20 and 5, respectively.

4.2. Discussion

The results of the evaluation on the EvaNIL dataset show that the NILINKER model still have room for improvement, in particular in the linking of ChEBI, CTD-Chem and HP entities. The number of documents used for training the models NILINKER-CTD-Chem (10,749), NILINKER-HP model (1501), NILINKER-ChEBI (69) and NILINKER-GO-BP (69) is substantially lower than the documents used for training the NILINKER-MEDIC model (294,926) and the NILINKER-CTD-ANAT model (282,153). So, we believe the relative lower performance of the former models is due to the smaller size of the training datasets. Since CTD-Chem and ChEBI are large KBs, we had to reduce the size of the training datasets due to memory constraints. So, expanding the training datasets while solving the memory issues may improve the performance of these models, and, consequently, increase their impact on downstream tasks.

BioSyn obtained a low performance in the NIL entity linking evaluation executed on the EvaNIL dataset which was expected since BioSyn was originally developed for the NEL task, whereas the NIL entity linking task has a distinct goal: associate a given NIL entity not with the most appropriate KB concept but with its direct ancestor.

Counterintuitively, the performance of the BioSyn and of the StringMatcher models was lower in the test-refined sets of the EvaNIL dataset than the original test sets. This is partially related with the imbalance of the dataset. In the test set of the EvaNIL-MEDIC partition, the 10 most common annotations and respective frequencies are: 'hypertension', (2576), 'stroke' (1821), 'inflammation' (1820), 'schizophrenia' (1613), 'pain' (1376), 'heart failure' (1258), 'body weight' (1089), 'recurrence' (1037), 'insulin resistance' (1018), 'multiple sclerosis' (920). By its hand, in the test-refined set of the same partition, the 10 most common annotations and respective frequencies are: 'LYMPHEDEMA' (4) 'Munchausen Syndrome' (3) 'CONSTIPATION' (2), 'anticholinergic syndrome' (2), 'SARCOPENIA' (2), 'Granulomatous Mastitis' (2), 'HYPEREOSINOPHILIC SYNDROME' (2), 'calcium metabolism disorders' (2), 'Neglected Diseases' (2), 'No-Reflow Phenomenon' (2). So, the class imbalance in the test-refined set is substantially lower than the original test set. The performance of StringMatcher and BioSyn in the original test set is not as high as expected, but in the test-refined sets is comparatively higher due to the lower repetition of annotations. For example, BioSyn wrongly predicts the gold label of the annotation 'hypertension', the most common annotation in the test set: the top prediction is the label 'MESH:D000075222', but the gold label in the dataset is 'MESH:D014652'. This corresponds to 2576 wrong predictions, whereas in the test-refined set this annotation is not even present. This means that, in the test-refined set, the weight of the wrong predictions is lower than the weight of the wrong predictions in test set.

Although the difference in the performance of the NILINKER model compared with the baseline REEL is small (effect sizes of 0.143 in the BC5CDR-Disease dataset and 0.0913 in the NCBI Disease dataset), the difference is consistent since it was observed in evaluation datasets with different characteristics. We did our best to present an extensive evaluation as possible, so the results are probably generalizable to other entities in the biomedical domain.

We performed an extensive analysis to find the factors behind the impact of NILINKER in the NEL task. The analysis focused on the following questions:

1. Is the impact of NILINKER in the NEL task associated with the percentage of NIL entities in the evaluation dataset?
2. Is the impact of NILINKER in the NEL task associated with its performance determined on the respective partition of the EvaNIL dataset (determined in the test set)?

[9] https://huggingface.co/dmis-lab/biosyn-sapbert-bc5cdr-disease.

[10] https://huggingface.co/dmis-lab/biosyn-sapbert-bc5cdr-chemical.

[11] https://huggingface.co/dmis-lab/biosyn-sapbert-ncbi-disease.

Table 3
Evaluation results on each partition of the EvaNIL dataset for the NIL entity linking task, along with the significance levels obtained from the paired Wilcoxon Signed-Rank test (comparison with the baseline performance) and the Wilcoxon effect size (between parenthesis) in the F_1 columns. In the NIL entity linking task, we assume that the gold label does not exist in the target KB so the goal is to associate the NIL entity with the direct ancestor of the original gold label. Highest value by metric for each subset (test and test-refined) is highlighted.

Partition	Model	Test			Test-refined		
		P	R	F1	P	R	F1
HP	StringMatcher	0.0295	**1.0000**	0.0573	0.0439	**1.0000**	0.0841
	BioSyn	0.0473	0.9524	0.0902*** (0.1330)	0.0578	0.8947	0.1086^{ns} (0.1160)
	NILINKER	**0.4274**	**1.0000**	**0.5988****** (0.6310)	**0.1351**	**1.0000**	**0.2381****** (0.3020)
ChEBI	StringMatcher	0.0343	**1.0000**	0.0664	0.0492	**1.0000**	0.0938
	BioSyn	0.0098	**1.0000**	0.0194* (0.1570)	0.0164	**1.0000**	0.0323^{ns} (0.1810)
	NILINKER	**0.3971**	**1.0000**	**0.5684****** (0.6020)	**0.0820**	**1.0000**	**0.1515**^{ns} (0.1810)
MEDIC	StringMatcher	0.0450	**1.0000**	0.0861	0.0530	**1.0000**	0.1006
	BioSyn	0.0533	0.8364	0.1002**** (0.0879)	0.0530	**1.0000**	0.1006^{ns} (0)
	NILINKER	**0.8869**	**1.0000**	**0.9401****** (0.9180)	**0.5563**	**1.0000**	**0.7149****** (0.7090)
CTD-Chem	StringMatcher	0.0336	**1.0000**	0.0650	0.0674	**1.0000**	0.1263
	BioSyn	0.0111	**1.0000**	0.0220^{ns} (0.0225)	0.0253	**1.0000**	0.0493*** (0.2050)
	NILINKER	**0.3242**	**1.0000**	**0.4897****** (0.5390)	**0.0955**	**1.0000**	**0.1744**** (0.1680)
GO-BP	StringMatcher	0.0117	**1.0000**	0.0232	0.0360	**1.0000**	0.0694
	BioSyn	0.0047	**1.0000**	0.00934^{ns} (0.0839)	0.0140	**1.0000**	0.0284^{ns} (0.1470)
	NILINKER	**0.4742**	**1.0000**	**0.6433****** (0.6800)	**0.1079**	**1.00000**	**0.1948**** (0.2680)
CTD-ANAT	StringMatcher	0.0323	**1.0000**	0.0626	0.0759	**1.0000**	0.1412
	BioSyn	0.0311	0.9692	0.0603**** (0.0348)	0.1039	0.8000	0.1839^{ns} (0.1590)
	NILINKER	**0.9156**	**1.0000**	**0.9560****** (0.9400)	**0.4937**	**1.0000**	**0.6610****** (0.6460)

P: Precision; R: Recall; F1: F1-score.

$^{*}p < 0.05$.

$^{**}p < 0.01$.

$^{***}p < 0.0001$.

$^{****}p \approx 0$.

Table 4
Impact in the NEL task of applying a NIL entity linking model: $acc@1$ for the models BioSyn, REEL, REEL-StringMatcher, REEL-NILINKER in six different evaluation datasets (Test and Test-refined sets, along with the significance levels obtained from the paired Wilcoxon Signed-Rank test (comparison with the baseline performance) and the Wilcoxon effect size (between parenthesis)

Dataset	Model	Test	Test-refined
BC5CDR-Disease	REEL	0.7462	0.5665
	REEL-StringMatcher	0.7464^{ns} (0.0153)	0.5751* (0.0931)
	REEL-NILINKER	0.7667**** (0.1430)	0.5882*** (0.1470)
	BioSyn	**0.9384****** (0.4380)	**0.7861****** (0.4690)
BC5CDR-Chemical	REEL	0.9378	0.8456
	REEL-StringMatcher	0.9386^{ns} (0.0282)	0.8479^{ns} (0.0473)
	REEL-NILINKER	0.9376^{ns} (0.0141)	0.8434^{ns} (0.0473)
	BioSyn	**0.9665*** (0.1250)	**0.8810*** (0.1250)
NCBI Disease	REEL	0.7250	0.4904
	REEL-StringMatcher	0.7396*** (0.1210)	0.4760^{ns} (0.1200)
	REEL-NILINKER	0.7333** (0.0913)	0.5096^{ns} (0.1390)
	BioSyn	**0.9219****** (0.4440)	**0.7692****** (0.5280)
GSC+	REEL	**0.6012**	–
	REEL-StringMatcher	0.5943**** (0.0828)	–
	REEL-NILINKER	0.6004^{ns} (0.0269)	–
CHR	REEL	0.6647	0.4106
	REEL-StringMatcher	**0.7280****** (0.2520)	0.4125^{ns} (0.0436)
	REEL-NILINKER	0.6647^{ns} (0.0057)	0.4106^{ns} (0)
PHAEDRA	REEL	0.7740	**0.1864**
	REEL-StringMatcher	0.7722^{ns} (0.0429)	0.1695^{ns} (0.1300)
	REEL-NILINKER	**0.7746**^{ns} (0.0247)	**0.1864**^{ns} (0)

$^{*}p < 0.05$.

$^{**}p < 0.01$.

$^{***}p < 0.0001$.

$^{****}p \approx 0$.

3. Does the presence of more or less general entities in the evaluation dataset influence NILINKER's impact in the NEL task? NILINKER was trained to link a given entity to more general entities in the target KBs, but if general entities were already present in a dataset, NILINKER would just link the entities to the root concept of a KB. Some entities were considered as NIL by the REEL model, but they are still associated with a KB identifier in the respective dataset, which generally does not correspond to the root concept identifier.
4. How exactly NILINKER impacts the NEL task? In each evaluation dataset, there are entities that REEL consider as NIL, even if they are associated with a given KB id. NILINKER retrieves the top-k candidates for those entities. Does the correct KB ID for the NIL entities is present in the retrieved candidates' list?

Table 5
Analysis of the results. Total entities and NILs count in each evaluation dataset (identified by the REEL model), as well the relative percentage of NILs to the number of unique entities present. F_1 refers to the $F_1 score$ value obtained by the NILINKER model on the respective EvaNIL partition. Δ refers to the difference in performance of the REEL-NILINKER model compared with the baseline model REEL in given dataset in terms of $acc@1$. Solution in list indicates the percentage of cases where REEL retrieved a non-empty candidates' list in given dataset.

Dataset	Target KB	Entities	NILs	% NILs	Avg ancestors	F_1	Δ	NIL model	Solution in list %
BC5CDR-Disease	MEDIC	4287	1194	27.8516%	3.5	0.9401	+0.0205	None NILINKER	72.1250 73.9678
GSC+	HP	2773	1005	36.2423%	5.0	0.5988	−0.0008	None NILINKER	63.7577 64.0462
NCBI Disease	MEDIC	960	314	32.7083%	3.7	0.9401	+0.0083	None NILINKER	67.2917 67.9167
BC5CDR-Chemical	CTD-CHEM	5381	1547	28.7493%	4.7	0.4897	−0.0002	None NILINKER	71.2507 71.2693
CHR	CHEBI	30,556	12,532	41.0132%	8.0	0.5684	+0.0000	None NILINKER	58.9868 58.9901
PHAEDRA	CTD-CHEM	1633	434	26.5769%	5.0	0.4897	+0.0006	None NILINKER	73.4231 73.4231

Table 5 includes the result of the analysis done to answer the questions above. The impact of the NILINKER model is not associated with the number of NIL entities in the evaluation dataset. However, the higher the $F_1 score$ obtained on the EvaNIL evaluation, the higher the increase of performance in the NEL task. The analysis also suggest that the impact of the NILINKER model is independent of the "specificity" (average number of ancestors) of the entities in the evaluation datasets.

With respect to the question 4, the results suggest that applying NILINKER increases the number of cases where the correct candidates are present in the candidates' lists of NIL entities. The datasets where the number of correct candidates increased more (BC5CDR-Disease and NCBI Disease) were also the datasets where the model REEL-NILINKER improved more the performance compared with the baseline REEL. One example showing the impact of the integrating NILINKER with the REEL model is shown in Fig. 3.

The impact of NILINKER in the NEL task is also indirect, i.e., the retrieved candidates are added to the disambiguation graph, this will include more candidates/nodes, but, more crucially, the graph will now include more edges between all the nodes/candidates. The number of edges in the graph is essential for the application of the PPR algorithm. The score of each node/candidate is mostly based on the number of edges it is involved in the graph. The retrieved candidates, even if none of them corresponds to the correct candidate for a NIL entity, will increase the semantic information stored in the graph, which by its turn helps in the disambiguation of the other linkable entities.

In the NEL task, the inclusion of a large number of candidates outputted by NILINKER for the NIL entities can increase the noise in the disambiguation graph, which leads to performance loss. The number of candidates proposed by NILINKER is specified by the parameter `top-k`, so it is crucial to find its optimal value for each dataset that NILINKER is being applied: a candidate list with too many candidates can increase noise, but a list with too few can also miss relevant semantic information.

If an entity has more than words, NILINKER truncates the entity and only considers the first two words, which leads to a loss of information of unknown impact. One way of improving the performance could pass by adapting the architecture of the model to accept any input entity size without resorting to truncation, thus preserving the input data. However, this adaptation would not be trivial, which puts it outside the scope of the present work, so we leave that task to future work.

For simplicity, we only included annotations with exact one direct ancestor in the EvaNIL dataset, which leaves out of the dataset a large part of the concepts of each KB. This relative low annotation diversity hinders the training of the several NILINKER models, and consequently, the impact of these in downstream tasks.

We used a simple rule to tokenize the entities in the generation of the word-concept dictionary and in the preprocessing of input entities that was based on white spaces and also on hyphens for chemical entities. This is not optimal for biomedical entities since these are frequently composed by complex rare words.

The NEL datasets used in the evaluation only included in-KB or linkable entities, but for future work it would be interesting to assess the performance of the REEL-NILINKER model in other datasets including out-of-KB entities, such as the one proposed by Miftahutdinov et al. [14].

Another limitation of NILINKER is that it uses non-contextual embeddings to represent the words that are part of the input entities. The model encodes two entities sharing their string with the same vector, but their context, thus their meaning, may be different. Both limitations impact the first step of the proposed approach, candidate retrieval, and consequently, the following steps. Candidate retrieval is essential because the attention weights are assigned according to the candidates associated with the entity, and the attention weights will influence the prediction scores.

5. Conclusion

In this work, we propose NILINKER, a model which can be used to partially link biomedical NIL entities to concepts belonging to several KBs (MEDIC, HP, CTD-Chemicals, ChEBI, GO-BP and CTD-ANAT) and to improve the performance of NEL models. NILINKER includes several components, namely a candidate retrieval from a word-concept dictionary, and a neural network leveraging the attention mechanism to determine the relevance of each KB concept to the input NIL entity. We also propose a new evaluation dataset specifically developed for the NIL entity linking task (EvaNIL), and we provide the results of the evaluation of several models. The best results were obtained for the partition EvaNIL-MEDIC ($F_1 score$ of 0.9401). On the other hand, we integrated NILINKER with REEL, a graph-based NEL model, and determined if the integration translated into an increase in performance in the NEL task. The results of the evaluation done in several NEL datasets show that integrating NILINKER with a NEL model leads to an increase of performance. Thus, we can conclude that the answers to the initial research questions are positive.

We focused on answering the initial research questions, but there is still room to improve NILINKER. So, future work should focus on solving the limitations that NILINKER still presents. Possible directions include the use of contextual embeddings to improve the representation layer, the adaptation of the model to accept entities with more than two words, the use of tokenizers trained on biomedical text and, crucially, the improvement of the training dataset EvaNIL, by increasing the diversity of annotations that are present (for example, each entity could be associated with more distant ancestors in the respective KB hierarchy). It would be also interesting to develop a pipeline for end-to-end Entity Extraction integrating Named Entity Recognition, NEL, and NIL entity linking.

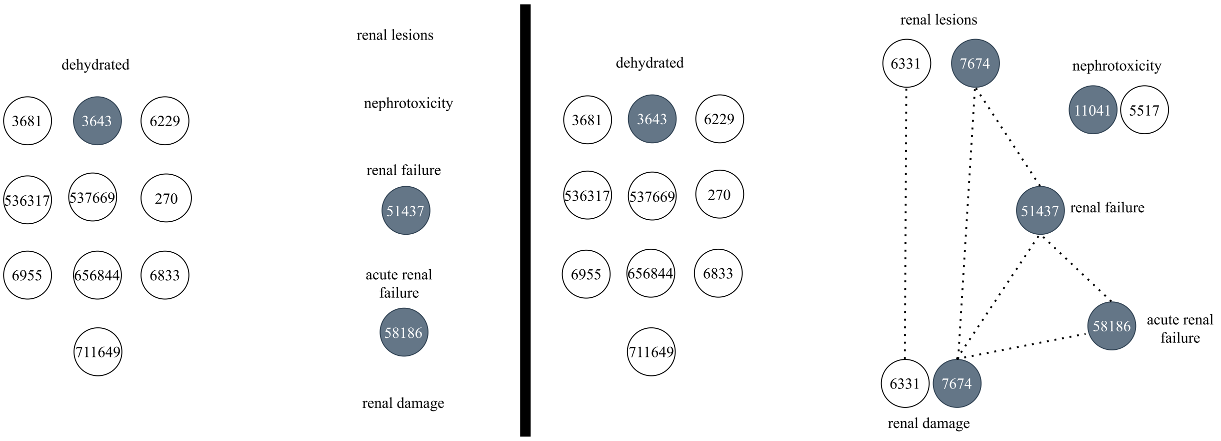


Fig. 3. Example of the impact of NILINKER-MEDIC in the disambiguation graph of REEL (document 931801 of the dataset BC5CDR-Disease). In the left, it is present the disambiguation graph produced by the REEL model including the MEDIC candidates (circles) for three linkable entity ("dehydrated", "renal failure" and "acute renal failure") and also several NIL entities ("renal lesions", "nephrotoxicity", "renal damage") that have an empty candidates' list. Grey circles correspond to the highest scored candidates for the respective entity. In the right, it is present the disambiguation graph after applying the model REEL-NILINKER: the previous NIL entities now have associated candidates, which by its turn will be involved in more edges (dashed lines), increasing the semantic information of the graph. In this case the entities "renal lesions" and "renal damage" are correctly disambiguated to the candidate 7674 (MESH:D007674).

CRediT authorship contribution statement

Pedro Ruas: Conceptualization, Methodology, Software, Writing – original draft, Investigation. **Francisco M. Couto:** Conceptualization, Writing – review & editing, Funding acquisition, Supervision.

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.

Acknowledgements

This work was supported by FCT (Fundação para a Ciência e a Tecnologia) through funding of the Deep Semantic Tagger (DeST) project (ref. PTDC/CCI-BIO/28685/2017), the PhD Scholarship with ref. 2020.05393.BD, and LASIGE Research Unit (ref. UIDB/00408/2020 and ref. UIDP/00408/2020).

References

[1] D. Rao, P. McNamee, M. Dredze, Entity linking: Finding extracted entities in a knowledge base, in: T. Poibeau, H. Saggion, J. Piskorski (Eds.), Multi-Source, Multilingual Information Extraction and Summarization. Theory and Applications of Natural Language Processing, Springer, Berlin, Heidelberg, 2013, pp. 93–115, http://dx.doi.org/10.1007/978-3-642-28569-1_5, URL http://link.springer.com/10.1007/978-3-642-28569-1_5.

[2] H. Ji, R. Grishman, H.T. Dang, K. Griffitt, J. Ellis, Overview of the TAC 2011 knowledge base population track, in: TAC 2011 Proceedings Papers, 2011, pp. 1–25.

[3] M. Dredze, P. Mcnamee, D. Rao, A. Gerber, T. Finin, Entity disambiguation for knowledge base population, in: Proceedings of the 23rd International Conference on Computational Linguistics, no. August, 2010, pp. 277–285, http://dx.doi.org/10.3115/1119176.1119181, arXiv:1604.00734.

[4] E. Meij, K. Balog, D. Odijk, Entity linking and retrieval for semantic search, in: WSDM 2014 - Proceedings of the 7th ACM International Conference on Web Search and Data Mining, no. February 2014, New York, New York, USA, 2014, p. 683, http://dx.doi.org/10.1145/2556195.2556201.

[5] D. Sorokin, I. Gurevych, Mixing context granularities for improved entity linking on question answering data across entity categories, in: 7th Joint Conference on Lexical and Computational Semantics (*SEM), Association for Computational Linguistics, 2018, pp. 65–75.

[6] S.F. Sung, C.Y. Lin, Y.H. Hu, EMR-based phenotyping of ischemic stroke using supervised machine learning and text mining techniques, IEEE J. Biomed. Health Inf. 24 (10) (2020) 2922–2931, http://dx.doi.org/10.1109/JBHI.2020.2976931.

[7] C. Combi, M. Zorzi, G. Pozzani, E. Arzenton, U. Moretti, Normalizing spontaneous reports into MedDRA: Some experiments with MagiCoder, IEEE J. Biomed. Health Inf. 23 (1) (2019) 95–102, http://dx.doi.org/10.1109/JBHI.2018.2861213.

[8] W. Shen, J. Wang, J. Han, Entity linking with a knowledge base: Issues, techniques, and solutions, IEEE Trans. Knowl. Data Eng. 27 (2) (2015) 443–460, http://dx.doi.org/10.1109/TKDE.2014.2327028.

[9] R. Bunescu, M. Pasca, Using encyclopedic knowledge for named entity disambiguation, in: Proceedings of the 11th Conference of the European Chapter of the Association for the Association for Computational Linguistics (EACL-06), no. April, Trento, Italy, 2006, pp. 9–16.

[10] K. Blissett, H. Ji, Cross-lingual NIL entity clustering for low-resource languages, in: Proceedings of the 2nd Workshop on Computational Models of Reference, Anaphora and Coreference (CRAC 2019), 2019 Association for Computational Linguistics, Minneapolis, USA, June 7, 2019., 2019, pp. 20–25, http://dx.doi.org/10.18653/v1/w19-2804, URL https://www.nist.gov/itl/iad/mig/lorehlt-evaluations.

[11] L. Chen, G. Varoquaux, F.M. Suchanek, A Lightweight Neural Model for Biomedical Entity Linking, Tech. rep., 2021, arXiv:2012.08844v1. URL www.aaai.org.

[12] M. Färber, A. Rettinger, B. El Asmar, On emerging entity detection, in: E. Blomqvist, P. Ciancarini, F. Poggi, F. Vitali (Eds.), Lecture Notes in Computer Science, Vol 10024, Springer, Cham, 2016, pp. 223–238, http://dx.doi.org/10.1007/978-3-319-49004-5_15, Ch. Knowledge.

[13] T. Lin, Mausam, O. Etzioni, No noun phrase left behind: Detecting and typing unlinkable entities, in: EMNLP-CoNLL 2012 - 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, Proceedings of the Conference, no. July, Association for Computational Linguistics, Jeju Island, Korea, 12–14 July 2012, 2012, pp. 893–903.

[14] Z. Miftahutdinov, A. Kadurin, R. Kudrin, E. Tutubalina, Medical concept normalization in clinical trials with drug and disease representation learning, Bioinformatics 37 (21) (2021) 3856–3864, http://dx.doi.org/10.1093/bioinformatics/btab474.

[15] L. Bornmann, R. Mutz, Growth rates of modern science: A bibliometric analysis based on the number of publications and cited references, J. Assoc. Inf. Sci. Technol. 66 (11) (2015) 2215–2222, http://dx.doi.org/10.1002/asi.23329, arXiv:1402.4578.

[16] R. Leaman, R.I. Doğan, Z. Lu, DNorm: Disease name normalization with pairwise learning to rank, Bioinformatics 29 (22) (2013) 2909–2917, http://dx.doi.org/10.1093/bioinformatics/btt474.

[17] J. D'Souza, V. Ng, Sieve-based entity linking for the biomedical domain, in: Proceedings Ofthe 53rd Annual Meeting Ofthe Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Short Papers), 2015 Association for Computational Linguistics, Beijing, China, July 26-31, 2015, 2015, pp. 297–302, http://dx.doi.org/10.3115/V1/P15-2049.

[18] P. Ruas, A. Lamurias, F.M. Couto, Linking chemical and disease entities to ontologies by integrating PageRank with extracted relations from literature, J. Cheminform. 12 (1) (2020) 1–11, http://dx.doi.org/10.1186/s13321-020-00461-4.

[19] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C.H. So, J. Kang, BioBERT: A pre-trained biomedical language representation model for biomedical text mining, Bioinformatics 36 (4) (2020) 1234–1240, http://dx.doi.org/10.1093/bioinformatics/btz682, arXiv:1901.08746.

[20] Y. Gu, R. Tinn, H. Cheng, M. Lucas, N. Usuyama, X. Liu, T. Naumann, J. Gao, H. Poon, Domain-specific language model pretraining for biomedical natural language processing, ACM Trans. Comput. Healthc. 3 (1) (2022) 1–23, http://dx.doi.org/10.1145/3458754, arXiv:2007.15779. URL http://arxiv.org/abs/2007.15779.

[21] I. Beltagy, K. Lo, A. Cohan, SCIBERT: A pretrained language model for scientific text, in: EMNLP-IJCNLP 2019 - 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Proceedings of the Conference, Association for Computational Linguistics (ACL), Hong Kong, China, 2020, pp. 3615–3620, http://dx.doi.org/10.18653/v1/d19-1371, arXiv:1903.10676. URL http://arxiv.org/abs/1903.10676.

[22] Z. Ji, Q. Wei, H. Xu, BERT-based ranking for biomedical entity normalization, 2019, arXiv:1908.03548. URL http://arxiv.org/abs/1908.03548.

[23] M. Sung, M. Jeong, Y. Choi, D. Kim, J. Lee, J. Kang, BERN2: an advanced neural biomedical named entity recognition and normalization tool, 2022, pp. 1–6, arXiv:2201.02080. URL http://arxiv.org/abs/2201.02080.

[24] G. Wu, Y. He, X. Hu, Entity linking: An issue to extract corresponding entity with knowledge base, IEEE Access 6 (2018) 6220–6231, http://dx.doi.org/10.1109/ACCESS.2017.2787787, URL http://ieeexplore.ieee.org/document/8246707/.

[25] L. Derczynski, E. Nichols, M. van Erp, N. Limsopatham, Results of the WNUT2017 shared task on novel and emerging entity recognition, in: Proceedings Ofthe 3rd Workshop on Noisy User-Generated Text, Association for Computational Linguistics, Copenhagen, Denmark, September 7, 2017, 2018, pp. 140–147, http://dx.doi.org/10.18653/v1/w17-4418, URL https://stackexchange.com.

[26] A. Galassi, M. Lippi, P. Torroni, Attention in natural language processing, IEEE Trans. Neural Netw. Learn. Syst. (2019) 1–18, http://dx.doi.org/10.1109/TNNLS.2020.3019893, arXiv:1902.02181. URL http://arxiv.org/abs/1902.02181.

[27] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: 31st Conference on Neural Information Processing Systems (NIPS 2017, Long Beach, CA, USA, 2017.

[28] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, 2018, arXiv:1810.04805. URL http://arxiv.org/abs/1810.04805.

[29] E. Alsentzer, J. Murphy, W. Boag, W.-H. Weng, D. Jindi, T. Naumann, M. McDermott, Publicly available clinical BERT embeddings, in: Proceedings Ofthe 2nd Clinical Natural Language Processing Workshop, Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 72–78, http://dx.doi.org/10.18653/v1/w19-1909.

[30] F. Li, Y. Jin, W. Liu, B.P.S. Rawat, P. Cai, H. Yu, Fine-tuning bidirectional encoder representations from transformers (BERT)–based models on large-scale electronic health record notes: An empirical study, J. Med. Internet Res. 21 (9) (2019) http://dx.doi.org/10.2196/14830.

[31] X. Yin, Y. Huang, B. Zhou, A. Li, L. Lan, Y. Jia, Deep entity linking via eliminating semantic ambiguity with BERT, IEEE Access 7 (2019) 169434–169445, http://dx.doi.org/10.1109/ACCESS.2019.2955498.

[32] F. Nie, Y. Cao, J. Wang, C.Y. Lin, R. Pan, Mention and entity description co-attention for entity disambiguation, in: 32nd AAAI Conference on Artificial Intelligence, AAAI 2018, 2018, pp. 5908–5915.

[33] F. Qi, J. Huang, C. Yang, Z. Liu, X. Chen, Q. Liu, M. Sun, Modeling semantic compositionality with sememe knowledge, in: Proceedings Ofthe 57th Annual Meeting Ofthe Association for Computational Linguistics, Association for Computational Linguistics, Florence, Italy, July 28 - August 2, 2019, 2019, pp. 5706–5715, http://dx.doi.org/10.18653/v1/p19-1571, arXiv:1907.04744.

[34] F.J. Pelletier, The principle of semantic compositionality, Topoi 13 (1) (1994) 11–24, http://dx.doi.org/10.1007/BF00763644.

[35] Z. Dong, Q. Dong, HowNet - A hybrid language and knowledge resource, in: Proceedings of the International Conference on Natural Language Processing and Knowledge Engineering (NLP-KE) 2003, 2003, pp. 820–824, http://dx.doi.org/10.1109/NLPKE.2003.1276017.

[36] T. Mikolov, I. Sutskever, K.C. Chen, G.S. Corrado, J.J. Dean, Distributed representations of words and phrases and their compositionality, in: NIPS'13: Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2, Vol. 26, Curran Associates Inc.57 Morehouse LaneRed HookNYUnited States, 2013, pp. 3111–3119.

[37] Y. Zhang, Q. Chen, Z. Yang, H. Lin, Z. Lu, BioWordVec, improving biomedical word embeddings with subword information and mesh, Sci. Data 6 (1) (2019) 52, http://dx.doi.org/10.1038/s41597-019-0055-0.

[38] A. Grover, J. Leskovec, Node2Vec: Scalable feature learning for networks, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, San Francisco, USA, August 13 - 17, 2016, 2016, pp. 855–864, http://dx.doi.org/10.1145/2939672.2939754.

[39] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, J. Mach. Learn. Res. 12 (2011) 2825–2830.

[40] A.P. Davis, C.J. Grondin, R.J. Johnson, D. Sciaky, J. Wiegers, T.C. Wiegers, C.J. Mattingly, Comparative toxicogenomics database (CTD): update 2021, Nucleic Acids Res. (2020) http://dx.doi.org/10.1093/nar/gkaa891, arXiv:https://academic.oup.com/nar/advance-article-pdf/doi/10.1093/nar/gkaa891/33931329/gkaa891.pdf. gkaa891.

[41] J. Hastings, G. Owen, A. Dekker, M. Ennis, N. Kale, V. Muthukrishnan, S. Turner, N. Swainston, P. Mendes, C. Steinbeck, ChEBI in 2016: Improved services and an expanding collection of metabolites, Nucleic Acids Res. 44 (D1) (2016) D1214—9, http://dx.doi.org/10.1093/nar/gkv1031, URL https://europepmc.org/articles/PMC4702775.

[42] The Gene Ontology Consortium, The gene ontology resource: 20 years and still going strong, Nucleic Acids Res. 47 (D1) (2018) D330–D338, http://dx.doi.org/10.1093/nar/gky1055, arXiv:https://academic.oup.com/nar/article-pdf/47/D1/D330/27437640/gky1055.pdf.

[43] S. Köhler, M. Gargano, N. Matentzoglu, L.C. Carmody, D. Lewis-Smith, N.A. Vasilevsky, D. Danis, G. Balagura, G. Baynam, A.M. Brower, T.J. Callahan, C.G. Chute, J.L. Est, P.D. Galer, S. Ganesan, M. Griese, M. Haimel, J. Pazmandi, M. Hanauer, N.L. Harris, M.J. Hartnett, M. Hastreiter, F. Hauck, Y. He, T. Jeske, H. Kearney, G. Kindle, C. Klein, K. Knoflach, R. Krause, D. Lagorce, J.A. McMurry, J.A. Miller, M.C. Munoz-Torres, R.L. Peters, C.K. Rapp, A.M. Rath, S.A. Rind, A.Z. Rosenberg, M.M. Segal, M.G. Seidel, D. Smedley, T. Talmy, Y. Thomas, S.A. Wiafe, J. Xian, Z. Yüksel, I. Helbig, C.J. Mungall, M.A. Haendel, P.N. Robinson, The human phenotype ontology in 2021, Nucleic Acids Res. 49 (D1) (2021) D1207–D1217, http://dx.doi.org/10.1093/NAR/GKAA1043, URL https://pubmed.ncbi.nlm.nih.gov/33264411/.

[44] E. Tutubalina, A. Kadurin, Z. Miftahutdinov, Fair evaluation in concept normalization: a large-scale comparative analysis for BERT-based models, in: Proceedings Ofthe 28th International Conference on Computational Linguistics, Barcelona, Spain (Online), December 8-13, 2020, 2021, pp. 6710–6716, http://dx.doi.org/10.18653/v1/2020.coling-main.588.

[45] J. Li, Y. Sun, R.J. Johnson, D. Sciaky, C.H. Wei, R. Leaman, A.P. Davis, C.J. Mattingly, T.C. Wiegers, Z. Lu, BioCreative V CDR task corpus: a resource for chemical disease relation extraction, Database 2016 (2016) 1–10, http://dx.doi.org/10.1093/database/baw068.

[46] R.I. Doğan, R. Leaman, Z. Lu, NCBI disease corpus: A resource for disease name recognition and concept normalization, J. Biomed. Inform. 47 (2014) 1–10, http://dx.doi.org/10.1016/j.jbi.2013.12.006.

[47] T. Groza, S. Köhler, S. Doelken, N. Collier, A. Oellrich, D. Smedley, F.M. Couto, G. Baynam, A. Zankl, P.N. Robinson, Automatic concept recognition using the human phenotype ontology reference and test suite corpora, Database 2015 (2015) 1–13, http://dx.doi.org/10.1093/database/bav005.

[48] M. Lobo, A. Lamurias, F.M. Couto, Identifying human phenotype terms by combining machine learning and validation rules, BioMed Res. Int. 2017 (2017) http://dx.doi.org/10.1155/2017/8565739.

[49] S.K. Sahu, F. Christopoulou, M. Miwa, S. Ananiadou, Inter-sentence relation extraction with document-level graph convolutional neural network, in: ACL 2019 - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, Association for Computational Linguistics (ACL), Florence, Italy, July 28 - August 2, 2019, 2019, pp. 4309–4316, http://dx.doi.org/10.18653/v1/p19-1423, arXiv:1906.04684.

[50] P. Thompson, S. Daikou, K. Ueno, R. Batista-Navarro, J. Tsujii, S. Ananiadou, Annotation and detection of drug effects in text for pharmacovigilance, J. Cheminform. 10 (1) (2018) 1–33, http://dx.doi.org/10.1186/s13321-018-0290-y.

[51] S. Vashishth, D. Newman-Griffis, R. Joshi, R. Dutt, C.P. Rosé, Improving broad-coverage medical entity linking with semantic type prediction and large-scale datasets, J. Biomed. Inform. 121 (2021) 103880, http://dx.doi.org/10.1016/j.jbi.2021.103880.

[52] K. Bretonnel Cohen, K. Verspoor, K. Fort, C. Funk, M. Bada, M. Palmer, L. Hunter, The colorado richly annotated full text (CRAFT) corpus: Multi-model annotation in the biomedical domain, in: Handbook of Linguistic Annotation, 2016, URL https://hal.inria.fr/hal-01159065.

[53] S. Mohan, D. Li, Medmentions: A large biomedical corpus annotated with UMLS concepts, 2019, arXiv:1902.09476.

[54] R. Dror, G. Baumer, S. Shlomov, R. Reichart, The hitchhiker's guide to testing statistical significance in natural language processing, in: Proceedings Ofthe 56th Annual Meeting Ofthe Association for Computational Linguistics (Long Papers), Association for Computational Linguistics, Melbourne, Australia, July 15 - 20, 2018, 2018, pp. 1383–1392, URL https://github.com/rtmdrr/.

[55] G.M. Sullivan, R. Feinn, Using effect size—or why the P value is not enough, J. Grad. Med. Educ. 4 (3) (2012) 279, http://dx.doi.org/10.4300/JGME-D-12-00156.1, URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3444174/.