Neural Entity Linking: A Survey of Models Based on Deep Learning

Özge Sevgili [a,*], Artem Shelmanov [d,b,c,**], Mikhail Arkhipov [e], Alexander Panchenko [b], Chris Biemann [a]

[a] *Language Technology Group, Universität Hamburg, Informatikum, Vogt-Kölln-Straße 30, 22527 Hamburg, Germany*
E-mails: oezge.sevgili.ergueven@studium.uni-hamburg.de, christian.biemann@uni-hamburg.de
[b] *Center for Artificial Intelligence Technologies, Skolkovo Institute of Science and Technology, Bolshoy Boulevard 30, bld. 1, 121205, Moscow, Russia*
E-mail: a.panchenko@skoltech.ru
[c] *Research Computing Center, Lomonosov Moscow State University, GSP-1, Leninskie Gory, 119991, Moscow, Russia*
[d] *AIRI, Nizhny Susalny lane 5 p. 19, 105064, Moscow, Russia*
E-mail: shelmanov@airi.net
[e] *Neural Networks and Deep Learning Laboratory, Moscow Institute of Physics and Technology, 9 Institutskiy per., Dolgoprudny, 141701, Moscow, Russia*
E-mail: arkhipov@yahoo.com

Abstract. This survey presents a comprehensive description of recent neural entity linking (EL) systems developed since 2015 as a result of the "deep learning revolution" in natural language processing. Its goal is to systemize design features of neural entity linking systems and compare their performance to the remarkable classic methods on common benchmarks. This work distills a generic architecture of a neural EL system and discusses its components, such as candidate generation, mention-context encoding, and entity ranking, summarizing prominent methods for each of them. The vast variety of modifications of this general architecture are grouped by several common themes: joint entity mention detection and disambiguation, models for global linking, domain-independent techniques including zero-shot and distant supervision methods, and cross-lingual approaches. Since many neural models take advantage of entity and mention/context embeddings to represent their meaning, this work also overviews prominent entity embedding techniques. Finally, the survey touches on applications of entity linking, focusing on the recently emerged use-case of enhancing deep pre-trained masked language models based on the Transformer architecture.

Keywords: Entity Linking, Deep Learning, Neural Networks, Natural Language Processing, Knowledge Graphs

---

[*]Equal contribution. Corresponding author. E-mail: oezge.sevgili.ergueven@studium.uni-hamburg.de.
[**]Equal contribution. Corresponding author. E-mail: shelmanov@airi.net.

1. Introduction

Knowledge Graphs (KGs), such as Freebase [14], DBpedia [92], and Wikidata [184], contain rich and

precise information about entities of all kinds, such as persons, locations, organizations, movies, and scientific theories, just to name a few. Each entity has a set of carefully defined relations and attributes, e.g. "was born in" or "play for". This wealth of structured information gives rise to and facilitates the development of semantic processing algorithms as they can directly operate on and benefit from such entity representations. For instance, imagine a search engine that is able to retrieve mentions in the news during the last month of all retired NBA players with a net income of more than 1 billion US dollars. The list of players together with their income and retirement information may be available in a knowledge graph. Equipped with this information, it appears to be straightforward to look up mentions of retired basketball players in the newswire. However, the main obstacle in this setup is the lexical ambiguity of entities. In the context of this application, one would want to only retrieve all mentions of "Michael Jordan (basketball player)"[1] and exclude mentions of other persons with the same name such as "Michael Jordan (mathematician)"[2].

This is why Entity Linking (EL) – the process of matching a mention, e.g. "Michael Jordan", in a textual context to a KG record (e.g. "basketball player" or "mathematician") fitting the context – is the key technology enabling various semantic applications. Thus, EL is the task of identifying an entity mention in the (unstructured) text and establishing a link to an entry in a (structured) knowledge graph.

Entity linking is an essential component of many information extraction (IE) and natural language understanding (NLU) pipelines since it resolves the lexical ambiguity of entity mentions and determines their meanings in context. A link between a textual mention and an entity in a knowledge graph also allows us to take advantage of the information encompassed in a semantic graph, which is shown to be useful in such NLU tasks as information extraction, biomedical text processing, or semantic parsing and question answering (see Section 5). This wide range of direct applications is the reason why entity linking is enjoying great interest from both academy and industry for more than two decades.

1.1. Goal and Scope of this Survey

Recently, a new generation of approaches for entity linking based on neural models and deep learning emerged, pushing the state-of-the-art performance in this task to a new level. The goal of our survey is to provide an overview of this latest wave of models, emerging from 2015.

Models based on neural networks have managed to excel in EL as in many other natural language processing tasks due to their ability to learn useful distributed semantic representations of linguistic data [11, 30, 203]. These current state-of-the-art neural entity linking models have shown significant improvements over "classical"[3] machine learning approaches [27, 84, 148] to name a few that are based on shallow architectures, e.g. Support Vector Machines, and/or depend mostly on hand-crafted features. Such models often cannot capture all relevant statistical dependencies and interactions [53]. In contrast, deep neural networks are able to learn sophisticated representations within their deep layered architectures. This reduces the burden of manual feature engineering and enables significant improvements in EL and other tasks.

In this survey, we systemize recently proposed neural models, distilling one generic architecture used by the majority of neural EL models (illustrated in Figures 2 and 5). We describe the models used in each component of this architecture, e.g. candidate generation, mention-context encoding, entity ranking. Prominent variations of this generic architecture, e.g. end-to-end EL or global models, are also discussed. To better structure the sheer amount of available models, various types of methods are illustrated in taxonomies (Figures 3 and 6), while notable features of each model are carefully assembled in a tabular form (Table 2). We discuss the performance of the models on commonly used entity linking/disambiguation benchmarks and an entity relatedness dataset. Because of the sheer amount of work, it was not possible for us to try available software and to compare approaches on further parameters, such as computational complexity, run-time, and memory requirements. Nevertheless, we created a comprehensive collection of references to publicly available official implementations of EL models and systems discussed in this survey (see Table 7 in Appendix A).

---

[1]https://en.wikipedia.org/wiki/Michael_Jordan
[2]https://en.wikipedia.org/wiki/Michael_I._Jordan

[3]On classical ML vs deep learning: https://towardsdatascience. com/deep-learning-vs-classical-machine-learning-9a42c6d48aa

An important component of neural entity linking systems is distributed entity representations and entity encoding methods. It has been shown that encoding the KG structure (entity relationships), entity definitions, or word/entity co-occurrence statistics from large textual corpora in low-dimensional vectors improves the generalization capabilities of EL models [53, 70]. Therefore, we also summarize distributed entity representation models and novel methods for entity encoding.

Many natural language processing systems take advantage of deep pre-trained language models like ELMo [138], BERT [36], and their modifications. EL made its path into these models as a way of introducing information stored in KGs, which helps to adapt word representations to some text processing tasks. We discuss this novel application of EL and its further development.

1.2. Article Collection Methodology

We do not have a strict article collection algorithm for the review like e.g., the one conducted by Oliveira et al. [130]. Our main goal is to provide and describe a conceptual framework that can be applied to the majority of recently presented neural approaches to EL. Nevertheless, as with all surveys, we had to draw the line somewhere. The main criteria for including papers into this survey was that they had been published during or after 2015, and they primarily address the task of EL, i.e. resolving textual mentions to entries in KGs, or discussing EL applications. We explicitly exclude related work e.g., on (fine-grained) entity typing (see [4, 28]), which also encompasses a disambiguation task, and work that employs KGs for other tasks than EL. This survey also does not try to cover all EL methods designed for specific domains like biomedical texts or messages in social media. For the general-purpose EL models evaluated on well-established benchmarks, we try to be as comprehensive as possible with respect to recent-enough papers that fit into the conceptual framework, no matter where they have appeared (however, with a focus on top conferences and journals in the fields of natural language processing and Semantic Web).

1.3. Previous Surveys

One of the first surveys on EL was prepared by Shen et al. [160] in 2015. They cover the main approaches to entity linking (within the modules, e.g. candidate generation, ranking), its applications, evaluation methods, and future directions. In the same year, Ling et al. [97] presented a work that aims to provide (1) a standard problem definition to reduce confusion that appears due to the existence of variant similar tasks related to EL (e.g., Wikification [112] and named entity linking [67]), and (2) a clear comparison of models and their various aspects.

There are also other surveys that address a wider scope. The work of Martínez-Rodríguez et al. [106], published in 2020, involves information extraction models and semantic web technologies. Namely, they consider named entity recognition, entity linking, terminology extraction, keyphrase extraction, topic modeling, topic labeling, and relation extraction tasks. In a similar vein, the work of Al-Moslmi et al. [3], released in 2020, overviews the research in named entity recognition, named entity disambiguation, and entity linking published between 2014 and 2019.

Another recent survey paper by Oliveira et al. [130], published in 2020, analyses and summarizes EL approaches that exhibit some holism. This viewpoint limits the survey to the works that exploit various peculiarities of the EL task: additional metadata stored in specific input like microblogs, specific features that can be extracted from this input like geographic coordinates in tweets, timestamps, interests of users posted these tweets, and specific disambiguation methods that take advantage of these additional features. In the concurrent work, Möller et al. [118] overview models developed specifically for linking English entities to the Wikidata [184] and discuss features of this KG that can be exploited for increasing the linking performance.

Previous surveys on similar topics (a) do not cover many recent publications [97, 160], (b) broadly cover numerous topics [3, 106], or (c) are focused on the specific types of methods [130] or a knowledge graph [118]. There is not yet, to our knowledge, a detailed survey specifically devoted to recent neural entity linking models. The previous surveys also do not address the topics of entity and context/mention encoding, applications of EL to deep pre-trained language models, and cross-lingual EL. We are also the first to summarize the domain-independent approaches to EL, several of which are based on zero-shot techniques.

1.4. Contributions

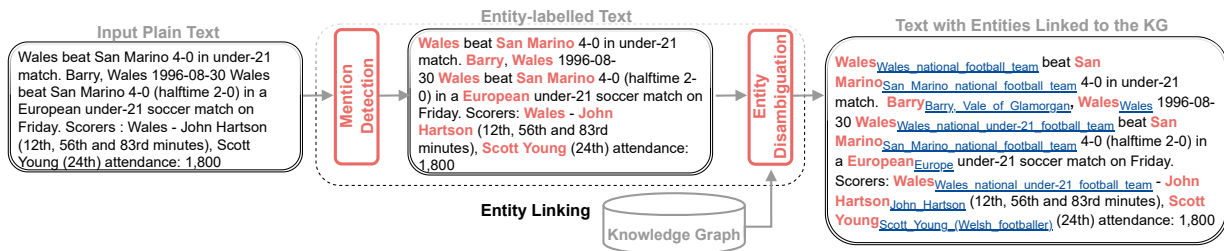More specifically, this article makes the following contributions:

Fig. 1. **The entity linking task**. An Entity Linking (EL) model takes a raw textual input and enriches it with entity mentions linked to nodes in a Knowledge Graph (KG). The task is commonly split into entity mention detection and entity disambiguation sub-tasks.

- a survey of state-of-the-art neural entity linking models;
- a systematization of various features of neural EL methods and their evaluation results on popular benchmarks;
- a summary of entity and context/mention embedding techniques;
- a discussion of recent domain-independent (zero-shot) and cross-lingual EL approaches;
- a survey of EL applications to modeling word representations.

The structure of this survey is the following. We start with defining the EL task in Section 2. In Section 3.1, the general architecture of neural entity linking systems is presented. Modifications and variations of this basic pipeline are discussed in Section 3.2. In Section 4, we summarize the performance of EL models on standard benchmarks and present results of the entity relatedness evaluation. Section 5 is dedicated to applications of EL with a focus on recently emerged applications for improving neural language models. Finally, Section 6 concludes the survey and suggests promising directions of future work.

2. Task Description

2.1. Informal Definition

Consider the example presented in Figure 1 with an entity mention *Scott Young* in a soccer-game-related context. Literally, this common name can refer to at least three different people: the *American football player*, the *Welsh football player*, or the *writer*. The EL task is to (1) correctly detect the entity mention in the text, (2) resolve its ambiguity and ultimately provide a link to a corresponding entity entry in a KG, e.g. provide for the *Scott Young* mention in this context

a link to the *Welsh footballer*[4] instead of the *writer*[5]. To achieve this goal, the task is usually decomposed into two sub-tasks, as illustrated in Figure 1: Mention Detection (MD) and Entity Disambiguation (ED).

2.2. Formal Definition

2.2.1. Knowledge Graph (KG)

A KG contains entities, relations, and facts, where facts are denoted as triples (i.e. head entity, relation, tail entity) as defined in Ji et al. [77]. Formally, as defined by Färber et al. [45], a KG is a set of RDF triples where each triple (s, p, o) is an ordered set of the following terms: a subject $s \in U \cup B$, a predicate $p \in U$, and an object $o \in U \cup B \cup L$. An RDF term is either a URI $u \in U$, a blank node $b \in B$, or a literal $l \in L$. URI (or IRI) nodes are for the global identification of entities on the Web; literal nodes are for strings and other datatype values (e.g. integers, dates); and the blank node is for anonymous nodes, which are not assigned an identifier, as explained in Hogan et al. [68].

This RDF representation can be considered as a multi-relational graph $G = (E, \mathbb{A} = \{A_0, A_1, ..., A_m \subseteq (E \times E)\})$, where E is a set of all entities of a KG, and $\mathbb{A}$ is a family of typed edge sets of length m. For example, A_0 is the "occupation" predicate adjacency matrix, A_1 is the "founded" predicate adjacency matrix, etc.

There is also an equivalent three-way tensor representation of a KG $\mathcal{A} \in \{0, 1\}^{n \times m \times n}$, where

$$\mathcal{A}_{i,k,j} = \begin{cases} 1 & \text{if } (i, j) \in A_k : k \leqslant m \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

2.2.2. Mention Detection (MD)

The goal of mention detection is to identify an entity mention span, while entity disambiguation performs
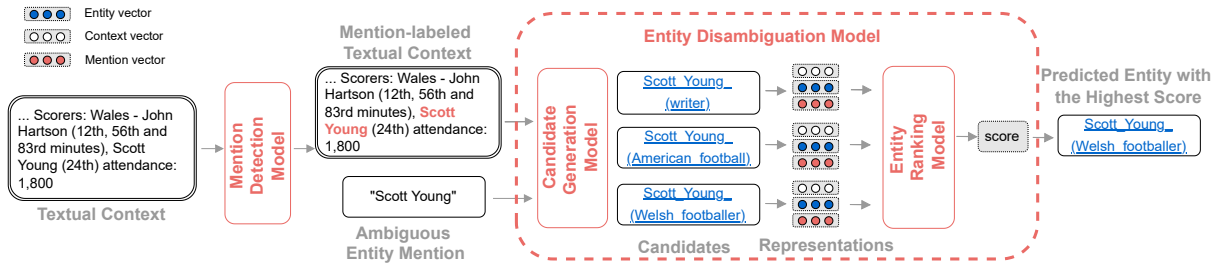

Fig. 2. **General architecture for neural entity linking.** Entity Linking (EL) consists of two main steps: *Mention Detection (MD)*, when entity mention boundaries in a text are identified, and *Entity Disambiguation (ED)*, when a corresponding entity is predicted for the given mention. Entity disambiguation is further carried out in two steps: *Candidate Generation*, when possible candidate entities are selected for the mention, and *Entity Ranking*, when a correspondence score between context/mention and each candidate is computed through the comparison of their vector representations.

linking of found mentions to entries of a KG. We can consider this task as determining an MD function that takes as input a textual context $c_i \in C$ (e.g. a document in a document collection) and outputs a sequence of n mentions $(m_1, \ldots m_n)$ in this context $m_i \in M$, where M is a set of all possible text spans in the context:

$$\mathrm{MD} : C \to M^n. \quad (2)$$

In the majority of works on EL, it is assumed that the mentions are already given or detected, for example, using a named entity recognition (NER) system (sometimes called named entity recognition and classification (NERC) [4, 119]). We should note that, usually, in addition to MD, NER systems also tag/classify mentions with a predefined types [95, 107, 130, 181] that also can be leveraged for disambiguation [107].

2.2.3. Entity Disambiguation (ED)

The entity disambiguation task can be considered as determining a function ED that, given a sequence of n mentions in a document and their contexts $(c_1, \ldots, c_n)$, outputs an entity assignment $(e_1, \ldots, e_n), e_i \in E$, where E is a set of entities in a KG:

$$\mathrm{ED} : (M, C)^n \to E^n. \quad (3)$$

To learn a mapping from entity mentions in a context to entity entries in a KG, EL models use supervision signals like manually annotated mention-entity pairs. The size of KGs varies; they can contain hundreds of thousands or even millions of entities. Due to their large size, training data for EL would be extremely unbalanced; training sets can lack even a single example for a particular entity or mention, e.g. as in the popular AIDA corpus [67]. To deal with this prob-

lem, EL models should have wide generalization capabilities.

Despite KGs being usually large, they are incomplete. Therefore, some mentions in a text cannot be correctly mapped to any KG entry. Determining such unlinkable mentions, which usually is designated as linking to a NIL entry, is one of the current EL challenges. Methods that address this problem provide a separate function for it or extend the set of entities in the disambiguation function with this special entry:

$$\mathrm{ED} : (M, C)^n \to (E \cup \mathrm{NIL})^n. \quad (4)$$

2.3. Terminological Aspects

More or less, the same technologies and models are sometimes called differently in the literature. Namely, Wikification [26] and entity disambiguation are considered as subtypes of EL [115]. To be comprehensive in this survey, we assume that the entity linking task encompasses both entity mention detection and entity disambiguation. However, only a few studies suggest models that perform MD and ED jointly, while the majority of papers on EL focus exclusively on ED and assume that mention boundaries are given by an external entity recognizer [152] (which may lead to some terminological confusions). Numerous techniques that perform MD (e.g. in the NER task) without entity disambiguation are considered in many previous surveys [57, 95, 119, 159, 193] inter alia and are out of the scope of this work.

Entity linking in the general case is not restricted to linking mentions to graph nodes but rather to concepts in a knowledge base. However, most of the modern widely-used knowledge bases organize information in the form of a graph [14, 92, 184], even in par-

ticular domains, like e.g. the scholarly domain [34]. A basic statement in a data/knowledge base usually can be represented as a subject-predicate-object tuple (s, p, o), e.g. (John_Lennon, occupation, singer) or (New_-York_City, founded, 1624), and a set of such tuples can be represented as a multi-relational graph. This formalism helps to efficiently organize knowledge for many applications ranging from search engines to question answering and recommendation systems [68, 77]. Therefore, in this article, the terms Knowledge Graph (KG) and Knowledge Base (KB) are used interchangeably.

3. Neural Entity Linking

We start the discussion of neural entity linking approaches from the most general architecture of EL pipelines and continue with various specific modifications like joint entity mention detection and linking, disambiguation techniques that leverage global context, domain-independent EL approaches including zero-shot methods, and cross-lingual models.

3.1. General Architecture

Some of the attempts to EL based on neural networks treat it as a multi-class classification task in which entities correspond to classes. However, the straightforward approach results in a large number of classes, which leads to suboptimal performance without task-sharing [80]. The streamlined approach to EL is to treat it as a ranking problem. We present the generalized EL architecture in Figure 2, which is applicable to the majority of neural approaches. Here, the mention detection model identifies the mention boundaries in text. The next step is to produce a shortlist of possible entities (candidates) for the mention, e.g. producing Scott_Young_(writer) as a candidate rather than a completely random entity. Then, the mention encoder produces a semantic vector representation of a mention in a context. The entity encoder produces a set of vector representations of candidates. Finally, the entity ranking model compares mention and entity representations and estimates mention-entity correspondence scores. An optional step is to determine unlinkable mentions, for which a KG does not contain a corresponding entity. The categorization of each step in the general neural EL architecture is summarized in Figure 3.

3.1.1. Candidate Generation

An essential part of EL is candidate generation. The goal of this step is given an ambiguous entity mention, such as "Scott Young", to provide a list of its possible "senses" as specified by entities in a KG. EL is analogous to the Word Sense Disambiguation (WSD) task [115, 121] as it also resolves lexical ambiguity. Yet in WSD, each sense of a word can be clearly defined by WordNet [46], while in EL, KGs do not provide such an exact mapping between mentions and entities [22, 115, 121]. Therefore, a mention potentially can be linked to any entity in a KG, resulting in a large search space, e.g. "Big Blue" referring to IBM. In the candidate generation step, this issue is addressed by performing effective preliminary filtering of the entity list.

Formally, given a mention m_i, a candidate generator provides a list of probable entities, $e_1, e_2, ..., e_k$, for each entity mention in a document.

$$\text{CG} : M \rightarrow (e_1, e_2, ..., e_k). \tag{5}$$

Similar to [3, 160], we distinguish three common candidate generation methods in neural EL: (1) based on surface form matching, (2) based on expansion with aliases, and (3) based on a prior matching probability computation. In the first approach, a candidate list is composed of entities that match various surface forms of mentions in the text [87, 114, 211]. There are many heuristics for the generation of mention forms and matching criteria like the Levenshtein distance, n-grams, and normalization. For the example mention of "Big Blue", this approach would not work well, as the referent entity "IBM" or its long-form "International Business Machines" does not contain a mention string. Examples of candidate entity sets are presented in Table 1, where we searched a name matching of the mention "Big Blue" in the titles of all Wikipedia articles present in DBpedia and presented random 5 matches.

In the second approach, a dictionary of additional aliases is constructed using KG metadata like disambiguation/redirect pages of Wikipedia [43, 211] or using a dictionary of aliases and/or synonyms (e.g. "NYC" stands for "New York City"). This helps to improve the candidate generation recall as the surface form matching usually cannot catch such cases. Pershina et al. [137] expand the given mention to the longest mention in a context found using coreference
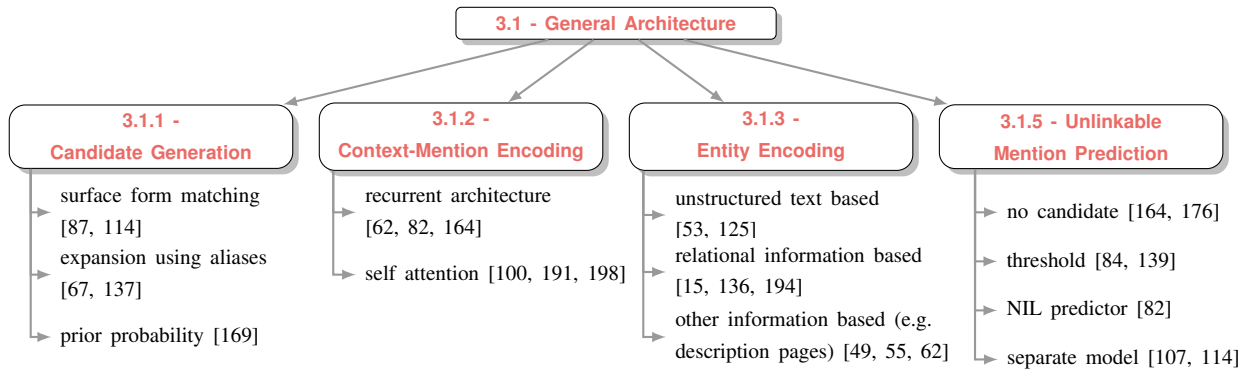

Fig. 3. **Reference map of the general architecture of neural EL systems.** The categorization of each step in the general neural EL architecture with alternative design choices and example references illustrating each of the choices.

Table 1

Candidate generation examples. Candidate entities for the example mention "Big Blue" obtained using several candidate generation methods. The highlighted candidates are "correct" entities assuming that the given mention refers to the IBM corporation and not a river, e.g. Big_Blue_-River_(Kansas).

Method	5 candidate entities for the example mention "Big Blue"
surface form matching based on DBpedia names[6]	Big_Blue_Trail, Big_Bluegrass, Big_Blue_Spring_cave_crayfish, Dexter_Bexley_and_the_Big_Blue_Beastie, IBM_Big_Blue_(X-League)
expansion using aliases from YAGO-means[7]	Big_Blue_River_(Indiana), Big_Blue_River_(Kansas), Big_Blue_(crane), Big_Red_(drink), **IBM**
probability + expansion using aliases from [53]: Anchor prob. + CrossWikis + YAGO[8]	**IBM**, Big_Blue_River_(Kansas), The_Big_Blue Big_Blue_River_(Indiana), Big_Blue_(crane)

resolution. Then, an entity is selected as a candidate if its title matches the longest version of the mention, or it is present in disambiguation/redirect pages of this mention. This resource is used in many EL models, e.g. [19, 107, 125, 131, 144, 164, 194]. Another well-known alternative is YAGO [170] – an ontology automatically constructed from Wikipedia and WordNet. Among many other relations, it provides *"means"* relations, and this mapping is utilized for candidate generation like in [53, 67, 157, 164, 194]. In this technique, the external information would help to disambiguate "Big Blue" as "IBM". Table 1 shows examples of candidates generated with the help of the YAGO-means candidate mapping dataset used in Hoffart et al. [67].

The third approach to candidate generation is based on pre-calculated prior probabilities of correspondence between certain mentions and entities, $p(e|m)$. Many studies rely on mention-entity priors computed based on Wikipedia entity hyperlinks. A URL of a hyperlink to an entity page of Wikipedia determines a candidate entity, and the anchor text of the hyperlink determines a mention. Another widely-used option is CrossWikis [169], which is an extensive resource that leverages the frequency of mention-entity links in web crawl data [53, 62].

It is common to apply multiple approaches to candidate generation at once. For example, the resource constructed by Ganea and Hofmann [53] and used in many other EL methods [82, 86, 139, 158, 198] relies on prior probabilities obtained from entity hyperlink count statistics of CrossWikis [169] and Wikipedia, as well as on entity aliases obtained from the "means" relationship of the YAGO ontology Hoffart et al. [67]. The illustrative mention "Big Blue" can be linked to its referent entity "IBM" with this method, as shown in Table 1. As another example, Fang et al. [44] utilize surface form matching and aliases. They share candidates between abbreviations and their expanded versions in the local context. The aliases are obtained from Wikipedia redirect and disambiguation pages, the Wikipedia search engine, and synonyms from Word-

---

[7]YAGO-means dataset of Hoffart et al. [67] – http://resources.mpi-inf.mpg.de/yago-naga/aida/download/aida_means.tsv.bz2

[8]We generated these examples using the source code of Peters et al. [139] – https://github.com/allenai/kb

Net [46]. Additionally, they submit mentions that are misspelled or contain multiple words to Wikipedia and Google search engines and search for the corresponding Wikipedia articles. It is also worth noting that some works also employ a candidate pruning step to reduce the number of candidates.

Recent zero-shot models [55, 100, 191] perform candidate generation without external resources. Section 3.2.3 describes them in detail.

3.1.2. Context-mention Encoding

To correctly disambiguate an entity mention, it is crucial to thoroughly capture the information from its context. The current mainstream approach is to construct a dense contextualized vector representation of a mention y_m using an encoder neural network.

$$\text{mENC} : (C, M)^n \rightarrow (y_{m_1}, y_{m_2}, ..., y_{m_n}). \qquad (6)$$

Several early techniques in neural EL utilize a convolutional encoder [49, 127, 168, 171], as well as attention between candidate entity embeddings and embeddings of words surrounding a mention [53, 86]. However, in recent models, two approaches prevail: recurrent networks and self-attention [182].

A recurrent architecture with LSTM cells [66] that has been a backbone model for many NLP applications, is adopted to EL in [43, 62, 82, 87, 107, 129, 164] inter alia. Gupta et al. [62] concatenate outputs of two LSTM networks that independently encode left and right contexts of a mention (including the mention itself). In the same vein, Sil et al. [164] encode left and right local contexts via LSTMs but also pool the results across all mentions in a coreference chain and postprocess left and right representations with a tensor network. A modification of LSTM – GRU [29] – is used by Eshel et al. [40] in conjunction with an attention mechanism [7] to encode left and right context of a mention. Kolitsas et al. [82] represent an entity mention as a combination of LSTM hidden states included in the mention span. Le and Titov [87] simply run a bidirectional LSTM network on words complemented with embeddings of word positions relative to a target mention. Shahbazi et al. [158] adopt pretrained ELMo [138] for mention encoding by averaging mention word vectors.

Encoding methods based on self-attention have recently become ubiquitous. The EL models presented in [25, 100, 139, 191, 198] and others rely on the outputs from pre-trained BERT layers [36] for context and mention encoding. In Peters et al. [139], a men-

tion representation is modeled by pooling over word pieces in a mention span. The authors also put an additional self-attention block over all mention representations that encode interactions between several entities in a sentence. Another approach to modeling mentions is to insert special tags around them and perform a reduction of the whole encoded sequence. Wu et al. [191] reduce a sequence by keeping the representation of the special pooling symbol '[CLS]' inserted at the beginning of a sequence. Logeswaran et al. [100] mark positions of a mention span by summing embeddings of words within the span with a special vector and using the same reduction strategy as Wu et al. [191]. Yamada et al. [198] concatenate text with all mentions in it and jointly encode this sequence via a self-attention model based on pre-trained BERT. In addition to the simple attention-based encoder of Ganea and Hofmann [53], Chen et al. [25] leverage BERT for capturing type similarity between a mention and an entity candidate. They replace mention tokens with a special "[MASK]" token and extract the embedding generated for this token by BERT. A corresponding entity representation is generated by averaging multiple embeddings of mentions.

3.1.3. Entity Encoding

To make EL systems robust, it is essential to construct distributed vector representations of entity candidates y_e in such a way that they capture semantic relatedness between entities in various aspects.

$$\text{eENC} : E^k \rightarrow (y_{e_1}, y_{e_2}, ..., y_{e_k}). \qquad (7)$$

For instance, in Figure 4, the most similar entities for *Scott Young* in the Scott_Young_(American_football) sense are related to American football, whereas the Scott_Young_(writer) sense is in the proximity of writer-related entities.

There are three common approaches to entity encoding in EL: (1) entity representations learned using unstructured texts and algorithms like word2vec [110] based on co-occurrence statistics and developed originally for embedding words; (2) entity representations constructed using relations between entities in KGs and various graph embedding methods; (3) training a full-fledged neural encoder to convert textual descriptions of entities and/or other information into embeddings.

In the first category, Ganea and Hofmann [53] collect entity-word co-occurrences statistics from two sources: entity description pages from Wikipedia; text
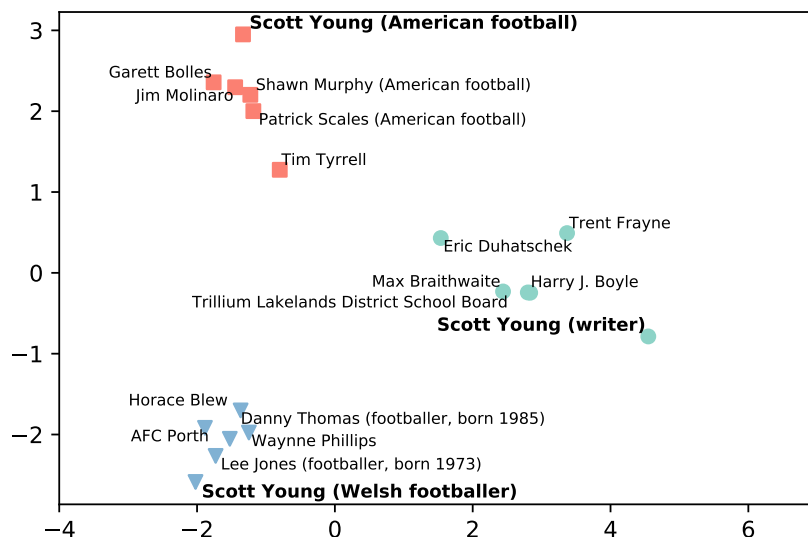
Fig. 4. **Visualization of entity embeddings.** Entity embedding space for entities related to the ambiguous entity mention "Scott Young". Three candidate entities from Wikipedia are illustrated. For each entity, their most similar 5 entities are shown in the same colors. Entity embeddings are visualized with PCA, which is utilized to reduce dimensionality (in this example, to 2D), using pre-trained embeddings provided by Yamada et al. [197][9].

surrounding anchors of hyperlinks to Wikipedia pages of corresponding entities. They train entity embeddings using the max-margin objective that exploits the negative sampling approach like in the word2vec model, so vectors of co-occurring words and entities lie closer to each other compared to vectors of random words and entities. Some other methods directly replace or extend mention annotations (usually anchor text of a hyperlink) with an entity identifier and straightforwardly train on the modified corpus a word representation model like word2vec [114, 176, 195, 210, 211]. In [53, 114, 125, 176], entity embeddings are trained in such a way that entities become embedded in the same semantic space as words or texts (i.e., sentences and paragraphs) [195]. For example, Newman-Griffis et al. [125] propose a distantly-supervised method that expands the word2vec objective to jointly learn words and entity representations in the shared space. The authors leverage distant supervision from terminologies that map entities to their surface forms (e.g. Wikipedia page titles and redirects or terminology from UMLS [12]).

In the second category of entity encoding methods that use relations between entities in a KG, Huang et al. [70] train a model that generates dense entity representations from sparse entity features (e.g. entity relations, descriptions) based on the entity relatedness. Several works expand their entity relatedness objective with functions that align words (or mentions) and entities in a unified vector space [19, 42, 144, 162, 194, 197], just like the methods from the first category. For example, Yamada et al. [194] jointly optimize three objectives to learn word and entity representations: prediction of neighbor words for the given target word, prediction of neighbor entities for the target entity based on the relationships in a KG, and prediction of neighbor words for the given entity.

Recently, knowledge graph embedding has become a prominent technique and facilitated solving various NLP and data mining tasks [187] from KG completion [15, 122, 189] to entity classification [128]. For entity linking, two major graph embedding algorithms are widely adopted: DeepWalk [136] and TransE [15].

The goal of the DeepWalk [136] algorithm is to produce embeddings of vertices that preserve their proximity in a graph [58]. It first generates several random walks for each vertex in a graph. The generated

---

[9]We used the English 100D embeddings from https://wikipedia2vec.github.io/wikipedia2vec/pretrained

walks are used as training data for the skip-gram algorithm. Like in word2vec for language modeling, given a vertex, the algorithm maximizes the probabilities of its neighbors in the generated walks. Parravicini et al. [135], Sevgili et al. [156] leverage DeepWalk-based graph embeddings built from DBpedia [92] for entity linking. Parravicini et al. [135] use entity embeddings to compute cosine similarity scores of candidate entities in global entity linking. Sevgili et al. [156] show that combining graph and text-based embeddings can slightly improve the performance of entity linking when compared to using only text-based embeddings.

The goal of the TransE [15] algorithm is to construct embeddings of both vertices and relations in such a way that they are compatible with the facts in a KG [187]. Consider the facts in a KG are represented in the form of triples (i.e. head entity, relation, tail entity). If a fact is contained in a KG, the TransE margin-based ranking criterion facilitates the presence of the following correspondence between embeddings: $head + relation \approx tail$. This means that the relationship in a KG should be a linear translation in the embedding space of entities. At the same time, if there is no such fact in a KG, this functional relationship should not hold. The TransE-based entity representations constructed from Wikidata [184] and Freebase [14] have been used for entity representation in language modeling [206] and in several works on EL [9, 124, 168]. Banerjee et al. [9], Sorokin and Gurevych [168] utilize Wikidata-based entity embeddings as an input component of neural models along with other types of information about entities. The ablation study conducted by Banerjee et al. [9] show that the TransE entity embeddings are the most important features for their entity linking model. They attribute this finding to the fact that graph embeddings contain rich information about the KG structure. Similarly, Sorokin and Gurevych [168] find that without KG structure information, their entity linker experiences a big performance drop. Nedelchev et al. [124] integrate knowledge graph embeddings built from Freebase and word embeddings in a single end-to-end model that solves entity and relation linking tasks jointly. The quantitative analysis shows that their KG-embedding-based method helps to pick correct entity candidates. Recently, Wu et al. [190] also utilize TransE embeddings with other types of entity embeddings, like Ganea and Hofmann [53] or dynamic representation, to compute pairwise entity relatedness scores.

There are many other techniques for KG embedding: [35, 59, 128, 175, 189, 199] inter alia and very

recent 5*E [123], which is designed to preserve complex graph structures in the embedding space. However, they are not widely used in entity linking right now. A detailed overview of all graph embedding algorithms is out of the scope of the current work. We refer the reader to the previous surveys on this topic [18, 58, 154, 187] and consider integration of novel KG embedding techniques in EL models a promising research direction.

In the last category, we place methods that produce entity representations using other types of information like entity descriptions and entity types. Often, an entity encoder is a full-fledged neural network, which is a part of an entity linking architecture. Sun et al. [171] use a neural tensor network to encode interactions between surface forms of entities and their category information from a KG. In the same vein, Francis-Landau et al. [49] and Nguyen et al. [127] construct entity representations by encoding titles and entity description pages with convolutional neural networks. In addition to a convolutional encoder for entity descriptions, Gupta et al. [62] also include an encoder for fine-grained entity types by using the type set of FIGER [96]. Gillick et al. [55] construct entity representations by encoding entity page titles, short entity descriptions, and entity category information with feed-forward networks. Le and Titov [87] use only entity type information from a KG and a simple feed-forward network for entity encoding. Hou et al. [69] also leverage entity types. However, instead of relying on existing type sets like in [62], they construct custom fine-grained semantic types using words from starting sentences of Wikipedia pages. To represent entities, they first average the word vectors of entity types and then linearly aggregate them with embeddings of Ganea and Hofmann [53].

Recent works leverage deep language models like BERT [36] or ELMo [138] for encoding entities. Nie et al. [129] use an architecture based on a recurrent network for obtaining entity representations from Wikipedia entity description pages. Subsequently, several models adopt BERT for the same purpose [100, 191] inter alia. Yamada et al. [198] propose a masked entity prediction task, where a model based on the BERT architecture learns to predict randomly masked input entities. This task makes the model learn also how to generate entity representations along with standard word representations. Shahbazi et al. [158] introduce E-ELMo that extends the ELMo model [138] with an additional objective. The model is trained in a multi-task fashion: to predict next/previous words,

as in a standard bidirectional language model, and to predict the target entity when encountering its mentions. As a result, besides the model for mention encoding, entity representations are obtained. Mulang' et al. [117] use bidirectional Transformers to jointly encode context of a mention, a candidate entity name, and multiple relationships of a candidate entity from a KG verbalized into textual triples: "[subject] [predicate] [object]". The input sequence of the encoder is composed simply by appending all these types of information delimited by a special separator token.

3.1.4. Entity Ranking

The goal of this stage is given a list of entity candidates $(e_1, e_2, ..., e_k)$ from a KG and a context C with a mention M to rank these entities assigning a score to each of them, as in Equation 8, where n is a number of entity mentions in a document, k is a number of candidate entities. Figure 5 depicts the typical architecture of the ranking component.

$$\text{RNK} : ((e_1, e_2, ..., e_k), C, M)^n \rightarrow \mathbb{R}^{n \times k}. \qquad (8)$$

The mention representation $\mathbf{y}_m$ generated in the mention encoding step is compared with candidate entity representations $\mathbf{y}_{e_i} (i = 1, 2, \ldots, k)$ according to the similarity measure $s(m, e_i)$. Entity representations can be pre-trained (see Section 3.1.3) or generated by another encoder as in some zero-shot approaches (see Section 3.2.3). The BERT-based model of Yamada et al. [198] simultaneously learns how to encode mentions and entity embeddings in the unified architecture.

Most of the state-of-the-art studies compute similarity $s(m, e)$ between representations of a mention m and an entity e using a dot product as in [53, 62, 82, 139, 191]:

$$s(m, e_i) = \mathbf{y}_m \cdot \mathbf{y}_{e_i}; \qquad (9)$$

or cosine similarity as in [49, 55, 171]:

$$s(m, e_i) = \cos(\mathbf{y}_m, \mathbf{y}_{e_i}) = \frac{\mathbf{y}_m \cdot \mathbf{y}_{e_i}}{\|\mathbf{y}_m\| \cdot \|\mathbf{y}_{e_i}\|}. \qquad (10)$$

The final disambiguation decision is inferred via a probability distribution $P(e_i|m)$, which is usually approximated by a softmax function over the candidates. The calculated similarity score or probability can be combined with mention-entity priors obtained during the candidate generation phase [49, 53, 82] or other features $f(e_i, m)$ such as various similarities, a string matching indicator, and entity types or type similarity

[25, 49, 157, 158, 164, 200]. One of the common techniques for that is to use an additional one or two-layer feedforward network $\phi(\cdot, \cdot)$ [49, 53, 158]. The obtained local similarity score $\Phi(e_i, m)$ or the probability distribution can be further utilized for global scoring (see Section 3.2.2).

$$P(e_i|m) = \frac{\exp(s(m, e_i))}{\sum_{i=1}^{k} \exp(s(m, e_i))}. \qquad (11)$$

$$\Phi(e_i, m) = \phi(P(e_i|m), f(e_i, m)). \qquad (12)$$

There are several approaches to framing a training objective in the literature on EL. Consider that we have k candidates for the target mention m, one of which is a true entity e_*. In some works, the models are trained with the standard negative log-likelihood objective like in classification tasks [100, 191]. However, instead of classes, negative candidates are used:

$$\mathcal{L}(m) = -s(m, e_*) + \log \sum_{i=1}^{k} \exp(s(m, e_i)). \qquad (13)$$

Instead of the the negative log-likelihood, some works use variants of a ranking loss. The idea behind such an approach is to enforce a positive margin $\gamma > 0$ between similarity scores of mentions to positive and negative candidates [53, 82, 139]:

$$\mathcal{L}(m) = \sum_i \ell(e_i, m), \text{ where} \qquad (14)$$

$$\ell(e_i, m) = [\gamma - \Phi(e_*, m) + \Phi(e_i, m)]_+. \qquad (15)$$

or

$$\ell(e_i, m) = \begin{cases} [\gamma - \Phi(e_i, m)]_+, & \text{if } e_i \text{ equal } e_* \\ [\Phi(e_i, m)]_+, & \text{otherwise.} \end{cases} \qquad (16)$$

3.1.5. Unlinkable Mention Prediction

The referent entities of some mentions can be absent in the KGs, e.g. there is no Wikipedia entry about *Scott Young* as a cricket player of the Stenhousemuir cricket club.[10] Therefore, an EL system should be able

---

[10]Information about *Scott Young* as a cricket player: https://www.stenhousemuircricketclub.com/teams/171906/player/scott-young-1828009

Local scores

Feed-forward Neural Network ← Mention-entity prior, ...

Similarity:
- Dot product
- Cosine, ...

Mention vector

Candidate entity vectors

Context-mention Encoder:
- LSTM
- BERT, ...

Entity Encoder:
- Graph embedding model
- Neural network encoder, ...

Mention   Context

Scott_Young_(American_football)
Scott_Young_(writer)
Scott_Young_(Welsh_footballer)

"**Scott Young** played for the Cleveland Browns."
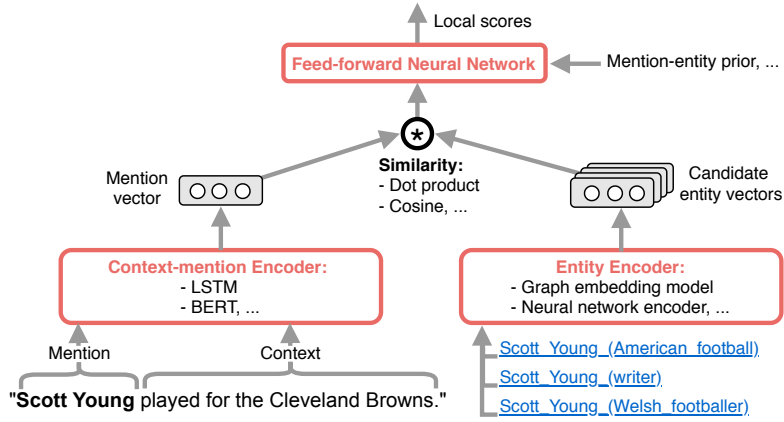
Fig. 5. **Entity ranking**. A generalized entity candidate ranking neural architecture: entity candidates are ranked according their appropriateness for a particular mention in the current context.

3.2 - Modifications of the General Architecture

3.2.1 - Joint Entity Mention Detection and Disambiguation Architecture

→ candidate based [82, 139]

→ multitask learning [107]

→ sequence labeling [17]

3.2.2 - Global Context Architecture

→ graph based [20, 211]

→ maximization of CRF potentials [53, 85]

→ sequential decision task [43, 198, 200]

→ others [42, 82]

3.2.3 - Domain Independent Architecture

→ distant learning [86, 87]

→ zero-shot [55, 100, 191]

3.2.4 - Cross-lingual Architecture

→ representation based [134, 176]
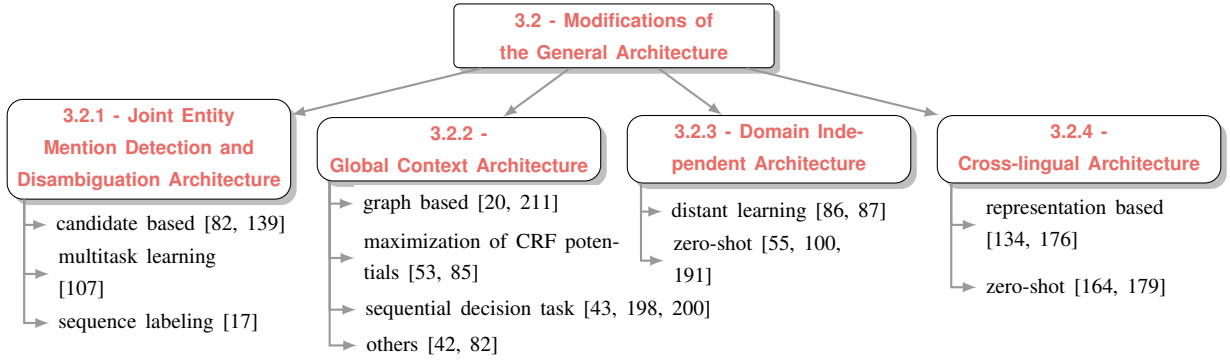
→ zero-shot [164, 179]

Fig. 6. **Reference map of the modifications of the general architecture for neural EL.** The categorization of each modification with various design choices and example references illustrating each choice. Sections 3.2.3 and 3.2.4 are categorized based on their EL solutions, here.

to predict the absence of a reference if a mention appears in specific contexts, which is known as the NIL prediction task:

$$\mathsf{NILp} : (C, M)^n \rightarrow \{0, 1\}^n. \qquad (17)$$

The NIL prediction task is essentially a classification with a reject option [51, 64, 65]. There are four common ways to perform NIL prediction. Sometimes a candidate generator does not yield any corresponding entities for a mention; such mentions are trivially considered unlikable [164, 176]. One can set a threshold for the best linking probability (or a score), below which a mention is considered unlinkable [84, 139]. Some models introduce an additional special "NIL" entity in the ranking phase, so models can predict it as the best match for the mention [82]. It is also possible to train an additional binary classifier that accepts mention-entity pairs after the ranking phase, as well as several additional features (best linking score, whether mentions are also detected by a dedicated NER system, etc.), as input and makes the final decision about whether a mention is linkable or not [107, 114].

3.2. Modifications of the General Architecture

This section presents the most notable modifications and improvements of the general architecture of neural entity linking models presented in Section 3.1 and Figures 2 and 5. The categorization of each modification is summarized in Figure 6.

3.2.1. Joint Entity Mention Detection and Disambiguation

While it is common to separate the mention detection (cf. Equation 2) and entity disambiguation stages (cf. Equation 3), as illustrated in Figure 1, a few sys-

tems provide *joint* solutions for entity linking where entity mention detection and disambiguation are done at the same time by the same model. Formally, the task becomes to detect a mention $m_i \in M$ and predict an entity $e_i \in E$ for a given context $c_i \in C$, for all n entity mentions in the context:

$$\text{EL} : C \rightarrow (M, E)^n. \tag{18}$$

Undoubtedly, solving these two problems simultaneously makes the task more challenging. However, the interaction between these steps can be beneficial for improving the quality of the overall pipeline due to their natural mutual dependency. While first competitive models that provide joint solutions were probabilistic graphical models [102, 126], we focus on purely neural approaches proposed recently [17, 23, 33, 82, 107, 139, 142, 168].

The main difference of joint models is the necessity to produce also mention candidates. For this purpose, Kolitsas et al. [82] and Peters et al. [139] enumerate all spans in a sentence with a certain maximum width, filter them by several heuristics (remove mentions with stop words, punctuation, ellipses, quotes, and currencies), and try to match them to a pre-built index of entities used for the candidate generation. If a mention candidate has at least one corresponding entity candidate, it is further treated by a ranking neural network that can also discard it by considering it unlinkable to any entity in a KG (see Section 3.1.4). Therefore, the decision during the entity disambiguation phase affects mention detection. In a similar fashion, Sorokin and Gurevych [168] treat each token n-gram up to a certain length as a possible mention candidate. They use an additional binary classifier for filtering candidate spans, which is trained jointly with an entity linker. Banerjee et al. [9] also enumerates all possible n-grams and expands each of them with candidate entities, which results in a long sequence of points corresponding to a candidate entity for a particular mention n-gram. This sequence is further processed by a single-layer BiLSTM pointer network [183] that generates index numbers of potential entities in the input sequence. Li et al. [94] consider various possible spans as mention candidates and introduce a loss component for boundary detection, which is optimized along with the loss for disambiguation.

Martins et al. [107] describe the approach with tighter integration between detection and linking phases via multi-task learning. The authors propose a stack-based bidirectional LSTM network with a shift-reduce

mechanism and attention for entity recognition that propagates its internal states to the linker network for candidate entity ranking. The linker is supplemented with a NIL predictor network. The networks are trained jointly by optimizing the sum of losses from all three components.

Broscheit [17] goes further by suggesting a completely end-to-end method that deals with mention detection and linking jointly without explicitly executing a candidate generation step. In this work, the EL task is formulated as a sequence labeling problem, where each token in the text is assigned an entity link or a NIL class. They leverage a sequence tagger based on pre-trained BERT for this purpose. This simplistic approach does not supersede [82] but outperforms the baseline, in which candidate generation, mention detection, and linking are performed independently. In the same vein, Chen et al. [23] use a sequence tagging framework for joint entity mention detection and disambiguation. However, they experiment with both settings: when a candidate list is available and not, and demonstrate that it is possible to achieve high linking performance without candidate sets. Similar to Li et al. [94], they optimize the joint loss for linking and mention boundary detection.

Poerner et al. [142] propose a model E-BERT-MLM, in which they repurpose the masked language model (MLM) objective for the selection of entity candidates in an end-to-end EL pipeline. The candidate mention spans and candidate entity sets are generated in the same way as in [82]. For candidate selection, E-BERT-MLM inserts a special "[E-MASK]" token into the text before the considered candidate mention span and tries to restore an entity representation for it. The model is trained by minimizing the cross-entropy between the generated entity distribution of the potential spans and gold entities. In addition to the standard BERT architecture, the model contains a linear transformation pre-trained to align entity embeddings with embeddings of word-piece tokens.

De Cao et al. [33] recently have proposed a generative approach to performing mention detection and disambiguation jointly. Their model, which is based on BART [93], performs a sequence-to-sequence autoregressive generation of text markup with information about mention spans and links to entities in a KG. The generation process is constrained by a markup format and a candidate set, which is retrieved from standard pre-built candidate resources. Most of the time, the network works in a copy-paste regime when it copies input tokens into the output. When it finds a begin-
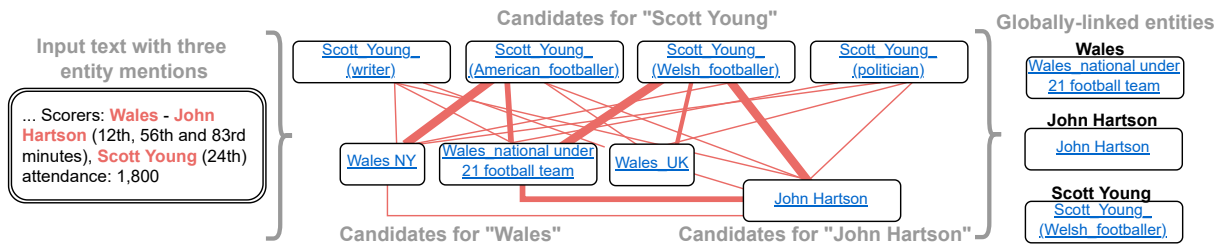
Fig. 7. **Global entity disambiguation**. The global entity linking resolves all mentions simultaneously based on entity coherence. Bolder lines indicate expected higher degrees of entity-entity similarity.

ning of a mention, the model marks it with a square bracket, copies all tokens of a mention, adds a finishing square bracket, and generates a link to an entity. Although this approach to EL, at the first glance, is counterintuitive and completely different from the solutions with a standard bi-encoder architecture, this model achieves near state-of-the-art results for joint MD and ED and competitive performances on ED-only benchmarks. However, as it is shown in the paper, to achieve such impressive results, the model had to be pre-trained on a large annotated Wikipedia-based dataset [191]. The authors also note that the memory footprint of the proposed model is much smaller than that of models based on the standard architecture due to no need for storing entity embeddings.

3.2.2. Global Context Architectures

Two kinds of contextual information are available in entity disambiguation: local and global. In local approaches to ED, each mention is disambiguated independently based on the surrounding words, as in the following function:

$$\text{LED} : (M, C) \rightarrow E. \tag{19}$$

Global approaches to ED take into account semantic consistency (coherence) across multiple entities in a context. In this case, all q entity mentions in a group are disambiguated interdependently: a disambiguation decision for one entity is affected by decisions made for other entities in a context as illustrated in Figure 7 and Equation 20.

$$\text{GED} : ((m_1, m_2, ..., m_q), C) \rightarrow E^q. \tag{20}$$

In the example presented in Figure 7, the consistency score between correct entity candidates: the *national football team* sense of *Wales* and the *Welsh footballer* sense of *Scott Young* and *John Hartson*, is expected to be higher than between incorrect ones.

Besides involving consistency, the considered context of a mention in global methods is usually larger than in local ones or even extends to the whole document. Although modeling consistency between entities and the extra information of the global context improves the disambiguation accuracy, the number of possible entity assignments is combinatorial [54], which results in high time complexity of disambiguation [53, 200]. Another difficulty is an attempt to assign an entity its consistency score since this score is not possible to compute in advance due to the simultaneous disambiguation [194].

The typical approach to global disambiguation is to generate a graph including candidate entities of mentions in a context and perform some graph algorithms, like random walk algorithms (e.g. PageRank [133]) or graph neural networks, over it to select highly consistent entities [61, 137, 210, 211]. Recently, Xue et al. [192] propose a neural recurrent random walk network learning algorithm based on the transition matrix of candidate entities containing relevance scores, which are created from hyperlinks information and cosine similarity of entities. Cao et al. [20] construct a subgraph from the candidates of neighbor mentions, integrate local and global features of each candidate, and apply a graph convolutional network over this subgraph. In this approach, the graph is static, which would be problematic in such cases that two mentions would co-occur in different documents with different topics, however, the produced graphs will be the same, and so, could not catch the different information [190]. To address it, Wu et al. [190] propose a dynamic graph convolution architecture, where entity relatedness scores are computed and updated in each layer based on the previous layer information (initialized with some features, including context scores) and entity similarity scores. Globerson et al. [56] introduce a model with an attention mechanism that takes into account only the subgraph of the target mention, rather

than all interactions of all the mentions in a document and restrict the number of mentions with an attention.

Some works approach global ED by maximizing the Conditional Random Field (CRF) potentials, where the first component Ψ represents a local entity-mention score, and the other component Φ measures coherence among selected candidates [53, 54, 85, 86], as defined in Ganea and Hofmann [53]:

$$g(e,m,c) = \sum_{i=1}^{n} \Psi(e_i, m_i, c_i) + \sum_{i<j} \Phi(e_i, e_j). \quad (21)$$

However, model training and its exact inference are NP-hard. Ganea and Hofmann [53] utilize truncated fitting of loopy belief propagation [54, 56] with differentiable and trainable message passing iterations using pairwise entity scores to reduce the complexity. Le and Titov [85] expand it in a way that pairwise scores take into account relations of mentions (e.g. located_in, or coreference: the mentions are coreferent if they refer to the same entity) by modeling relations between mentions as latent variables. Shahbazi et al. [157] develop a greedy beam search strategy, which starts from a locally optimal initial solution and is improved by searching for possible corrections with the focus on the least confident mentions.

Despite the optimizations proposed like in some aforementioned works, taking into account coherence scores among candidates of all mentions at once can be prohibitively slow. It also can be malicious due to erroneous coherence among wrong entities [43]. For example, if two mentions have coherent erroneous candidates, this noisy information may mislead the final global scoring. To resolve this issue, some studies define the global ED problem as a sequential decision task, where the disambiguation of new entities is based on the already disambiguated ones with high confidence. Fang et al. [43] train a policy network for sequential selection of entities using reinforcement learning. The disambiguation of mentions is ordered according to the local score, so the mentions with high confident entities are resolved earlier. The policy network takes advantage of output from the LSTM global encoder that maintains the information about earlier disambiguation decisions. Yang et al. [200] also utilize reinforcement learning for mention disambiguation. They use an attention model to leverage knowledge from previously linked entities. The model dynamically selects the most relevant entities for the target mention and calculates the coherence scores. Ya-

mada et al. [198] iteratively predict entities for yet unresolved mentions with a BERT model, while attending on the previous most confident entity choices. Similarly, Gu et al. [60] sort mentions based on their ambiguity degrees produced by their BERT-based local model and update query/context based on the linked entities so that the next prediction can leverage the previous knowledge. They also utilize a gate mechanism to control historical cues – representations of linked entities. Yamada et al. [194] and Radhakrishnan et al. [144] measure the similarity first based on unambiguous mentions and then predict entities for complex cases. Nguyen et al. [127] use an RNN to implicitly store information about previously seen mentions and corresponding entities. They leverage the hidden states of the RNN to reach this information as a feature for the computation of the global score. Tsai and Roth [176] directly use embeddings of previously linked entities as features for the disambiguation model. Recently, Fang et al. [44] combine sequential approaches with graph based methods, where the model dynamically changes the graph depending on the current state. The graph is constructed with previously resolved entities, current candidate entities, and subsequent mention's candidates. The authors use a graph attention network over this graph to make a global scoring. As explained before, Wu et al. [190] also change the entity graph dynamically depending on the outputs from previous layers of a GCN. Zwicklbauer et al. [211] include to the candidates graph a topic node created from the set of already disambiguated entities.

Some studies, for example, Kolitsas et al. [82] model the coherence component as an additional feedforward neural network that uses the similarity score between the target entity and an average embedding of the candidates with a high local score. Fang et al. [42] use the similarity score between the target entity and its surrounding entity candidates in a specified window as a feature for the disambiguation model.

Another approach that can be considered as global is to make use of a document-wide context, which usually contains more than one mention and helps to capture the coherence implicitly instead of explicitly designing an entity coherence component [49, 62, 114, 139].

3.2.3. Domain-Independent Architectures

Domain independence is one of the most desired properties of EL systems. Annotated resources are very limited and exist only for a few domains. Obtaining labeled data in a new domain requires much la-

bor. Earlier, this problem is tackled by few domain-independent approaches based on unsupervised [19, 125, 186] and semi-supervised models [84]. Recent studies provide solutions based on distant learning and zero-shot methods.

Le and Titov [86, 87] propose distant learning techniques that use only unlabeled documents. They rely on the weak supervision coming from a surface matching heuristic, and the EL task is framed as binary multi-instance learning. The model learns to distinguish between a set of positive entities and a set of random negatives. The positive set is obtained by retrieving entities with a high word overlap with the mention and that have relations in a KG to candidates of other mentions in the sentence. While showing promising performance, which in some cases rivals results of fully supervised systems, these approaches require either a KG describing relations of entities [87] or mention-entity priors computed from entity hyperlink statistics extracted from Wikipedia [86].

Recently proposed zero-shot techniques [100, 173, 191, 201] tackle problems related to adapting EL systems to new domains. In the zero-shot setting, the only entity information available is its description. As well as in other settings, texts with mention-entity pairs are also available. The key idea of zero-shot methods is to train an EL system on a domain with rich labeled data resources and apply it to a new domain with only minimal available data like descriptions of domain-specific entities. One of the first studies that proposes such a technique is Gupta et al. [62] (not purely zero-shot because they also use entity typings). Existing zero-shot systems do not require such information resources as surface form dictionaries, prior entity-mention probabilities, KG entity relations, and entity typing, which makes them particularly suited for building domain-independent solutions. However, the limitation of information sources raises several challenges.

Since only textual descriptions of entities are available for the target domain, one cannot rely on pre-built dictionaries for candidate generation. All zero-shot works rely on the same strategy to tackle candidate generation: pre-compute representations of entity descriptions (sometimes referred to as caching), compute a representation of a mention, and calculate its similarity with all the description representations. Pre-computed representations of descriptions save a lot of time at the inference stage. Particularly, Logeswaran et al. [100] use the BM25 information retrieval formula [78], which is a similarity function for count-based representations.

A natural extension of count-based approaches is embeddings. The method proposed by Gillick et al. [55], which is a predecessor of zero-shot approaches, uses average unigram and bigram embeddings followed by dense layers to obtain representations of mentions and descriptions. The only aspect that separates this approach from pure zero-shot techniques is the usage of entity categories along with descriptions to build entity representations. Cosine similarity is used for the comparison of representations. Due to the computational simplicity of this approach, it can be used in a single stage fashion where candidate generation and ranking are identical. For further speedup, it is possible to make this algorithm two-staged. In the first stage, an approximate search can be used for candidate set retrieval. In the second stage, the retrieved smaller set can be used for exact similarity computation. Instead of simple embeddings, Wu et al. [191] suggest using a BERT-based bi-encoder for candidate generation. Two separate encoders generate representations of mentions and entity descriptions. Similar to the previous work, the candidate selection is based on the score obtained via a dot-product of mention/entity representations.

For entity ranking, a very simple embedding-based approach of Gillick et al. [55] described above shows very competitive scores on the TAC KBP-2010 benchmark, outperforming some complex neural architectures. The recent studies of Logeswaran et al. [100] and Wu et al. [191] utilize a BERT-based cross-encoder to perform joint encoding of mentions and entities. The cross-encoder takes a concatenation of a context with a mention and an entity description to produce a scalar score for each candidate. The cross-attention helps to leverage the semantic information from the context and the definition on each layer of the encoder network [71, 150]. In both studies, cross-encoders achieve superior results compared to bi-encoders and count-based approaches. For entity linking, cross-attention between mention context representations and entity descriptions is also used by Nie et al. [129]. However, they leverage recurrent architectures for encoding. Yao et al. [201] introduce a small tweak of positional embeddings in the Logeswaran et al. [100]'s architecture aimed at better handling long contexts. Tang et al. [173] address the problem of the limited size of the mention context and the entity description that could be processed by the standard BERT model. They argue that the input size of 512 tokens is not enough to capture context and entity description relatedness since the evidence for linking could scatter in differ-

ent paragraphs and suggest a novel architecture that resolves this problem. Roughly speaking, their model splits the context of a mention and entity description into multiple paragraphs, performs cross-attention between representations of these paragraphs, and aggregates the results for disambiguation. The experimental results show that their model substantially improves the zero-shot performance keeping the inference time in an acceptable range.

Evaluation of zero-shot systems requires data from different domains. Logeswaran et al. [100] proposes the *Zero-shot EL*[11] dataset, constructed from several Wikias[12]. In the proposed setting, training is performed on one set of Wikias while evaluation is performed on others. Gillick et al. [55] construct the Wikinews dataset. This dataset can be used for evaluation after training on Wikipedia data.

Clearly, heavy neural architectures pre-trained on general-purpose open corpora substantially advance the performance of zero-shot techniques. As highlighted by Logeswaran et al. [100] further unsupervised pre-training on source data, as well as on the target data is beneficial. The development of better approaches to the utilization of unlabeled data might be a fruitful research direction. Furthermore, closing the performance gap of entity ranking between a fast representation based bi-encoder and a computationally intensive cross-encoder is an open question.

3.2.4. Cross-lingual Architectures

An abundance of labeled data for EL in English contrasts with the amount of data available in other languages. The cross-lingual EL (sometimes called XEL) methods [76] aim at overcoming the lack of annotation for resource-poor languages by leveraging supervision coming from their resource-rich counterparts. Many of these methods are feasible due to the presence of a unique source of supervision for EL – Wikipedia, which is available for a variety of languages. The inter-language links in Wikipedia that map pages in one language to equivalent pages in another language also help to map corresponding entities in different languages.

Challenges in XEL start at candidate generation and mention detection steps since a resource-poor language can lack mappings between mention strings and entities. In addition to the standard mention-entity priors based on inter-language links [164, 176, 179], can-

didate generation can be approached by mining a translation dictionary [134], training a translation and alignment model [177, 180], or applying a neural character-level string matching model [151, 207]. In the latter approach, the model is trained to match strings from a high-resource pivot language to strings in English. If a high-resource pivot language is similar to the target low-resource one, such a model is able to produce reasonable candidates for the latter. The neural string matching approach can be further improved with simpler average n-gram encoding and extending entity-entity pairs with mention-entity examples [208]. Such an approach can also be applied to entity recognition [31]. Fu et al. [50] criticize methods that solely rely on Wikipedia due to the lack of inter-language links for resource-poor languages. They propose a candidate generation method that leverages results from querying online search engines (Google and Google Maps) and show that due to its much higher recall compared to other methods, it is possible to substantially increase the performance of XEL.

There are several approaches to candidate ranking that take advantage of cross-lingual data for dealing with the lack of annotated examples. Pan et al. [134] use the Abstract Meaning Representation (AMR) [8] statistics in English Wikipedia and mention context for ranking. To train an AMR tagger, pseudo-labeling [89] is used. Tsai and Roth [176] train monolingual embeddings for words and entities jointly by replacing every entity mention with corresponding entity tokens. Using the inter-language links, they learn the projection functions from multiple languages into the English embedding space. For ranking, context embeddings are averaged, projected into the English space, and compared with entity embeddings. The authors demonstrate that this approach helps to build better entity representations and boosts the EL accuracy in the cross-lingual setting by more than 1% for Spanish and Chinese. Sil et al. [164] propose a method for zero-shot transfer from a high-resource language. The authors extend the previous approach with the least squares objective for embedding projection learning, the CNN context encoder, and a trainable re-weighting of each dimension of context and entity representations. The proposed approach demonstrates improved performance as compared to previous non-zero-shot approaches. Upadhyay et al. [179] argues that the success of zero-shot cross-lingual approaches [164, 176] might be largely originating from a better estimation of mention-entity prior probabilities. Their approach extends [164] with global context information and in-

---

[11] https://github.com/lajanugen/zeshel
[12] https://www.wikia.com

corporation of typing information into context and entity representations (the system learns to predict typing during the training). The authors report a significant drop in performance for zero-shot cross-lingual EL without mention-entity priors, while showing state-of-the-art results with priors. They also show that training on a resource-rich language might be very beneficial for low-resource settings.

The aforementioned techniques of cross-lingual entity linking heavily rely on pre-trained multilingual embeddings for entity ranking. While being effective in settings with at least prior probabilities available, the performance in realistic zero-shot scenarios drops drastically. Along with the recent success of the zero-shot multilingual transfer of large pre-trained language models, this is a motivation to utilize powerful multilingual self-supervised models. Botha et al. [16] use the zeros-shot monolingual architecture of Logeswaran et al. [100], Wu et al. [191] and mBERT [141] to build a massively multilingual EL model for more than 100 languages. Their system effectively selects proper entities among almost 20 million of candidates using a bi-encoder, hard negative mining, and an additional cross-lingual entity description retrieval task. The biggest improvements over the baselines are achieved in the zero-shot and few-shot settings, which demonstrates the benefits of training on a large amount of multilingual data.

3.3. Methods that do not Fit the General Architecture

There are a few works that propose methods not fitting the general architecture presented in Figures 2 and 5. Raiman and Raiman [146] rely on the intermediate supplementary task of entity typing instead of directly performing entity disambiguation. They learn a type system in a KG and train an intermediate type classifier of mentions that significantly refines the number of candidates for the final linking model. Onoe and Durrett [131] leverage distant supervision from Wikipedia pages and the Wikipedia category system to train a fine-grained entity typing model. At test time, they use the soft type predictions and the information about candidate types derived from Wikipedia to perform the final disambiguation. The authors claim that such an approach helps to improve the domain independence of their EL system. Kar et al. [80] consider a classification approach, where each entity is considered as a separate class or a task. They show that the straightforward classification is difficult due to exceeding memory requirements. Therefore, they experiment

with multitask learning, where parameter learning is decomposed into solving groups of tasks. Globerson et al. [56] do not have any encoder components; instead, they rely on contextual and pairwise feature-based scores. They have an attention mechanism for global ED with a non-linear optimization as described in Section 3.2.2.

3.4. Summary

We summarize design features for neural EL models in Table 2 and also links to their publicly available implementations in Table 7 in Appendix A. The mention encoders have made a shift to self-attention architectures and started using deep pre-trained models like BERT. The majority of studies still rely on external knowledge for the candidate generation step. There is a surge of models that tackle the domain adaptation problem in a zero-shot fashion. However, the task of zero-shot joint entity mention detection and linking has not been addressed yet. It is shown in several works that the cross-encoder architecture is superior compared to models with separate mention and entity encoders. The global context is widely used, but there are few recent studies that focus only on local EL.

Each column in Table 2 corresponds to a model feature. The **encoder type** column presents the architecture of the mention encoder of the neural entity linking model. It contains the following options:

- n/a – a model does not have a neural encoder for mentions / contexts.
- CNN – an encoder based on convolutional layers (usually with pooling).
- Tensor net. – an encoder that uses a tensor network.
- Atten. – means that a context-mention encoder leverages an attention mechanism to highlight the part of the context using an entity candidate.
- GRU – an encoder based on a recurrent neural network and gated recurrent units [29].
- LSTM – an encoder based on a recurrent neural network and long short-term memory cells [66] (might be also bidirectional).
- FFNN – an encoder based on a simple feedforward neural network.
- ELMo – an encoder based on a pre-trained ELMo model [138].
- BERT – an encoder based on a pre-trained BERT model [36].

Table 2

Features of neural EL models. Neural entity linking models compared according to their architectural features. The description of columns is presented in the beginning of Section 3.4. The footnotes in the table are enumerated in the end of Section 3.4.

Model	Encoder Type	Global	MD+ED	NIL Pred.	Ent. Encoder Source based on	Candidate Generation	Learning Type for Disam.	Cross-lingual
Sun et al. (2015) [171]	CNN+Tensor net.				ent. specific info.	surface match+aliases	supervised	
Francis-Landau et al. (2016) [49]	CNN	✗[3]		✗	ent. specific info.	surface match+prior	supervised	
Fang et al. (2016) [42]	word2vec-based	✗			relational info.	n/a	supervised	
Yamada et al. (2016) [194]	word2vec-based	✗			relational info.	aliases	supervised	
Zwicklbauer et al. (2016b) [211]	word2vec-based	✗		✗	unstructured text + ent. specific info.	surface match	unsupervised[5]	
Tsai and Roth (2016) [176]	word2vec-based	✗		✗	unstructured text	prior	supervised	✗
Nguyen et al. (2016b) [127]	CNN	✗		✗	ent. specific info.	surface match+prior	supervised	
Globerson et al. (2016) [56]	n/a	✗			n/a	prior+aliases	supervised	
Cao et al. (2017) [19]	word2vec-based	✗			relational info.	aliases	supervised or unsupervised	
Eshel et al. (2017) [40]	GRU+Atten.				unstructured text[1]	aliases or surface match	supervised	
Ganea and Hofmann (2017) [53]	Atten.	✗			unstructured text	prior+aliases	supervised	
Moreno et al. (2017) [114]	word2vec-based	✗[3]		✗	unstructured text	surface match+aliases	supervised	
Gupta et al. (2017) [62]	LSTM	✗[3]			ent. specific info.	prior	supervised[4]	
Nie et al. (2018) [129]	LSTM+CNN	✗			ent. specific info.	surface match+prior	supervised	
Sorokin and Gurevych (2018) [168]	CNN	✗	✗		relational info.	surface match	supervised	
Shahbazi et al. (2018) [157]	Atten.	✗			unstructured text	prior+aliases	supervised	
Le and Titov (2018) [85]	Atten.	✗			unstructured text	prior+aliases	supervised	
Newman-Griffis et al. (2018) [125]	word2vec-based				unstructured text	aliases	unsupervised	
Radhakrishnan et al. (2018) [144]	n/a	✗			relational info.	aliases	supervised	
Kolitsas et al. (2018) [82]	LSTM	✗	✗		unstructured text	prior+aliases	supervised	
Sil et al. (2018) [164]	LSTM+Tensor net.			✗	ent. specific info.	prior or prior+aliases	zero-shot	✗
Upadhyay et al. (2018a) [179]	CNN	✗[3]			ent. specific info.	prior	zero-shot	✗
Cao et al. (2018) [20]	Atten.	✗			relational info.	prior+aliases	supervised	
Raiman and Raiman (2018) [146]	n/a	✗			n/a	prior+type classifier	supervised	✗
Mueller and Durrett (2018) [116]	GRU+Atten.+CNN				unstructured text[1]	surface match	supervised	
Shahbazi et al. (2019) [158]	ELMo				unstructured text	prior+aliases or aliases	supervised	
Logeswaran et al. (2019) [100]	BERT				ent. specific info.	BM25	zero-shot	
Gillick et al. (2019) [55]	FFNN				ent. specific info.	nearest neighbors	supervised[4]	
Peters et al. (2019) [139][2]	BERT	✗[3]	✗	✗	unstructured text	prior+aliases	supervised	
Le and Titov (2019b) [87]	LSTM				ent. specific info.	surface match	weakly-supervised	
Le and Titov (2019a) [86]	Atten.	✗			unstructured text	prior+aliases	weakly-supervised	
Fang et al. (2019) [43]	LSTM	✗			unstructured text + ent. specific info.	aliases	supervised	
Martins et al. (2019) [107]	LSTM		✗	✗	unstructured text	aliases	supervised	
Yang et al. (2019) [200]	Atten. or CNN	✗			unstructured text or ent. specific. info.	prior+aliases	supervised	
Xue et al. (2019) [192]	CNN	✗			ent. specific info.	prior+aliases	supervised	
Zhou et al. (2019) [207]	n/a	✗			unstructured text	prior+char.-level model	zero-shot	✗
Broscheit (2019) [17]	BERT		✗	✗	n/a	n/a	supervised	
Hou et al. (2020) [69]	Atten.	✗			ent. specific info.+ unstructured text	prior+aliases	supervised	
Onoe and Durrett (2020) [131]	ELMo+Atten. +CNN+LSTM				n/a	prior or aliases	supervised[4]	
Chen et al. (2020) [23]	BERT		✗		relational info.	n/a or aliases	supervised	
Wu et al. (2020b) [191]	BERT				ent. specific info.	nearest neighbors	zero-shot	
Banerjee et al. (2020) [9]	fastText		✗		relational info.	surface match	supervised	
Wu et al. (2020a) [190]	ELMo	✗			unstructured text+ relational info.	prior+aliases	supervised	
Fang et al. (2020) [44]	BERT	✗			ent. specific info.	surface match+aliases+ Google Search	supervised	
Chen et al. (2020) [25]	Atten.+BERT	✗			unstructured text	prior+aliases	supervised	
Botha et al. (2020) [16]	BERT				ent. specific info.	nearest neighbors	zero-shot	✗
Yao et al. (2020) [201]	BERT				ent. specific info.	BM25	zero-shot	
Li et al. (2020) [94]	BERT		✗		ent. specific info.	nearest neighbors	zero-shot	
Poerner et al. (2020) [142][2]	BERT	✗	✗	✗	relational info.	prior+aliases	supervised	
Fu et al. (2020) [50]	M-BERT				ent. specific info.	Google Search Google Maps	zero-shot	✗
Mulang' et al. (2020) [117]	Atten. or CNN or BERT	✗			relational info.	prior+aliases	supervised	
Yamada et al. (2021) [198]	BERT	✗			unstructured text	prior+aliases or aliases	supervised	
Gu et al. (2021) [60]	BERT	✗		✗	ent. specific info.	surface match+prior or aliases	supervised	
Tang et al. (2021) [173]	BERT				ent. specific info.	BM25	zero-shot	
De Cao et al. (2021) [33]	BART	✗	✗		n/a	prior+aliases	supervised	

- fastText – an encoder based on a pre-trained fast-Text model [13].
- word2vec-based – an encoder that leverages principles of CBOW or skip-gram algorithms [88, 110, 111].

Note that the theoretical complexity of various types of encoders is different. As discussed by Vaswani et al. [182], complexity per layer of self-attention is $O(n^2 \cdot d)$, as compared to $O(n \cdot d^2)$ for a recurrent layer, and $O(k \cdot n \cdot d^2)$ for a convolutional layer, where n is the length of an input sequence, d is the dimensionality, and k is the kernel size of convolutions. At the same time, the self-attention allows for a better parallelization than the recurrent networks as the number of sequentially executed operations for self-attention requires a constant number of sequentially executed operations of $O(1)$, while a recurrent layer requires $O(n)$ sequential operations. Overall, estimation of the computational complexity of training and inference of various neural networks is certainly beyond the scope of the goal of this survey. The interested reader may refer to [182] and specialized literature on this topic, e.g. [99, 132, 165].

The **global** column shows whether a system uses a global solution (see Section 3.2.2). The **MD+ED** column refers to joint entity mention detection and disambiguation models, where detection and disambiguation of entities are performed collectively (Section 3.2.1). The **NIL prediction** column points out models that also label unlinkable mentions. The **entity embedding** column presents which resource is used to train entity representations based on the categorization in Section 3.1.3, where

- n/a – a model does not have a neural encoder for entities.
- unstructured text – entity representations are constructed from unstructured text using approaches based on co-occurrence statistics developed originally for word embeddings like word2vec [110].
- relational info. – a model uses relations between entities in KGs.
- ent. specific info. – an entity encoder uses other types of information, like entity descriptions, types, or categories.

In the **candidate generation** column, the candidate generation methods are specified (Section 3.1.1). It contains the following options:

- n/a – the solution that does not have an explicit candidate generation step (e.g. the method presented by Broscheit [17]).
- surface match – surface form matching heuristics.
- aliases – a supplementary aliases for entities in a KG.
- prior – filtering candidates with pre-calculated mention-entity prior probabilities or frequency counts.
- type classifier – Raiman and Raiman [146] filter candidates using a classifier for an automatically learned type system.
- BM25 – a variant of TF-IDF to measure similarity between a mention and a candidate entity based on description pages.
- nearest neighbors – the similarity between mention and entity representations is calculated, and entities that are nearest neighbors of mentions are retrieved as candidates. Wu et al. [191] train a supplementary model for this purpose.
- Google search – leveraging Google Search Engine to retrieve entity candidates.
- char.-level model – a neural character-level string matching model.

The **learning type for disambiguation** column shows whether a model is *'supervised', 'unsupervised', 'weakly-supervised', or 'zero-shot'*. The **cross-lingual** column refers to models that provide cross-lingual EL solutions (Section 3.2.4).

In addition, the following superscript notations are used to denote specific features of methods shown as a note in the Table 2:

1. These works use only entity description pages, however, they are labeled as the first category (unstructured text) since their training method is based on principals from word2vec.
2. The authors provide EL as a subsystem of language modeling.
3. These solutions do not rely on global coherence but are marked as "global" because they use document-wide context or multiple mentions at once for resolving entity ambiguity.
4. These studies are domain-independent as discussed in Section 3.2.3.
5. Zwicklbauer et al. [211] may not be accepted as purely unsupervised since they have some thresh-

old parameters in the disambiguation algorithm tuned on a labeled set.

4. Evaluation

In this section, we present evaluation results for the entity linking and entity relatedness tasks on the commonly used datasets.

4.1. Entity Linking

4.1.1. Experimental Setup

The evaluation results are reported based on two different evaluation settings. The first setup is entity disambiguation (ED) where the systems have access to the mention boundaries. The second setup is entity mention detection and disambiguation (MD+ED) where the input for the systems that perform MD and ED jointly is only plain text. We presented their results in separate tables since the scores for the joint models accumulate the errors made during the mention detection phase.

Datasets We report the evaluation results of monolingual EL models on the English datasets widely-used in recent research publications: AIDA [67], TAC KBP 2010 [75], MSNBC [32], AQUAINT [112], ACE2004 [148], CWEB [52, 61], and WW [61]. AIDA is the most popular dataset for benchmarking EL systems. For AIDA, we report the results calculated for the test set (AIDA-B).

The cross-lingual EL results are reported for the TAC KBP 2015 [76] Spanish (es) and Chinese (zh) datasets. The descriptive statistics of the datasets and their text genres are presented in Table 3 according to information reported in [39, 53, 75, 76, 191].

Evaluation Metrics For the ED setting, we present micro F1 or accuracy scores reported by model authors. We note that, since mentions are provided as an input, the number of mentions predicted by the model is equal to the number of mentions in the ground truth [160], so micro F1, precision, recall, and accuracy scores are equal in this setting as explained in Shen et al. [160]:

$$F1 = Acc = \frac{\#\ correctly\ disamb.\ mentions}{\#\ total\ mentions}.$$
$$(22)$$

For the MD+ED setting, where joint models are evaluated, we report micro F1 scores based on strong annotation matching. The formulas to compute F1 scores are shown below, as described in Shen et al. [160] and Ganea et al. [54]:

$$P = \frac{\#\ correctly\ detected\ and\ disamb.\ mentions}{\#\ predicted\ mentions\ by\ model},$$
$$(23)$$

$$R = \frac{\#\ correctly\ detected\ and\ disamb.\ mentions}{\#\ mentions\ in\ ground\ truth},$$
$$(24)$$

$$F1 = \frac{2 \cdot P \cdot R}{P + R}.$$
$$(25)$$

We note that results reported in multiple considered papers are usually obtained using GERBIL [153] − a platform for benchmarking EL models. It implements various experimental setups, including entity disambiguation denoted as D2KB and a combination of mention detection and disambiguation denoted as A2KB. GERBIL encompasses many evaluation datasets in a standartized way along with annotations and provides the computation of evaluation metrics, i.e. micro-macro precision, recall, and F-measure.

Baseline Models While our goal is to perform a survey of neural EL systems, we also report results of several indicative and prominent classic non-neural systems as baselines to underline the advances yielded by neural models. More specifically, we report results of DBpedia Spotlight (2011) [108], AIDA (2011) [67], Ratinov et al. (2011) [148], WAT (2014) [140], Babelfy (2014) [115], Lazic et al. (2015) [84], Chisholm and Hachey (2015) [27], and PBOH (2016) [54].

For considered neural EL systems, we present the best scores reported by the authors. For the baseline systems, the results are reported according to Kolitsas et al. [82][13] and Ganea and Hofmann [53].

4.1.2. Discussion of Results

Entity Disambiguation Results We start our discussion of the results from the entity disambiguation (ED) models, for which mention boundaries are provided. Figure 8 shows how the performance of the entity disambiguation models on the most widely-used dataset AIDA improved during the course of the last decade

---

[13]Some of the baseline scores are presented in the appendix of [82], which is available at https://arxiv.org/pdf/1808.07699.pdf

Table 3

Evaluation datasets. Descriptive statistics of the evaluation datasets used in this survey to compare the EL models. The values for MSNBC, AQUAINT, and ACE2004 datasets are based on the update by Guo and Barbosa [61]. The statistics for AIDA-B, MSNBC, AQUAINT, ACE2004, CWEB, and WW is reported according to [53] (# of mentions takes into account only non-NIL entity references). The TAC KBP dataset statistics is reported according to [39, 75, 76, 191] (# of mentions takes into account also NIL entity references).

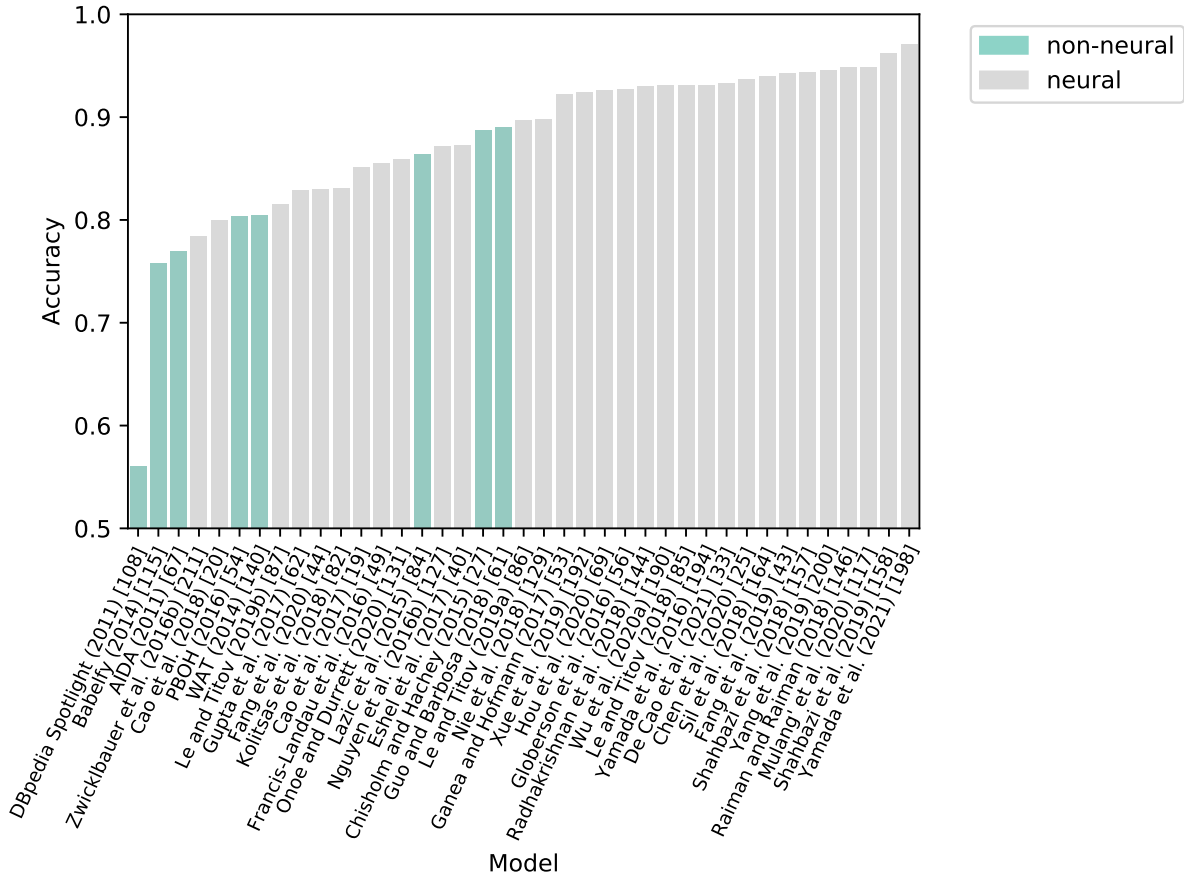| Corpus	Text Genre	# of Documents	# of Mentions
AIDA-B [67]	News	231	4,485
MSNBC [32]	News	20	656
AQUAINT [112]	News	50	727
ACE2004 [148]	News	36	257
CWEB [52, 61]	Web & Wikipedia	320	11,154
WW [61]	Web & Wikipedia	320	6,821
TAC KBP 2010 [75]	News & Web	2,231	2,250
TAC KBP 2015 Chinese [76]	News & Forums	166	11,066
TAC KBP 2015 Spanish [76]	News & Forums	167	5,822

Fig. 8. **Entity disambiguation progress**. Performance of the classic entity linking models (green) with the more recent neural models (gray) on the AIDA test set shows an improvement (around 10 points of accuracy).

and how the best disambiguation models based on classical machine learning methods (denoted as "non-neural") correspond to the recent state-of-the-art models based on deep neural networks (denoted as "neu-

Table 4

Entity disambiguation evaluation. Micro F1/Accuracy scores of neural entity disambiguation as compared to some classic models on common evaluation datasets.

| Model	AIDA-B	KBP'10	MSNBC	AQUAINT	ACE-2004	CWEB	WW	KBP'15 (es)	KBP'15 (zh)
	Accuracy	Accuracy	Micro F1	Micro F1	Micro F1	Micro F1	Micro F1	Accuracy	Accuracy
Non-Neural Baseline Models									
DBpedia Spotlight (2011) [108]	0.561	-	0.421	0.518	0.539	-	-	-	-
AIDA (2011) [67]	0.770	-	0.746	0.571	0.798	-	-	-	-
Ratinov et al. (2011) [148]	-	-	0.750	0.830	0.820	0.562	0.672	-	-
WAT (2014) [140]	0.805	-	0.788	0.754	0.796	-	-	-	-
Babelfy (2014) [115]	0.758	-	0.762	0.704	0.619	-	-	-	-
Lazic et al. (2015) [84]	0.864	-	-	-	-	-	-	-	-
Chisholm and Hachey (2015) [27]	0.887	-	-	-	-	-	-	-	-
PBOH (2016) [54]	0.804	-	0.861	0.841	0.832	-	-	-	-
Guo and Barbosa (2018) [61]	0.890	-	0.920	0.870	0.880	0.770	0.845	-	-
Neural Models									
Sun et al. (2015) [171]	-	0.839	-	-	-	-	-	-	-
Francis-Landau et al. (2016) [49]	0.855	-	-	-	-	-	-	-	-
Fang et al. (2016) [42]	-	0.889	0.755	0.852	0.808	-	-	-	-
Yamada et al. (2016) [194]	0.931	0.855	-	-	-	-	-	-	-
Zwicklbauer et al. (2016b) [211]	0.784	-	0.911	0.842	0.907	-	-	-	-
Tsai and Roth (2016) [176]	-	-	-	-	-	-	-	0.824	0.851
Nguyen et al. (2016b) [127]	0.872	-	-	-	-	-	-	-	-
Globerson et al. (2016) [56]	0.927	0.872	-	-	-	-	-	-	-
Cao et al. (2017) [19]	0.851	-	-	-	-	-	-	-	-
Eshel et al. (2017) [40]	0.873	-	-	-	-	-	-	-	-
Ganea and Hofmann (2017) [53]	0.922	-	0.937	0.885	0.885	0.779	0.775	-	-
Gupta et al. (2017) [62]	0.829	-	-	-	0.907	-	-	-	-
Nie et al. (2018) [129]	0.898	0.891	-	-	-	-	-	-	-
Shahbazi et al. (2018) [157]	0.944	0.879	-	-	-	-	-	-	-
Le and Titov (2018) [85]	0.931	-	0.939	0.884	0.900	0.775	0.780	-	-
Radhakrishnan et al. (2018) [144]	0.930	0.896	-	-	-	-	-	-	-
Kolitsas et al. (2018) [82]	0.831	-	0.864	0.832	0.855	-	-	-	-
Sil et al. (2018) [164]	0.940	0.874	-	-	-	-	-	0.823	0.844
Upadhyay et al. (2018a) [179]	-	-	-	-	-	-	-	**0.844**	**0.860**
Cao et al. (2018) [20]	0.800	0.910	-	0.870	0.880	-	0.860	-	-
Raiman and Raiman (2018) [146]	0.949	0.909	-	-	-	-	-	-	-
Shahbazi et al. (2019) [158]	0.962	0.883	0.923	0.901	0.887	0.784	0.798	-	-
Gillick et al. (2019) [55]	-	0.870	-	-	-	-	-	-	-
Le and Titov (2019b) [87]	0.815	-	-	-	-	-	-	-	-
Le and Titov (2019a) [86]	0.897	-	0.922	0.907	0.881	0.782	0.817	-	-
Fang et al. (2019) [43]	0.943	-	0.928	0.875	0.912	0.785	0.828	-	-
Yang et al. (2019) [200]	0.946	-	0.946	0.885	0.901	0.756	0.788	-	-
Xue et al. (2019) [192]	0.924		0.944	0.919	0.911	0.801	0.855	-	-
Zhou et al. (2019) [207]	-	-	-	-	-	-	-	0.829	0.855
Hou et al. (2020) [69]	0.926	-	0.943	0.912	0.907	0.785	0.819	-	-
Onoe and Durrett (2020) [131]	0.859	-	-	-	-	-	-	-	-
Wu et al. (2020b) [191]	-	**0.945**	-	-	-	-	-	-	-
Wu et al. (2020a) [190]	0.931	-	0.927	0.894	0.906	**0.814**	0.792	-	-
Fang et al. (2020) [44]	0.830	-	0.800	0.880	0.890	-	-	-	-
Chen et al. (2020) [25]	0.937	-	0.945	0.898	0.908	0.782	0.810	-	-
Mulang' et al. (2020) [117]	0.949	-	-	-	-	-	-	-	-
Yamada et al. (2021) [198]	**0.971**	-	**0.963**	**0.935**	**0.919**	0.789	**0.892**	-	-
De Cao et al. (2021) [33]	0.933	-	0.943	0.909	0.911	0.773	0.879	-	-
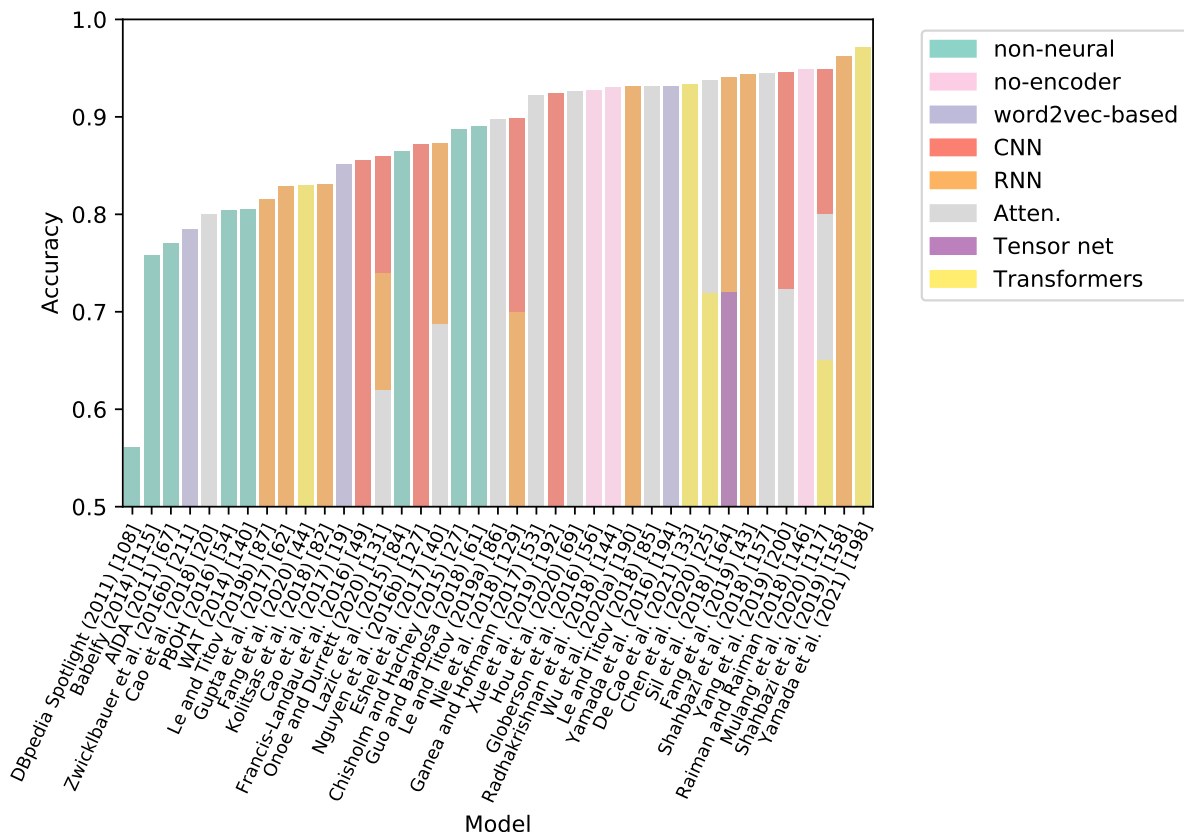
Fig. 9. **Mention/context encoder type for entity disambiguation**. Performance of the entity disambiguation models on the AIDA test set with mention/context encoder displayed with different colors as defined in Table 2. The bars with multiple colors refer to the models that use different types of encoder models; the bars do not reflect any meaning on the percentage. Note: we assigned the "RNN" label for the models LSTM, GRU, and ELMo; the "Transformers" label for BERT and BART models.

ral"). As one may observe, the models based on deep learning substantially improve the EL performance pushing the state of the art by around 10 percentage points in terms of accuracy.

Table 4 presents the comparison of the ED models in detail on several datasets presented above. The model of Yamada et al. [198] yields the best result on AIDA and appears to behave robustly across different datasets, getting top scores or near top scores for most of them. Here, we should also mention that none of the non-neural baselines reach the best results on any dataset.

Among local models for disambiguation, the best results are reported by Shahbazi et al. [158] and Wu et al. [191]. It is worth noting that the latter model can be used in the zero-shot setting. Shahbazi et al. [158] has the best score on AIDA among other local models outperforming them by a substantial margin. However, this is due to the use of the less-ambiguous resource

of Pershina et al. [137] for candidate generation, while many other works use the YAGO-based resource provided by Ganea and Hofmann [53], which typically yields lower results.

The common trend is that the global models (those trying to disambiguate several entity occurrences at once) outperform the local ones (relying on a single mention and its context). The best considered ED model of Yamada et al. [198] is global. Its performance improvements over competitors are attributed by the authors to the novel masked entity prediction objective that helps to fine-tune pre-trained BERT for producing contextualized entity embeddings and to the multi-step global disambiguation algorithm.

Finally, as one could see from Table 4, the least number of experiments is reported on the non-English datasets (TAC KBP datasets for Chinese and Spanish). Among the four reported results, the approach of Upadhyay et al. [180] provides the best scores, yet out-
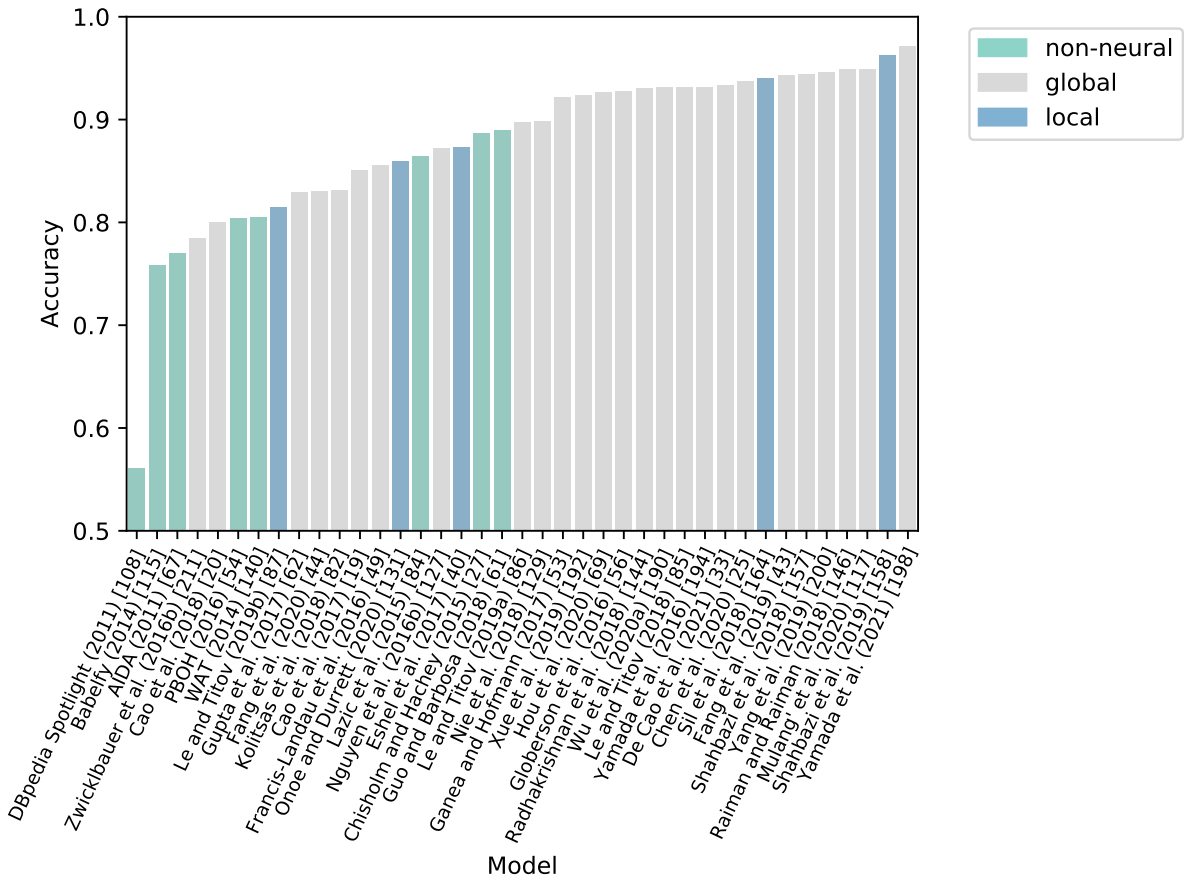
Fig. 10. **Local-Global entity disambiguation**. Performance of the entity disambiguation models on the AIDA test set with local/global models displayed with different colors as defined in Table 2. Note, some models, like Francis-Landau et al. [49], do not rely on global coherence, but they use document-wide context or multiple mentions at once, as explained in Table 2.

performing the other three approaches only by a small margin.

Mention/Context Encoder Type Figure 9 provides further analysis of the performance of entity disambiguation models presented above. The top performing model by Yamada et al. [198] is based on Transformers. It is followed by the model of Shahbazi et al. [158], which relies on RNNs: more specifically, it relies on the ELMo encoder that is based on pre-trained bidirectional LSTM cells. Overall, RNN is a popular choice for the mention-context encoder. However, recently, self-attention-based encoders, and especially the ones based on pre-trained Transformer networks, have gained popularity.

Several approaches, such as Yamada et al. [194], rely on simpler encoders based on the word2vec models, yet none of them manage to outperform more complex deep architectures.

Local-global models Figure 10 visualizes the usage of the local and global context in various models for entity disambiguation. As one can observe from the plot, the majority of models perform global entity disambiguation, including the top-performing model by Yamada et al. [198]. Although Shahbazi et al. [158] provide a local model, they also show a good performance.

Joint Entity Mention Detection and Disambiguation Table 5 presents results of the joint MD and ED models. Only a fraction of the models presented in Table 2 is capable of performing both entity mention detection and disambiguation; thus, the list of results is much shorter. Among the joint MD and ED solutions, the best results on the AIDA dataset are reported by Chen et al. [23]. However, Poerner et al. [142] note that these results might not be directly comparable with others due to a different evaluation protocol. The best com-

Table 5

Evaluation of joint MD-ED models. Micro F1 scores for joint entity mention detection and entity disambiguation evaluation on AIDA-B and MSNBC datasets.

Model	AIDA-B	MSNBC
Non-Neural Baseline Models		
DBpedia Spotlight (2011) [108]	0.578	0.406
AIDA (2011) [67]	0.728	0.651
WAT (2014) [140]	0.730	0.645
Babelfy (2014) [115]	0.485	0.397
Neural Models		
Kolitsas et al. (2018) [82]	0.824	0.724
Martins et al. (2019) [107]	0.819	-
Peters et al. (2019) [139]	0.744	-
Broscheit (2019) [17]	0.793	-
Chen et al. (2020) [23]	**0.877**	-
Poerner et al. (2020) [142]	0.850	-
De Cao et al. (2021) [33]	0.837	**0.737**

parable results on the AIDA dataset are shown by E-BERT [142]. On the MSNBC dataset, the top scores are achieved by De Cao et al. [33] with an autoregressive model. The scores of the systems that solve both tasks at once fall behind the disambiguation-only systems since they rely on noisy mention boundaries produced by themselves. In the joint MD and ED setting, the neural models also substantially (up to around 10 points) outperform the classic models.

On Effect of Hyperparameter Search As explained above, in Tables 4 and 5, we present the best scores reported by the authors of the models. In principle, each neural model can be further tuned as shown by Reimers and Gurevych [149], but also the variance of neural models is rather high in general. Therefore, it may be possible to further optimize meta-parameters of one (possibly simpler) neural model so that it outperforms a more complex (but tuned in a less optimal way) model. One common example of such a case is RoBERTa [98], which is basically the original BERT model, which was carefully and robustly optimized. This model outperformed many successors of the BERT model, showing the new state-of-the-art results on various tasks while keeping the original architecture.

4.2. Entity Relatedness

The quality of entity representations can be measured by how they capture semantic relatedness between entities [19, 53, 70, 162, 194]. Moreover, the semantic relatedness is an important feature in global EL [21, 38]. In this section, we present results of entity relatedness evaluation, which is different from evaluation of EL pipelines.

4.2.1. Experimental Setup

We summarize results from several works obtained on a benchmark of Ceccarelli et al. [21] for entity relatedness evaluation based on the dataset of Hoffart et al. [67]. Given a target entity and a list of candidate entities, the task is to rank candidates semantically related to the target higher than the others [53]. For the most of the considered works, the relatedness is measured by the cosine similarity of entity representations. For comparison, we also add results for two other approaches: a well-known Wikipedia hyperlink-based measure devised by Milne and Witten [112] known as WLM and a KG-based measure of El Vaigh et al. [38].

The evaluation metrics are normalized discounted cumulative gain (nDCG) [73] and a mean average precision (MAP) [105]. nDCG is a commonly used metric in information retrieval. It discounts the correct answers, depending on their rank in predictions Manning et al. [105]:

$$nDCG(Q,k) = \frac{1}{|Q|} \sum_{j=1}^{|Q|} Z_{kj} \sum_{m=1}^{k} \frac{2^{R(j,m)} - 1}{\log_2(1+m)},$$

(26)

where Q is the set of target entities (queries); Z_{kj} is a normalization factor, which corresponds to ideal ranking; k is a number of candidates for each query; $R(j,m) \in \{0,1\}$ is the gold-standard annotation of relatedness between the target entity j and a candidate m.

MAP is another common metric in information retrieval [105]:

$$MAP(Q) = \frac{1}{|Q|} \sum_{j=1}^{|Q|} \frac{1}{m_j} \sum_{k=1}^{m_j} Precision@r_{jk}, \quad (27)$$

where Q is a set of target entities (queries); m_j is the number of related candidate entities for the target j, and $Precision@r_{jk}$ is a precision at rank r_{jk}, where

Table 6

Entity relatedness evaluation. Reported results for entity relatedness evaluation on the test set of Ceccarelli et al. [21] .

Model	nDCG@1	nDCG@5	nDCG@10	MAP
Milne and Witten (2008) [112]	*0.540*	*0.520*	*0.550*	*0.480*
Huang et al. (2015) [70]	**0.810**	0.730	0.740	**0.680**
Yamada et al. (2016) [194]	0.590	0.560	0.590	0.520
Ganea and Hofmann (2017) [53]	0.632	0.609	0.641	0.578
Cao et al. (2017) [19]	0.613	0.613	0.654	0.582
El Vaigh et al. (2019) [38]	*0.690*	*0.640*	*0.580*	-
Shi et al. (2020) [162]	0.680	**0.814**	**0.820**	-

r_{jk} is a rank of each related candidate in the prediction $k = 1..m_j$ [105].

4.2.2. Discussion of Results

Table 6 summarizes the evaluation results in the entity relatedness task reported by the authors of the models. The scores of Milne and Witten [112] are taken from Huang et al. [70].

The highest scores of nDCG@1 and MAP are reported by Huang et al. [70], and the best scores of nDCG@5 and nDCG@10 are reported by Shi et al. [162]. The high scores of Huang et al. [70] can be attributed to the usage of different information sources for constructing entity representations, including entity types and entity relations [53]. Shi et al. [162] also use various types of data sources for constructing entity representations, including textual and knowledge graph information, like the types provided by a category hierarchy of a knowledge graph.

Note that cosine similarity based measures perform better in terms of nDCG@10 than the methods based on relations in KG (shown as italic in Table 6).

5. Applications of Entity Linking

In this section, we first give a brief overview of established applications of the entity linking technology and then discuss recently emerged use-cases specific to neural entity linking based on injection of these models as a part of a larger neural network, e.g. in a neural language model.

5.1. Established Applications

Text Mining An EL tool is a typical building block for text mining systems. Extracting and resolving the ambiguity of entity mentions is one of the first steps in a common information extraction pipeline. The ambiguity problem is especially crucial for such domains

as biomedical and clinical text processing due to variability of medical terms, the complexity of medical ontologies such as UMLS [12], and scarcity of annotated resources. There is a long history of development of EL tools for biomedical literature and electronic health record mining applications [6, 24, 83, 101, 109, 155, 167, 178, 209]. These tools have been successfully applied for summarization of clinical reports [104], extraction of drug-disease treatment relationships [81], mining chemical-induced disease relations [10], differential diagnosis [5], patient screening [41], and many other tasks. Besides medical text processing, EL is widely used for mining social networks and news [2, 113]. For example, Twitcident [1] uses the DBpedia Spotlight [108] EL system for mining Twitter messages for small scale incidents. Provatorova et al. [143] leverage a recently proposed EL toolkit REL [181] for mining historical newspapers for people, places, and other entities in the CLEF HIPE 2020 evaluation campaign [37]. Luo et al. [103] automatically construct a large-scale dataset of images and text captions that describe real and out-of-context news. They leverage REL for linking entities in image captions, which helps to automatically measure inconsistency between images and their text captions.

Knowledge graph population EL is one of the necessary steps of knowledge graph population algorithms. Before populating a KG with new facts extracted from raw texts, we have to determine mentioned concepts in these texts and link them to the corresponding graph nodes. A series of evaluation workshops TAC[14] provides a forum for KG population tools (TAC KBP), as well as benchmarks for various subsystems including EL. For example, Ji and Grishman [74] and Ellis et al. [39] overview various successful systems for knowledge graph population participated in the TAC

---

[14]https://tac.nist.gov/2019/index.html

KBP 2010 and 2015 tasks. Shen et al. [161] propose a knowledge graph population algorithm that not only uses the results of EL, but also helps to improve EL itself. It iteratively populates a KG, while the EL model benefits from added knowledge and continuously learns to disambiguate better.

Information retrieval and question-answering EL is also widely used in information retrieval and question-answering systems. EL helps to complement search results with additional semantic information, to resolve query ambiguity, and to restrict the search space. For example, Lee et al. [91] use EL to complement the results of a biomedical literature search engine with found entities: genes, diseases, drugs, etc. COVIDASK [90], a real-time question answering system that helps researchers to retrieve information related to coronavirus, uses the BioSyn model [172] for processing COVID-19 articles and linking mentions of drugs, symptoms, diseases to concepts in biomedical ontologies. Links to entity descriptions help users to navigate the search results, which enhances the usability of the system. Yih et al. [202] apply EL for pruning the search space of a question answering system. For the query: "Who first voiced Meg on Family Guy?", after linking "Meg" and "Family Guy" to entities in a KG, the task becomes to resolve the predicates to the "Family Guy (the TV show)" entry rather than all entries in the KG. Shnayderman et al. [163] develop a fast EL algorithm for pre-processing large corpora for their autonomous debating system [166] with the goal to conduct an argumentative dialog with an opponent on some topic and to prove a predefined point of view. The system uses the results of entity linking for corpus-based argument retrieval.

5.2. *Novel Applications: Neural Entity Linking for Training Better Neural Language Models*

Neural EL models have unlocked the new category of applications that have not been available for classical machine learning methods. Namely, neural models allow the integration of an entire entity linking system inside a larger neural network such as BERT. As they are both neural networks, such kind of integration becomes possible. After integrating an entity linker into another model's architecture, we can also expand the training objective with an additional EL-related task and train parameters of all neural components jointly:

$$\mathcal{L}_{\text{JOINT}} = \mathcal{L}_{\text{BERT}} + \mathcal{L}_{\text{EL-related}} . \qquad (28)$$

Neural entity linkers can be integrated in any other networks. The main novel trend is the use of EL information for representation learning. Several studies have shown that contextual word representations could benefit from information stored in KGs by incorporating EL into deep language models (LMs) for transfer learning.

KnowBERT [139] injects one or several entity linkers between top layers of the BERT architecture and optimizes the whole network for multiple tasks: the masked language model (MLM) task and next sentence prediction (NSP) from the original BERT model, as well as EL:

$$\mathcal{L}_{\text{BERT}} = \mathcal{L}_{\text{NSP}} + \mathcal{L}_{\text{MLM}} . \qquad (29)$$

$$\mathcal{L}_{\text{KnowBert}} = \mathcal{L}_{\text{NSP}} + \mathcal{L}_{\text{MLM}} + \mathcal{L}_{\text{EL}} . \qquad (30)$$

The authors adopt the general end-to-end EL architecture of [82] but use only the local context for disambiguation and an encoder based on self-attention over the representations generated by underlying BERT layers. If the EL subsystem detects an entity mention in a given sentence, corresponding pre-built entity representations of candidates are utilized for calculating the updated contextual word representations generated on the current BERT layer. These representations are used as input in a subsequent layer and can also be modified by a subsequent EL subsystem. Experiments with two EL subsystems based on Wikidata and WordNet show that presented modifications in KnowBERT help it to slightly surpass other deep pre-trained language models in tasks of relationship extraction, WSD, and entity typing.

ERNIE [206] expands the BERT [36] architecture with a knowledgeable encoder (K-Encoder), which fuses contextualized word representations obtained from the underlying self-attention network with entity representations from a pre-trained TransE model [15]. EL in this study is performed by an external tool TAGME [47]. For model pre-training, in addition to the MLM task, the authors introduce the task of restoring randomly masked entities in a given sequence keeping the rest of the entities and tokens. They refer to this procedure as a denoising entity auto-encoder (dEA):

$$\mathcal{L}_{\text{ERNIE}} = \mathcal{L}_{\text{NSP}} + \mathcal{L}_{\text{MLM}} + \mathcal{L}_{\text{dEA}} . \qquad (31)$$

Using English Wikipedia and Wikidata as training data, the authors show that introduced modifications

provide performance gains in entity typing, relation classification, and several GLUE tasks [185].

Wang et al. [188] train a disambiguation network named KEPLER using the composition of two losses: regular MLM and a Knowledge Embedding (KE) loss based on the TransE [15] objective for encoding graph structures:

$$\mathcal{L}_{\text{KEPLER}} = \mathcal{L}_{\text{MLM}} + \mathcal{L}_{\text{KE}}. \tag{32}$$

In the KE loss, representations of entities are obtained from their textual descriptions encoded with a self-attention network [98], and representations of relations are trainable vectors. The network is trained on a dataset of entity-relation-entity triplets with descriptions gathered from Wikipedia and Wikidata. Although the system exhibits a significant drop in performance on general NLP benchmarks such as GLUE [185], it shows increased performance on a wide range of KB-related tasks such as TACRED [205], FewRel [63], and OpenEntity [28].

Yamada et al. [196] propose a deep pre-trained model called "Language Understanding with Knowledge-based Embeddings" (LUKE). They modify RoBERTa [98] by introducing an additional pre-training objective and an entity-aware self-attention mechanism. The objective is a simple adoption of the MLM task to entities $\mathcal{L}_{MLMe}$, instead of tokens, the authors suggest restoring randomly masked entities in an entity-annotated corpus.

$$\mathcal{L}_{\text{LUKE}} = \mathcal{L}_{\text{MLM}} + \mathcal{L}_{\text{MLMe}}. \tag{33}$$

Although the corpus used in this work is constructed from Wikipedia by considering hyperlinks to other Wikipedia pages as mentions of entities in a KG, alternatively, it can be generated using an external entity linker.

The entity-aware attention mechanism helps LUKE differentiate between words and entities via introducing four different query matrices for matching words and entities: one for each pair of input types (entity-entity, entity-word, word-entity, and the standard word-word). The proposed modifications give LUKE exceptional performance improvements over previous models in five tasks: Open Entity (entity typing) [28], TACRED (relation classification) [205], CoNLL-2003 (named entity recognition) [174], ReCoRD (cloze-style question answering) [204], and SQuAD 1.1 (reading comprehension) [147].

Févry et al. [48] propose a method for training a language model and entity representations jointly, which they call Entities as Experts (EaE). The model is based on the Transformer architecture and is similar to KnowBERT [139]. However, in addition to the trainable word embedding matrix, EaE features a separate trainable matrix for entity embeddings referred to as "memory". The standard Transformer is also extended with an "entity memory" layer, which takes the output from the preceding Transformer layer and populates it with entity embeddings of mentions in the text. The retrieved entity embeddings are integrated into token representations by summation before layer normalization. To avoid dependence at inference on an external mention detector, the model applies a classifier to the output of Transformer blocks as in a sequence labeling model.

Analogously to [196], the EaE is trained on a corpus annotated with mentions and entity links. The final loss function sums up of three components: the standard MLM objective, mention boundary detection loss as in a sequence labeling model $\mathcal{L}_{NER}$, and an entity linking objective that facilitates entity representations generated in the model to be close to entity embedding of an annotated entity.

$$\mathcal{L}_{\text{EaE}} = \mathcal{L}_{\text{MLM}} + \mathcal{L}_{\text{NER}} + \mathcal{L}_{\text{EL}}. \tag{34}$$

This approach to integrating knowledge about entities into LMs provides a significant performance boost in open domain question answering. EaE, having only 367 million of parameters, outperforms the 11 billion parameter version of T5 [145] on the TriviaQA task [79]. The authors also show that EAE contains more factual knowledge than a comparably-sized BERT model.

Poerner et al. [142] present an E-BERT language model that also takes advantage of entity representations. This model is close to [206] as it also injects entities directly into the text and mixes entity representations with word embeddings in a similar way. However, instead of updating the weights of the whole pre-trained language model, they train only a linear transformation for aligning pre-trained entity representations with representations of word piece tokens of BERT. Such a small modification helps this model to outperform baselines on unsupervised question answering, supervised relation classification, and end-to-end entity linking.

The considered works demonstrate that the integration of structured KGs and LMs usually helps to solve

knowledge-oriented tasks: question answering (including open-domain QA), entity typing, relation extraction, and others. A high-precision supervision signal from KGs either leads to notable performance improvements or allows to reduce the number of trainable parameters of an LM while keeping a similar performance. Entity linking acts as a bridge between highly structured knowledge graphs and more flexible language models. We expect this approach to be crucial for the construction of future foundation models.

6. Conclusion

In this survey, we have analyzed recently proposed neural entity linking models, which generally solve the task with higher accuracy than classical methods. We provide a generic neural entity linking architecture, which is applicable for most of the neural EL systems, including the description of its components, e.g. candidate generation, entity ranking, mention and entity encoding. Various modifications of the general architecture are grouped into four common directions: (1) joint entity mention detection and linking models, (2) global entity linking models, (3) domain-independent approaches, including zero-shot and distant supervision methods, and (4) cross-lingual techniques. Taxonomy figures and feature tables are provided to explain the categorization and to show which prominent features are used in each method.

The majority of studies still rely on external knowledge for the candidate generation step. The mention encoders have made a shift from convolutional and recurrent models to self-attention architectures and start using pre-trained contextual language models like BERT. There is a current surge of methods that tackle the problem of adapting a model trained on one domain to another domain in a zero-shot fashion. These approaches do not need any annotated data in the target domain, but only descriptions of entities from this domain to perform such adaptation. It is shown in several works that the cross-encoder architecture is superior as compared to models with separate mention and entity encoders. The global context is widely used, but there are few recent studies that focus only on local EL.

Among the solutions that perform mention detection and entity disambiguation jointly, the leadership is owned by the entity-enhanced BERT model (E-BERT) of Poerner et al. [142] and the autoregressive model of De Cao et al. [33] based on BART. Among published local models for disambiguation, the best results are

reported by Shahbazi et al. [158] and Wu et al. [191]. The former solution leverages entity-aware ELMo (E-ELMo) trained to additionally predict entities along with words as in language-modelling task. The latter solution is based on a BERT bi-/cross-encoder and can be used in the zero-shot setting. Yamada et al. [198] report results that are consistently better in comparison to all other solutions. Their high scores are attributed to the masked entity prediction mechanism for entity embedding and the usage of the pre-trained model based on BERT with a multi-step global scoring function.

7. Future Directions

We identify five promising directions of future work in entity linking listed below:

1. **More end-to-end models without an explicit candidate generation step**: The candidate generation step relies on pre-constructed external resources or heuristics, as discussed in Section 3.1.1. Both the recall and precision of EL systems depend on their completeness and ambiguity. The necessity of building such resources is also an obvious obstacle for applying models in zero-shot / cross-lingual settings. Several recent works demonstrate that it is possible to achieve high EL performance without external pre-built resources [55, 191] or eliminate the candidate generation step [16, 17]. There is also a line of works devoted to methods that perform mention detection and entity disambiguation jointly [33, 82], which helps to avoid error propagation through multiple independent processing steps in an EL pipeline. We believe that a possible further research direction would be the development of entirely end-to-end trainable EL pipelines similar in spirit to the system of Broscheit [17].

2. **Further development of zero-shot approaches to address emerging entities**: We also expect that zero-shot EL will rapidly evolve, engaging other features like global coherence across all entities in a document, NIL prediction, joining MD and ED steps together, or providing completely end-to-end solutions. The latter would be an especially challenging task but also a fascinating research direction. To allow for a proper comparison, more standardized benchmarks and evaluation processes for zero-shot methods are dearly needed.

3. **More use-cases of EL-enriched language models**: Some studies [139, 142, 188, 206] have shown improvements over contextual language models by including knowledge stored in KGs. They incorporate entity linking into these deep models to use information in KGs. In future work, more use-cases are expected to enhance language models by using entity linking. The enriched representations would be used in downstream tasks, enabling improvements there.

4. **Integration of EL loss in more neural models**: It may be interesting to integrate EL loss in other neural models distinct from the language models, but in a similar fashion as the models described in Section 5.2. Due to the fact that an end-to-end EL model is also just a neural network, such integration with other networks is technically straightforward. Some multi task learning methods have been already proposed, e.g. joint relation extraction and entity linking [10]. Since entity linking is a key step in information extraction, injecting information about entities contained in an EL model and multitask learning are expected to be useful for solving other related tasks.

5. **Multimodal EL**: We witness the rise of a fascinating information extraction research direction that aims to build models capable of processing not only text, but also data from other modalities like images. For example, Moon et al. [113] and Adjali et al. [2] leverage both text and images in social media posts for entity linking. Without taking into account an additional modality it would be impossible to correctly disambiguate entities in a very noisy and limited textual context. Entity linking methods in the near future potentially could take advantage of multimodal cross-attention and a surge of other techniques recently developed to improve processing multiple types of data in a single architecture [72, 120]. We consider that vice-versa is also possible: EL could be seamlessly integrated into models for processing data with multiple modalities. EL not only provides disambiguation of mentions in the text but also connects a data instance to a knowledge graph, which opens the possibility of using reasoning elements during the solution of the final task.

Appendix A. Public Implementations of Neural Entity Linking Models

Table 7

Publicly available implementations (either provided in the paper or available at PapersWithCode.com) of the neural models presented in Table 2.

Model	Link for Source Code
Sun et al. (2015) [171]	-
Francis-Landau et al. (2016) [49]	https://github.com/matthewfl/nlp-entity-convnet
Fang et al. (2016) [42]	-
Yamada et al. (2016) [194]	https://github.com/wikipedia2vec/wikipedia2vec
Zwicklbauer et al. (2016b) [211]	https://github.com/quhfus/DoSeR
Tsai and Roth (2016) [176]	-
Nguyen et al. (2016b) [127]	-
Globerson et al. (2016) [56]	-
Cao et al. (2017) [19]	https://github.com/TaoMiner/bridgeGap
Eshel et al. (2017) [40]	https://github.com/yotam-happy/NEDforNoisyText
Ganea and Hofmann (2017) [53]	https://github.com/dalab/deep-ed
Moreno et al. (2017) [114]	-
Gupta et al. (2017) [62]	https://github.com/nitishgupta/neural-el
Nie et al. (2018) [129]	-
Sorokin and Gurevych (2018) [168]	https://github.com/UKPLab/starsem2018-entity-linking
Shahbazi et al. (2018) [157]	-
Le and Titov (2018) [85]	https://github.com/lephong/mulrel-nel
Newman-Griffis et al. (2018) [125]	https://github.com/OSU-slatelab/JET
Radhakrishnan et al. (2018) [144]	https://github.com/priyaradhakrishnan0/ELDEN
Kolitsas et al. (2018) [82]	https://github.com/dalab/end2end_neural_el
Sil et al. (2018) [164]	-
Upadhyay et al. (2018a) [179]	https://github.com/shyamupa/xelms
Cao et al. (2018) [20]	https://github.com/TaoMiner/NCEL
Raiman and Raiman (2018) [146]	https://github.com/openai/deeptype
Mueller and Durrett (2018) [116]	https://github.com/davidandym/wikilinks-ned
Shahbazi et al. (2019) [158]	-
Logeswaran et al. (2019) [100]	https://github.com/lajanugen/zeshel
Gillick et al. (2019) [55]	https://github.com/google-research/google-research/tree/master/dense_representations_for_entity_retrieval
Peters et al. (2019) [139]	https://github.com/allenai/kb
Le and Titov (2019b) [87]	https://github.com/lephong/dl4el
Le and Titov (2019a) [86]	https://github.com/lephong/wnel
Fang et al. (2019) [43]	-
Martins et al. (2019) [107]	-
Yang et al. (2019) [200]	https://github.com/YoungXiyuan/DCA
Xue et al. (2019) [192]	https://github.com/DeepLearnXMU/RRWEL
Zhou et al. (2019) [207]	https://github.com/shuyanzhou/burn_xel
Broscheit (2019) [17]	https://github.com/samuelbroscheit/entity_knowledge_in_bert
Hou et al. (2020) [69]	https://github.com/fhou80/EntEmb
Onoe and Durrett (2020) [131]	https://github.com/yasumasaonoe/ET4EL
Chen et al. (2020) [23]	-
Wu et al. (2020b) [191]	https://github.com/facebookresearch/BLINK
Banerjee et al. (2020) [9]	https://github.com/debayan/pnel
Wu et al. (2020a) [190]	https://github.com/wujsAct/DGCN_EL
Fang et al. (2020) [44]	https://github.com/fangzheng123/SGEL
Chen et al. (2020) [25]	-
Botha et al. (2020) [16]	http://goo.gle/mewsli-dataset
Yao et al. (2020) [201]	https://github.com/seasonyao/Zero-Shot-Entity-Linking
Li et al. (2020) [94]	https://github.com/facebookresearch/BLINK/tree/master/elq
Poerner et al. (2020) [142]	https://github.com/npoe/ebert
Fu et al. (2020) [50]	http://cogcomp.org/page/publication_view/911
Mulang' et al. (2020) [117]	https://github.com/mulangonando/Impact-of-KG-Context-on-ED
Yamada et al. (2021) [198]	https://github.com/studio-ousia/luke
Gu et al. (2021) [60]	-
Tang et al. (2021) [173]	-
De Cao et al. (2021) [33]	https://github.com/facebookresearch/GENRE

References

[1] F. Abel, C. Hauff, G.-J. Houben, R. Stronkman and K. Tao, Twitcident: Fighting Fire with Information from Social Web Streams, in: *Proceedings of the 21st International Conference on World Wide Web*, WWW '12 Companion, Association for Computing Machinery, New York, NY, USA, 2012, pp. 305–308–. ISBN 9781450312301. doi:10.1145/2187980.2188035.

[2] O. Adjali, R. Besançon, O. Ferret, H. Le Borgne and B. Grau, Multimodal Entity Linking for Tweets, in: *Advances in Information Retrieval*, J.M. Jose, E. Yilmaz, J. Magalhães, P. Castells, N. Ferro, M.J. Silva and F. Martins, eds, Springer International Publishing, Cham, 2020, pp. 463–478. ISBN 978-3-030-45439-5.

[3] T. Al-Moslmi, M. Gallofré Ocaña, A.L. Opdahl and C. Veres, Named Entity Extraction for Knowledge Graphs: A Literature Overview, *IEEE Access* **8** (2020), 32862–32881. doi:10.1109/ACCESS.2020.2973928.

[4] R. Aly, A. Vlachos and R. McDonald, Leveraging Type Descriptions for Zero-shot Named Entity Recognition and Classification, in: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Association for Computational Linguistics, Online, 2021, pp. 1516–1528. doi:10.18653/v1/2021.acl-long.120. https://aclanthology.org/2021.acl-long.120.

[5] H. Amiri, M. Mohtarami and I. Kohane, Attentive Multiview Text Representation for Differential Diagnosis, in: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, Association for Computational Linguistics, Online, 2021, pp. 1012–1019. doi:10.18653/v1/2021.acl-short.128. https://aclanthology.org/2021.acl-short.128.

[6] A.R. Aronson and F.-M. Lang, An overview of MetaMap: historical perspective and recent advances, *Journal of the American Medical Informatics Association* **17**(3) (2010), 229–236. doi:10.1136/jamia.2009.002733.

[7] D. Bahdanau, K. Cho and Y. Bengio, Neural machine translation by jointly learning to align and translate, in: *3rd International Conference on Learning Representations, ICLR 2015*, San-Diego, California, USA, 2015. http://arxiv.org/abs/1409.0473.

[8] L. Banarescu, C. Bonial, S. Cai, M. Georgescu, K. Griffitt, U. Hermjakob, K. Knight, P. Koehn, M. Palmer and N. Schneider, Abstract Meaning Representation for Sembanking, in: *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, Association for Computational Linguistics, Sofia, Bulgaria, 2013, pp. 178–186. https://aclanthology.org/W13-2322.

[9] D. Banerjee, D. Chaudhuri, M. Dubey and J. Lehmann, PNEL: Pointer Network Based End-To-End Entity Linking over Knowledge Graphs, in: *The Semantic Web – ISWC 2020 - 19th International Semantic Web Conference, Athens, Greece, November 2-6, 2020, Proceedings, Part I*, Vol. 12506, J.Z. Pan, V. Tamma, C. d'Amato, K. Janowicz, B. Fu, A. Polleres, O. Seneviratne and L. Kagal, eds, Springer International Publishing, Cham, 2020, pp. 21–38. ISBN 978-3-030-62419-4. doi:10.1007/978-3-030-62419-4_2.

[10] T. Bansal, P. Verga, N. Choudhary and A. McCallum, Simultaneously Linking Entities and Extracting Relations from Biomedical Text without Mention-Level Supervision, *Proceedings of the AAAI Conference on Artificial Intelligence* **34**(05) (2020), 7407–7414. doi:10.1609/aaai.v34i05.6236. https://ojs.aaai.org/index.php/AAAI/article/view/6236.

[11] Y. Bengio, R. Ducharme, P. Vincent and C. Janvin, A Neural Probabilistic Language Model, *J. Mach. Learn. Res. – Journal of Machine Learning Research* **3**(null) (2003), 1137–1155–.

[12] O. Bodenreider, The Unified Medical Language System (UMLS): integrating biomedical terminology, *Nucleic Acids Research* **32**(suppl_1) (2004), D267–D270. doi:10.1093/nar/gkh061.

[13] P. Bojanowski, E. Grave, A. Joulin and T. Mikolov, Enriching Word Vectors with Subword Information, *Transactions of the Association for Computational Linguistics* **5** (2017), 135–146. doi:10.1162/tacl_a_00051. https://aclanthology.org/Q17-1010.

[14] K. Bollacker, C. Evans, P. Paritosh, T. Sturge and J. Taylor, Freebase: A Collaboratively Created Graph Database for Structuring Human Knowledge, in: *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, SIGMOD '08, Association for Computing Machinery, New York, NY, USA, 2008, pp. 1247–1250–. ISBN 9781605581026. doi:10.1145/1376616.1376746.

[15] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston and O. Yakhnenko, Translating Embeddings for Modeling Multirelational Data, in: *Advances in neural information processing systems*, Vol. 26, C.J.C. Burges, L. Bottou, M. Welling, Z. Ghahramani and K.Q. Weinberger, eds, Stateline, Nevada, USA, 2013, pp. 2787–2795. https://papers.nips.cc/paper/2013/file/1cecc7a77928ca8133fa24680a88d2f9-Paper.pdf.

[16] J.A. Botha, Z. Shan and D. Gillick, Entity Linking in 100 Languages, in: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Association for Computational Linguistics, Online, 2020, pp. 7833–7845. doi:10.18653/v1/2020.emnlp-main.630. https://aclanthology.org/2020.emnlp-main.630.

[17] S. Broscheit, 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)*, Association for Computational Linguistics, Hong Kong, China, 2019, pp. 677–685. doi:10.18653/v1/K19-1063. https://aclanthology.org/K19-1063.

[18] H. Cai, V.W. Zheng and K. Chang, A Comprehensive Survey of Graph Embedding: Problems, Techniques, and Applications, *IEEE Transactions on Knowledge & Data Engineering* **30**(09) (2018), 1616–1637. doi:10.1109/TKDE.2018.2807452.

[19] Y. Cao, L. Huang, H. Ji, X. Chen and J. Li, Bridge Text and Knowledge by Learning Multi-Prototype Entity Mention Embedding, in: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, Vancouver, Canada, 2017, pp. 1623–1633. doi:10.18653/v1/P17-1149. https://aclanthology.org/P17-1149.

[20] Y. Cao, L. Hou, J. Li and Z. Liu, Neural Collective Entity Linking, in: *Proceedings of the 27th International Conference on Computational Linguistics*, Association for Computational

Linguistics, Santa Fe, New Mexico, USA, 2018, pp. 675–686. https://aclanthology.org/C18-1057.

[21] D. Ceccarelli, C. Lucchese, S. Orlando, R. Perego and S. Trani, Learning Relatedness Measures for Entity Linking, in: *Proceedings of the 22nd ACM International Conference on Information & Knowledge Management*, CIKM '13, Association for Computing Machinery, New York, NY, USA, 2013, pp. 139–148. ISBN 9781450322638. doi:10.1145/2505515.2505711.

[22] A. Chang, V.I. Spitkovsky, C.D. Manning and E. Agirre, A comparison of Named-Entity Disambiguation and Word Sense Disambiguation, in: *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, European Language Resources Association (ELRA), Portorož, Slovenia, 2016, pp. 860–867. https://aclanthology.org/L16-1139.

[23] H. Chen, X. Li, A. Zukov Gregoric and S. Wadhwa, Contextualized End-to-End Neural Entity Linking, in: *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, Association for Computational Linguistics, Suzhou, China, 2020, pp. 637–642. https://aclanthology.org/2020.aacl-main.64.

[24] L. Chen, G. Varoquaux and F.M. Suchanek, A Lightweight Neural Model for Biomedical Entity Linking, 2021, pp. 12657–12665. https://ojs.aaai.org/index.php/AAAI/article/view/17499.

[25] S. Chen, J. Wang, F. Jiang and C.-Y. Lin, Improving Entity Linking by Modeling Latent Entity Type Information, *Proceedings of the AAAI Conference on Artificial Intelligence* **34**(05) (2020), 7529–7537. doi:10.1609/aaai.v34i05.6251. https://ojs.aaai.org/index.php/AAAI/article/view/6251.

[26] X. Cheng and D. Roth, Relational Inference for Wikification, in: *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Seattle, Washington, USA, 2013, pp. 1787–1796. https://aclanthology.org/D13-1184.

[27] A. Chisholm and B. Hachey, Entity Disambiguation with Web Links, *Transactions of the Association for Computational Linguistics* **3** (2015), 145–156. doi:10.1162/tacl_a_00129. https://aclanthology.org/Q15-1011.

[28] E. Choi, O. Levy, Y. Choi and L. Zettlemoyer, Ultra-Fine Entity Typing, in: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, Melbourne, Australia, 2018, pp. 87–96. doi:10.18653/v1/P18-1009. https://aclanthology.org/P18-1009.

[29] J. Chung, C. Gulcehre, K. Cho and Y. Bengio, Empirical evaluation of gated recurrent neural networks on sequence modeling, in: *NIPS 2014 Workshop on Deep Learning*, Montréal, Canada, 2014. https://arxiv.org/abs/1412.3555.

[30] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu and P. Kuksa, Natural Language Processing (Almost) from Scratch, *J. Mach. Learn. Res. – Journal of Machine Learning Research* **12**(null) (2011), 2493–2537–.

[31] R. Cotterell and K. Duh, Low-Resource Named Entity Recognition with Cross-lingual, Character-Level Neural Conditional Random Fields, in: *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, Asian Federation of Natural Lan-

guage Processing, Taipei, Taiwan, 2017, pp. 91–96. https://aclanthology.org/I17-2016.

[32] S. Cucerzan, 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)*, Association for Computational Linguistics, Prague, Czech Republic, 2007, pp. 708–716. https://aclanthology.org/D07-1074.

[33] N. De Cao, G. Izacard, S. Riedel and F. Petroni, Autoregressive Entity Retrieval, in: *International Conference on Learning Representations*, 2021. https://openreview.net/forum?id=5k8F6UU39V.

[34] D. Dessì, F. Osborne, D. Reforgiato Recupero, D. Buscaldi and E. Motta, Generating knowledge graphs by employing Natural Language Processing and Machine Learning techniques within the scholarly domain, *Future Generation Computer Systems* **116** (2021), 253–264. doi:10.1016/j.future.2020.10.026. https://www.sciencedirect.com/science/article/pii/S0167739X2033003X.

[35] T. Dettmers, P. Minervini, P. Stenetorp and S. Riedel, Convolutional 2D Knowledge Graph Embeddings, *Proceedings of the AAAI Conference on Artificial Intelligence* **32**(1) (2018). https://ojs.aaai.org/index.php/AAAI/article/view/11573.

[36] J. Devlin, M.-W. Chang, K. Lee and K. Toutanova, BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, in: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. doi:10.18653/v1/N19-1423. https://aclanthology.org/N19-1423.

[37] M. Ehrmann, M. Romanello, A. Flückiger and S. Clematide, Overview of CLEF HIPE 2020: Named Entity Recognition and Linking on Historical Newspapers, in: *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, A. Arampatzis, E. Kanoulas, T. Tsikrika, S. Vrochidis, H. Joho, C. Lioma, C. Eickhoff, A. Névéol, L. Cappellato and N. Ferro, eds, Springer International Publishing, Cham, 2020, pp. 288–310. ISBN 978-3-030-58219-7. doi:10.1007/978-3-030-58219-7_21.

[38] C.B. El Vaigh, F. Goasdoué, G. Gravier and P. Sébillot, Using Knowledge Base Semantics in Context-Aware Entity Linking, in: *Proceedings of the ACM Symposium on Document Engineering 2019*, DocEng '19, Association for Computing Machinery, New York, NY, USA, 2019. ISBN 9781450368872. doi:10.1145/3342558.3345393.

[39] J. Ellis, J. Getman, D. Fore, N. Kuster, Z. Song, A. Bies and S.M. Strassel, Overview of Linguistic Resources for the TAC KBP 2015 Evaluations: Methodologies and Results, in: *Proceedings of the 2015 Text Analysis Conference, TAC 2015*, NIST, Gaithersburg, Maryland, USA, 2015. https://tac.nist.gov/publications/2015/additional.papers/TAC2015.KBP_resources_overview.proceedings.pdf.

[40] Y. Eshel, N. Cohen, K. Radinsky, S. Markovitch, I. Yamada and O. Levy, Named Entity Disambiguation for Noisy Text, in: *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, Association for Computational Linguistics, Vancouver, Canada, 2017,

pp. 58–68. doi:10.18653/v1/K17-1008. https://aclanthology. org/K17-1008.

[41] H. Eyre, A.B. Chapman, K.S. Peterson, J. Shi, P.R. Alba, M.M. Jones, T.L. Box, S.L. DuVall and O.V. Patterson, Launching into clinical space with medspaCy: a new clinical text processing toolkit in Python, *arXiv preprint arXiv:2106.07799* ((in press, n.d.)).

[42] W. Fang, J. Zhang, D. Wang, Z. Chen and M. Li, Entity Disambiguation by Knowledge and Text Jointly Embedding, in: *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, Association for Computational Linguistics, Berlin, Germany, 2016, pp. 260–269. doi:10.18653/v1/K16-1026. https:// aclanthology.org/K16-1026.

[43] Z. Fang, Y. Cao, Q. Li, D. Zhang, Z. Zhang and Y. Liu, Joint Entity Linking with Deep Reinforcement Learning, in: *The World Wide Web Conference*, WWW '19, Association for Computing Machinery, New York, NY, USA, 2019, pp. 438–447–. ISBN 9781450366748. doi:10.1145/3308558.3313517.

[44] Z. Fang, Y. Cao, R. Li, Z. Zhang, Y. Liu and S. Wang, High Quality Candidate Generation and Sequential Graph Attention Network for Entity Linking, in: *Proceedings of The Web Conference 2020*, WWW '20, Association for Computing Machinery, New York, NY, USA, 2020, pp. 640–650–. ISBN 9781450370233. doi:10.1145/3366423.3380146.

[45] M. Färber, F. Bartscherer, C. Menne and A. Rettinger, Linked Data Quality of DBpedia, Freebase, OpenCyc, Wikidata, and YAGO, *Semantic Web* **9**(1) (2018), 77–129. doi:10.3233/SW-170275. https://content.iospress.com/articles/semantic-web/ sw275.

[46] C. Fellbaum (ed.), *WordNet: An Electronic Lexical Database*, MIT Press, Cambridge, MA, 1998. ISBN 978-0-262-06197-1.

[47] P. Ferragina and U. Scaiella, TAGME: On-the-Fly Annotation of Short Text Fragments (by Wikipedia Entities), in: *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, CIKM '10, Association for Computing Machinery, New York, NY, USA, 2010, pp. 1625–1628. ISBN 9781450300995. doi:10.1145/1871437.1871689.

[48] T. Févry, L. Baldini Soares, N. FitzGerald, E. Choi and T. Kwiatkowski, Entities as Experts: Sparse Memory Access with Entity Supervision, in: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Association for Computational Linguistics, Online, 2020, pp. 4937–4951. doi:10.18653/v1/2020.emnlp-main.400. https://aclanthology.org/2020.emnlp-main.400.

[49] M. Francis-Landau, G. Durrett and D. Klein, Capturing Semantic Similarity for Entity Linking with Convolutional Neural Networks, in: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Association for Computational Linguistics, San Diego, California, 2016, pp. 1256–1261. doi:10.18653/v1/N16-1150. https: //aclanthology.org/N16-1150.

[50] X. Fu, W. Shi, X. Yu, Z. Zhao and D. Roth, Design Challenges in Low-resource Cross-lingual Entity Linking, in: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Association for Computational Linguistics, Online, 2020, pp. 6418–6432.

doi:10.18653/v1/2020.emnlp-main.521. https://aclanthology. org/2020.emnlp-main.521.

[51] G. Fumera, F. Roli and G. Giacinto, Reject option with multiple thresholds, *Pattern recognition* **33**(12) (2000), 2099–2101.

[52] E. Gabrilovich, M. Ringgaard and A. Subramanya, FACC1: Freebase annotation of ClueWeb corpora, version 1 (release date 2013-06-26, format version 1, correction level 0), 2013, Note: http://lemurproject.org/clueweb09/.

[53] O.-E. Ganea and T. Hofmann, Deep Joint Entity Disambiguation with Local Neural Attention, in: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Copenhagen, Denmark, 2017, pp. 2619–2629. doi:10.18653/v1/D17-1277. https://aclanthology.org/ D17-1277.

[54] O.-E. Ganea, M. Ganea, A. Lucchi, C. Eickhoff and T. Hofmann, Probabilistic Bag-Of-Hyperlinks Model for Entity Linking, in: *Proceedings of the 25th International Conference on World Wide Web*, WWW '16, International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 2016, pp. 927–938. ISBN 9781450341431. doi:10.1145/2872427.2882988.

[55] D. Gillick, S. Kulkarni, L. Lansing, A. Presta, J. Baldridge, E. Ie and D. Garcia-Olano, Learning Dense Representations for Entity Retrieval, in: *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, Association for Computational Linguistics, Hong Kong, China, 2019, pp. 528–537. doi:10.18653/v1/K19-1049. https://aclanthology.org/K19-1049.

[56] A. Globerson, N. Lazic, S. Chakrabarti, A. Subramanya, M. Ringgaard and F. Pereira, Collective Entity Resolution with Multi-Focal Attention, in: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, Berlin, Germany, 2016, pp. 621–631. doi:10.18653/v1/P16-1059. https:// aclanthology.org/P16-1059.

[57] A. Goyal, V. Gupta and M. Kumar, Recent Named Entity Recognition and Classification techniques: A systematic review, *Computer Science Review* **29** (2018), 21–43. doi:10.1016/j.cosrev.2018.06.001. https://www.sciencedirect. com/science/article/pii/S1574013717302782.

[58] P. Goyal and E. Ferrara, Graph embedding techniques, applications, and performance: A survey, *Knowledge-Based Systems* **151** (2018), 78–94. doi:10.1016/j.knosys.2018.03.022. https://www.sciencedirect.com/science/article/pii/ S0950705118301540.

[59] A. Grover and J. Leskovec, Node2vec: Scalable Feature Learning for Networks, in: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, Association for Computing Machinery, New York, NY, USA, 2016, pp. 855–864. ISBN 9781450342322. doi:10.1145/2939672.2939754.

[60] Y. Gu, X. Qu, Z. Wang, B. Huai, N.J. Yuan and X. Gui, Read, Retrospect, Select: An MRC Framework to Short Text Entity Linking, *Proceedings of the AAAI Conference on Artificial Intelligence* **35**(14) (2021), 12920–12928. https://ojs.aaai.org/ index.php/AAAI/article/view/17528.

[61] Z. Guo and D. Barbosa, Robust named entity disambiguation with random walks, *Semantic Web* **9**(4) (2018), 459–479. doi:10.3233/SW-170273. https://content.iospress.com/articles/semantic-web/sw273.

[62] N. Gupta, S. Singh and D. Roth, Entity Linking via Joint Encoding of Types, Descriptions, and Context, in: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Copenhagen, Denmark, 2017, pp. 2681–2690. doi:10.18653/v1/D17-1284. https://www.aclweb.org/anthology/D17-1284.

[63] X. Han, H. Zhu, P. Yu, Z. Wang, Y. Yao, Z. Liu and M. Sun, FewRel: A Large-Scale Supervised Few-Shot Relation Classification Dataset with State-of-the-Art Evaluation, in: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Brussels, Belgium, 2018, pp. 4803–4809. doi:10.18653/v1/D18-1514. https://aclanthology.org/D18-1514.

[64] M.E. Hellman, The Nearest Neighbor Classification Rule with a Reject Option, *IEEE Transactions on Systems Science and Cybernetics* **6**(3) (1970), 179–185. doi:10.1109/TSSC.1970.300339.

[65] R. Herbei and M.H. Wegkamp, Classification with Reject Option, *The Canadian Journal of Statistics / La Revue Canadienne de Statistique* **34**(4) (2006), 709–721. http://www.jstor.org/stable/20445230.

[66] S. Hochreiter and J. Schmidhuber, Long Short-Term Memory, *Neural Computation* **9**(8) (1997), 1735–1780–. doi:10.1162/neco.1997.9.8.1735.

[67] J. Hoffart, M.A. Yosef, I. Bordino, H. Fürstenau, M. Pinkal, M. Spaniol, B. Taneva, S. Thater and G. Weikum, Robust Disambiguation of Named Entities in Text, in: *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Edinburgh, Scotland, UK., 2011, pp. 782–792. https://aclanthology.org/D11-1072.

[68] A. Hogan, E. Blomqvist, M. Cochez, C. D'amato, G.D. Melo, C. Gutierrez, S. Kirrane, J.E.L. Gayo, R. Navigli, S. Neumaier, A.-C.N. Ngomo, A. Polleres, S.M. Rashid, A. Rula, L. Schmelzeisen, J. Sequeda, S. Staab and A. Zimmermann, Knowledge Graphs, *ACM Computing Surveys* **54**(4) (2021). doi:10.1145/3447772.

[69] F. Hou, R. Wang, J. He and Y. Zhou, Improving Entity Linking through Semantic Reinforced Entity Embeddings, in: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, Online, 2020, pp. 6843–6848. doi:10.18653/v1/2020.acl-main.612. https://aclanthology.org/2020.acl-main.612.

[70] H. Huang, L. Heck and H. Ji, Leveraging deep neural networks and knowledge graphs for entity disambiguation, *arXiv preprint arXiv:1504.07678* (2015). https://arxiv.org/abs/1504.07678.

[71] S. Humeau, K. Shuster, M.-A. Lachaux and J. Weston, Poly-encoders: Architectures and Pre-training Strategies for Fast and Accurate Multi-sentence Scoring, in: *International Conference on Learning Representations*, 2020. https://openreview.net/forum?id=SkxgnnNFvH.

[72] A. Jaegle, F. Gimeno, A. Brock, O. Vinyals, A. Zisserman and J. Carreira, Perceiver: General Perception with Iterative Attention, in: *Proceedings of the 38th International Conference on Machine Learning*, M. Meila and T. Zhang, eds, Proceedings of Machine Learning Research, Vol. 139, PMLR, 2021, pp. 4651–4664. https://proceedings.mlr.press/v139/jaegle21a.html.

[73] K. Järvelin and J. Kekäläinen, Cumulated Gain-Based Evaluation of IR Techniques, *ACM Transactions on Information Systems* **20**(4) (2002), 422–446–. doi:10.1145/582415.582418.

[74] H. Ji and R. Grishman, Knowledge Base Population: Successful Approaches and Challenges, in: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, Association for Computational Linguistics, Portland, Oregon, USA, 2011, pp. 1148–1158. https://aclanthology.org/P11-1115.

[75] H. Ji, R. Grishman, H.T. Dang, K. Griffitt and J. Ellis, Overview of the TAC 2010 knowledge base population track, in: *Third Text Analysis Conference (TAC)*, Gaithersburg, Maryland, USA, 2010. https://blender.cs.illinois.edu/paper/kbp2010overview.pdf.

[76] H. Ji, J. Nothman, B. Hachey and R. Florian, Overview of TAC-KBP2015 Tri-lingual Entity Discovery and Linking, in: *Proceedings of the 2015 Text Analysis Conference, TAC 2015*, NIST, Gaithersburg, Maryland, USA, 2015, pp. 16–17. https://tac.nist.gov/publications/2015/additional.papers/TAC2015.KBP_Trilingual_Entity_Discovery_and_Linking_overview.proceedings.pdf.

[77] S. Ji, S. Pan, E. Cambria, P. Marttinen and P.S. Yu, A Survey on Knowledge Graphs: Representation, Acquisition, and Applications, *IEEE Transactions on Neural Networks and Learning Systems* **33**(2) (2022), 494–514. doi:10.1109/TNNLS.2021.3070843.

[78] K.S. Jones, S. Walker and S.E. Robertson, A Probabilistic Model of Information Retrieval: Development and Comparative Experiments Part 2, *Information Processing & Management* **36**(6) (2000), 809–840–. doi:10.1016/S0306-4573(00)00016-9.

[79] M. Joshi, E. Choi, D. Weld and L. Zettlemoyer, TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension, in: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, Vancouver, Canada, 2017, pp. 1601–1611. doi:10.18653/v1/P17-1147. https://aclanthology.org/P17-1147.

[80] R. Kar, S. Reddy, S. Bhattacharya, A. Dasgupta and S. Chakrabarti, Task-Specific Representation Learning for Web-Scale Entity Disambiguation, *Proceedings of the AAAI Conference on Artificial Intelligence* **32**(1) (2018). https://ojs.aaai.org/index.php/AAAI/article/view/12066.

[81] R. Khare, J. Li and Z. Lu, LabeledIn: Cataloging labeled indications for human drugs, *Journal of Biomedical Informatics* **52** (2014), 448–456. doi:10.1016/j.jbi.2014.08.004. https://www.sciencedirect.com/science/article/pii/S1532046414001853.

[82] N. Kolitsas, O.-E. Ganea and T. Hofmann, End-to-End Neural Entity Linking, in: *Proceedings of the 22nd Conference on Computational Natural Language Learning*, Association for Computational Linguistics, Brussels, Bel-

gium, 2018, pp. 519–529. doi:10.18653/v1/K18-1050. https://aclanthology.org/K18-1050.

[83] Z. Kraljevic, T. Searle, A. Shek, L. Roguski, K. Noor, D. Bean, A. Mascio, L. Zhu, A.A. Folarin, A. Roberts, R. Bendayan, M.P. Richardson, R. Stewart, A.D. Shah, W.K. Wong, Z. Ibrahim, J.T. Teo and R.J.B. Dobson, Multi-domain clinical natural language processing with MedCAT: The Medical Concept Annotation Toolkit, *Artificial Intelligence in Medicine* **117** (2021), 102083. doi:10.1016/j.artmed.2021.102083. https://www.sciencedirect.com/science/article/pii/S0933365721000762.

[84] N. Lazic, A. Subramanya, M. Ringgaard and F. Pereira, Plato: A Selective Context Model for Entity Resolution, *Transactions of the Association for Computational Linguistics* **3** (2015), 503–515. doi:10.1162/tacl_a_00154. https://aclanthology.org/Q15-1036.

[85] P. Le and I. Titov, Improving Entity Linking by Modeling Latent Relations between Mentions, in: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, Melbourne, Australia, 2018, pp. 1595–1604. doi:10.18653/v1/P18-1148. https://aclanthology.org/P18-1148.

[86] P. Le and I. Titov, Boosting Entity Linking Performance by Leveraging Unlabeled Documents, in: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, Florence, Italy, 2019a, pp. 1935–1945. doi:10.18653/v1/P19-1187. https://aclanthology.org/P19-1187.

[87] P. Le and I. Titov, Distant Learning for Entity Linking with Automatic Noise Detection, in: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, Florence, Italy, 2019b, pp. 4081–4090. doi:10.18653/v1/P19-1400. https://aclanthology.org/P19-1400.

[88] Q. Le and T. Mikolov, Distributed Representations of Sentences and Documents, in: *Proceedings of the 31st International Conference on Machine Learning*, E.P. Xing and T. Jebara, eds, Proceedings of Machine Learning Research, Vol. 32, PMLR, Bejing, China, 2014, pp. 1188–1196. https://proceedings.mlr.press/v32/le14.html.

[89] D.-H. Lee, Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks, in: *Workshop on challenges in representation learning, ICML*, Vol. 3, JMLR, Atlanta, USA, 2013, p. 2. http://deeplearning.net/wp-content/uploads/2013/03/pseudo_label_final.pdf.

[90] J. Lee, S.S. Yi, M. Jeong, M. Sung, W. Yoon, Y. Choi, M. Ko and J. Kang, Answering Questions on COVID-19 in Real-Time, in: *Proceedings of the 1st Workshop on NLP for COVID-19 (Part 2) at EMNLP 2020*, Association for Computational Linguistics, Online, 2020. doi:10.18653/v1/2020.nlpcovid19-2.1. https://aclanthology.org/2020.nlpcovid19-2.1.

[91] S. Lee, D. Kim, K. Lee, J. Choi, S. Kim, M. Jeon, S. Lim, D. Choi, S. Kim, A.-C. Tan and J. Kang, BEST: Next-Generation Biomedical Entity Search Tool for Knowledge Discovery from Biomedical Literature, *PLOS ONE* **11**(10) (2016), 1–16. doi:10.1371/journal.pone.0164680.

[92] J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P.N. Mendes, S. Hellmann, M. Morsey, P. van Kleef, S. Auer and C. Bizer, DBpedia - A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia, *Semantic Web Journal* **6**(2) (2015), 167–195, doi:10.3233/SW-140134. https://content.iospress.com/articles/semantic-web/sw134.

[93] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov and L. Zettlemoyer, BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension, in: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, Online, 2020, pp. 7871–7880. doi:10.18653/v1/2020.acl-main.703. https://aclanthology.org/2020.acl-main.703.

[94] B.Z. Li, S. Min, S. Iyer, Y. Mehdad and W.-t. Yih, Efficient One-Pass End-to-End Entity Linking for Questions, in: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Association for Computational Linguistics, Online, 2020, pp. 6433–6441. doi:10.18653/v1/2020.emnlp-main.522. https://aclanthology.org/2020.emnlp-main.522.

[95] J. Li, A. Sun, J. Han and C. Li, A Survey on Deep Learning for Named Entity Recognition, *IEEE Transactions on Knowledge and Data Engineering* **34**(1) (2022), 50–70. doi:10.1109/TKDE.2020.2981314.

[96] X. Ling and D.S. Weld, Fine-Grained Entity Recognition, in: *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, AAAI'12, AAAI Press, 2012, pp. 94–100–. https://ojs.aaai.org/index.php/AAAI/article/view/8122.

[97] X. Ling, S. Singh and D.S. Weld, Design Challenges for Entity Linking, *Transactions of the Association for Computational Linguistics* **3** (2015), 315–328. doi:10.1162/tacl_a_00141. https://aclanthology.org/Q15-1023.

[98] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer and V. Stoyanov, Ro{BERT}a: A Robustly Optimized {BERT} Pretraining Approach, 2020. https://openreview.net/forum?id=SyxS0T4tvS.

[99] R. Livni, S. Shalev-Shwartz and O. Shamir, On the Computational Efficiency of Training Neural Networks, in: *Advances in Neural Information Processing Systems*, Vol. 27, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence and K.Q. Weinberger, eds, Curran Associates, Inc., 2014, pp. 855–863–. https://proceedings.neurips.cc/paper/2014/file/3a0772443a0739141292a5429b952fe6-Paper.pdf.

[100] L. Logeswaran, M.-W. Chang, K. Lee, K. Toutanova, J. Devlin and H. Lee, Zero-Shot Entity Linking by Reading Entity Descriptions, in: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, Florence, Italy, 2019, pp. 3449–3460. doi:10.18653/v1/P19-1335. https://aclanthology.org/P19-1335.

[101] D. Loureiro and A.M. Jorge, MedLinker: Medical Entity Linking with Neural Representations and Dictionary Matching, in: *Advances in Information Retrieval*, J.M. Jose, E. Yilmaz, J. Magalhães, P. Castells, N. Ferro, M.J. Silva and F. Martins, eds, Springer International Publishing, Cham, 2020, pp. 230–237. ISBN 978-3-030-45442-5. doi:10.1007/978-3-030-45442-5_29.

[102] G. Luo, X. Huang, C.-Y. Lin and Z. Nie, Joint Entity Recognition and Disambiguation, in: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Lisbon, Portugal, 2015, pp. 879–888. doi:10.18653/v1/D15-1104. https://aclanthology.org/D15-1104.

[103] G. Luo, T. Darrell and A. Rohrbach, NewsCLIPpings: Automatic Generation of Out-of-Context Multimodal Media, in: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Online and Punta Cana, Dominican Republic, 2021, pp. 6801–6817. doi:10.18653/v1/2021.emnlp-main.545. https://aclanthology.org/2021.emnlp-main.545.

[104] S. MacAvaney, S. Sotudeh, A. Cohan, N. Goharian, I. Talati and R.W. Filice, Ontology-Aware Clinical Abstractive Summarization, in: *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'19, Association for Computing Machinery, New York, NY, USA, 2019, pp. 1013–1016–. ISBN 9781450361729. doi:10.1145/3331184.3331319.

[105] C.D. Manning, P. Raghavan and H. Schütze, *Introduction to Information Retrieval*, Cambridge University Press, USA, 2008. ISBN 0521865719.

[106] J.L. Martínez-Rodríguez, A. Hogan and I. López-Arévalo, Information extraction meets the Semantic Web: A survey, *Semantic Web* **11**(2) (2020), 255–335. doi:10.3233/SW-180333. https://content.iospress.com/articles/semantic-web/sw180333.

[107] P.H. Martins, Z. Marinho and A.F.T. Martins, Joint Learning of Named Entity Recognition and Entity Linking, in: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, Association for Computational Linguistics, Florence, Italy, 2019, pp. 190–196. doi:10.18653/v1/P19-2026. https://aclanthology.org/P19-2026.

[108] P.N. Mendes, M. Jakob, A. García-Silva and C. Bizer, DBpedia Spotlight: Shedding Light on the Web of Documents, in: *Proceedings of the 7th International Conference on Semantic Systems*, I-Semantics '11, Association for Computing Machinery, New York, NY, USA, 2011, pp. 1–8–. ISBN 9781450306218. doi:10.1145/2063518.2063519.

[109] Z. Miftahutdinov, A. Kadurin, R. Kudrin and E. Tutubalina, Drug and Disease Interpretation Learning with Biomedical Entity Representation Transformer, in: *Advances in Information Retrieval*, D. Hiemstra, M.-F. Moens, J. Mothe, R. Perego, M. Potthast and F. Sebastiani, eds, Springer International Publishing, Cham, 2021, pp. 451–466. ISBN 978-3-030-72113-8. doi:10.1007/978-3-030-72113-8_30.

[110] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado and J. Dean, Distributed Representations of Words and Phrases and Their Compositionality, in: *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'13, Curran Associates Inc., Red Hook, NY, USA, 2013a, pp. 3111–3119–. https://papers.nips.cc/paper/2013/file/9aa42b31882ec039965f3c4923ce901b-Paper.pdf.

[111] T. Mikolov, K. Chen, G.S. Corrado and J. Dean, Efficient Estimation of Word Representations in Vector Space, in: *1st International Conference on Learning Representations, ICLR 2013*, Scottsdale, Arizona, USA, 2013b.

[112] D. Milne and I.H. Witten, Learning to Link with Wikipedia, in: *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, CIKM '08, Association for Computing Machinery, New York, NY, USA, 2008, pp. 509–518–. ISBN 9781595939913. doi:10.1145/1458082.1458150.

[113] S. Moon, L. Neves and V. Carvalho, Multimodal Named Entity Disambiguation for Noisy Social Media Posts, in: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, Melbourne, Australia, 2018, pp. 2000–2008. doi:10.18653/v1/P18-1186. https://aclanthology.org/P18-1186.

[114] J.G. Moreno, R. Besançon, R. Beaumont, E. D'hondt, A.-L. Ligozat, S. Rosset, X. Tannier and B. Grau, Combining Word and Entity Embeddings for Entity Linking, in: *The Semantic Web*, E. Blomqvist, D. Maynard, A. Gangemi, R. Hoekstra, P. Hitzler and O. Hartig, eds, Springer International Publishing, Cham, 2017, pp. 337–352. ISBN 978-3-319-58068-5. doi:10.1007/978-3-319-58068-5_21.

[115] A. Moro, A. Raganato and R. Navigli, Entity Linking meets Word Sense Disambiguation: a Unified Approach, *Transactions of the Association for Computational Linguistics* **2** (2014), 231–244. doi:10.1162/tacl_a_00179. https://aclanthology.org/Q14-1019.

[116] D. Mueller and G. Durrett, Effective Use of Context in Noisy Entity Linking, in: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Brussels, Belgium, 2018, pp. 1024–1029. doi:10.18653/v1/D18-1126. https://aclanthology.org/D18-1126.

[117] I.O. Mulang', K. Singh, C. Prabhu, A. Nadgeri, J. Hoffart and J. Lehmann, Evaluating the Impact of Knowledge Graph Context on Entity Disambiguation Models, in: *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, CIKM '20, Association for Computing Machinery, New York, NY, USA, 2020, pp. 2157–2160–. ISBN 9781450368599. doi:10.1145/3340531.3412159.

[118] C. Möller, J. Lehmann and R. Usbeck, Survey on English Entity Linking on Wikidata: Datasets and approaches, Vol. Pre-press, IOS Press, 2022, pp. 1–42. doi:10.3233/SW-212865. https://content.iospress.com/articles/semantic-web/sw212865.

[119] D. Nadeau and S. Sekine, A survey of named entity recognition and classification, *Lingvisticæ Investigationes* **30**(1) (2007), 3–26. doi:10.1075/li.30.1.03nad.

[120] A. Nagrani, S. Yang, A. Arnab, A. Jansen, C. Schmid and C. Sun, Attention Bottlenecks for Multimodal Fusion, in: *Advances in Neural Information Processing Systems*, A. Beygelzimer, Y. Dauphin, P. Liang and J.W. Vaughan, eds, 2021. https://openreview.net/forum?id=KJ5h-yfUHa.

[121] R. Navigli, Word Sense Disambiguation: A Survey, *ACM Computing Surveys* **41**(2) (2009). doi:10.1145/1459352.1459355.

[122] M. Nayyeri, S. Vahdati, J. Lehmann and H.S. Yazdi, Soft Marginal TransE for Scholarly Knowledge Graph Completion, *CoRR* **abs/1904.12211** (2019). http://arxiv.org/abs/1904.12211.

[123] M. Nayyeri, S. Vahdati, C. Aykul and J. Lehmann, 5* Knowledge Graph Embeddings with Projective Transformations, *Proceedings of the AAAI Conference on Artificial Intelligence* **35**(10) (2021), 9064–9072. https://ojs.aaai.org/index.php/AAAI/article/view/17095.

[124] R. Nedelchev, D. Chaudhuri, J. Lehmann and A. Fischer, End-to-End Entity Linking and Disambiguation lever-

aging Word and Knowledge Graph Embeddings, *CoRR* **abs/2002.11143** (2020). https://arxiv.org/abs/2002.11143.

[125] D. Newman-Griffis, A.M. Lai and E. Fosler-Lussier, Jointly Embedding Entities and Text with Distant Supervision, in: *Proceedings of The Third Workshop on Representation Learning for NLP*, Association for Computational Linguistics, Melbourne, Australia, 2018, pp. 195–206. doi:10.18653/v1/W18-3026. https://aclanthology.org/W18-3026.

[126] D.B. Nguyen, M. Theobald and G. Weikum, J-NERD: Joint Named Entity Recognition and Disambiguation with Rich Linguistic Features, *Transactions of the Association for Computational Linguistics* **4** (2016a), 215–229. doi:10.1162/tacl_-a_00094. https://aclanthology.org/Q16-1016.

[127] T.H. Nguyen, N. Fauceglia, M. Rodriguez Muro, O. Hassanzadeh, A. Massimiliano Gliozzo and M. Sadoghi, Joint Learning of Local and Global Features for Entity Linking via Neural Networks, in: *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, The COLING 2016 Organizing Committee, Osaka, Japan, 2016b, pp. 2310–2320. https://aclanthology.org/C16-1218.

[128] M. Nickel, V. Tresp and H.-P. Kriegel, A Three-Way Model for Collective Learning on Multi-Relational Data, in: *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, Omnipress, Madison, WI, USA, 2011, pp. 809–816–. ISBN 9781450306195.

[129] F. Nie, Y. Cao, J. Wang, C.-Y. Lin and R. Pan, Mention and Entity Description Co-Attention for Entity Disambiguation, *Proceedings of the AAAI Conference on Artificial Intelligence* **32**(1) (2018). https://ojs.aaai.org/index.php/AAAI/article/view/12043.

[130] I.L. Oliveira, R. Fileto, R. Speck, L.P.F. Garcia, D. Moussallem and J. Lehmann, Towards holistic Entity Linking: Survey and directions, *Information Systems* **95** (2021), 101624. doi:10.1016/j.is.2020.101624. https://www.sciencedirect.com/science/article/pii/S0306437920300958.

[131] Y. Onoe and G. Durrett, Fine-Grained Entity Typing for Domain Independent Entity Linking, *Proceedings of the AAAI Conference on Artificial Intelligence* **34**(05) (2020), 8576–8583. doi:10.1609/aaai.v34i05.6380. https://ojs.aaai.org/index.php/AAAI/article/view/6380.

[132] P. Orponen, Computational Complexity of Neural Networks: A Survey, *Nordic Journal of Computing* **1**(1) (1994), 94–110–.

[133] L. Page, S. Brin, R. Motwani and T. Winograd, The PageRank Citation Ranking: Bringing Order to the Web., Technical Report, 1999-66, Stanford InfoLab, 1999, Previous number = SIDL-WP-1999-0120. http://ilpubs.stanford.edu:8090/422/.

[134] X. Pan, B. Zhang, J. May, J. Nothman, K. Knight and H. Ji, Cross-lingual Name Tagging and Linking for 282 Languages, in: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, Vancouver, Canada, 2017, pp. 1946–1958. doi:10.18653/v1/P17-1178. https://aclanthology.org/P17-1178.

[135] A. Parravicini, R. Patra, D.B. Bartolini and M.D. Santambrogio, 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)*, GRADES-NDA'19, Association for Computing Machinery, New York, NY, USA, 2019. ISBN 9781450367899. doi:10.1145/3327964.3328499.

[136] B. Perozzi, R. Al-Rfou and S. Skiena, DeepWalk: Online Learning of Social Representations, in: *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, Association for Computing Machinery, New York, NY, USA, 2014, pp. 701–710–. ISBN 9781450329569. doi:10.1145/2623330.2623732.

[137] M. Pershina, Y. He and R. Grishman, Personalized Page Rank for Named Entity Disambiguation, in: *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Association for Computational Linguistics, Denver, Colorado, 2015, pp. 238–243. doi:10.3115/v1/N15-1026. https://aclanthology.org/N15-1026.

[138] M.E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee and L. Zettlemoyer, Deep Contextualized Word Representations, in: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, Association for Computational Linguistics, New Orleans, Louisiana, 2018, pp. 2227–2237. doi:10.18653/v1/N18-1202. https://aclanthology.org/N18-1202.

[139] M.E. Peters, M. Neumann, R. Logan, R. Schwartz, V. Joshi, S. Singh and N.A. Smith, Knowledge Enhanced Contextual Word Representations, in: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Association for Computational Linguistics, Hong Kong, China, 2019, pp. 43–54. doi:10.18653/v1/D19-1005. https://aclanthology.org/D19-1005.

[140] F. Piccinno and P. Ferragina, From TagME to WAT: A New Entity Annotator, in: *Proceedings of the First International Workshop on Entity Recognition &; Disambiguation*, ERD '14, Association for Computing Machinery, New York, NY, USA, 2014, pp. 55–62–. ISBN 9781450330237. doi:10.1145/2633211.2634350.

[141] T. Pires, E. Schlinger and D. Garrette, How Multilingual is Multilingual BERT?, in: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, Florence, Italy, 2019, pp. 4996–5001. doi:10.18653/v1/P19-1493. https://aclanthology.org/P19-1493.

[142] N. Poerner, U. Waltinger and H. Schütze, E-BERT: Efficient-Yet-Effective Entity Embeddings for BERT, in: *Findings of the Association for Computational Linguistics: EMNLP 2020*, Association for Computational Linguistics, Online, 2020, pp. 803–818. doi:10.18653/v1/2020.findings-emnlp.71. https://aclanthology.org/2020.findings-emnlp.71.

[143] V. Provatorova, S. Vakulenko, E. Kanoulas, K. Dercksen and J.M. van Hulst, Named Entity Recognition and Linking on Historical Newspapers: UvA.ILPS & REL at CLEF HIPE 2020, in: *Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum*, L. Cappellato, C. Eickhoff, N. Ferro and A. Névéol, eds, CEUR Workshop Proceedings, Vol. 2696, CEUR-WS.org, Thessaloniki, Greece, 2020. http://ceur-ws.org/Vol-2696/paper_209.pdf.

[144] P. Radhakrishnan, P. Talukdar and V. Varma, ELDEN: Improved Entity Linking Using Densified Knowledge Graphs, in: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, Association for Computational Linguistics, New Orleans, Louisiana, 2018, pp. 1844–1853. doi:10.18653/v1/N18-1167. https://aclanthology.org/N18-1167.

[145] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li and P.J. Liu, Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer, *Journal of Machine Learning Research* **21**(140) (2020), 1–67. http://jmlr.org/papers/v21/20-074.html.

[146] J. Raiman and O. Raiman, DeepType: Multilingual Entity Linking by Neural Type System Evolution, in: *AAAI Conference on Artificial Intelligence*, New Orleans, Louisiana, USA., 2018. https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/17148.

[147] P. Rajpurkar, J. Zhang, K. Lopyrev and P. Liang, SQuAD: 100,000+ Questions for Machine Comprehension of Text, in: *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Austin, Texas, 2016, pp. 2383–2392. doi:10.18653/v1/D16-1264. https://aclanthology.org/D16-1264.

[148] L. Ratinov, D. Roth, D. Downey and M. Anderson, Local and Global Algorithms for Disambiguation to Wikipedia, in: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, Association for Computational Linguistics, USA, 2011, pp. 1375–1384–. ISBN 9781932432879. http://dl.acm.org/citation.cfm?id=2002472.2002642.

[149] N. Reimers and I. Gurevych, Reporting Score Distributions Makes a Difference: Performance Study of LSTM-networks for Sequence Tagging, in: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Copenhagen, Denmark, 2017, pp. 338–348. doi:10.18653/v1/D17-1035. https://aclanthology.org/D17-1035.

[150] N. Reimers and I. Gurevych, Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks, in: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Association for Computational Linguistics, Hong Kong, China, 2019, pp. 3982–3992. doi:10.18653/v1/D19-1410. https://aclanthology.org/D19-1410.

[151] S. Rijhwani, J. Xie, G. Neubig and J. Carbonell, Zero-Shot Neural Transfer for Cross-Lingual Entity Linking, *Proceedings of the AAAI Conference on Artificial Intelligence* **33**(01) (2019), 6924–6931. doi:10.1609/aaai.v33i01.33016924. https://ojs.aaai.org/index.php/AAAI/article/view/4670.

[152] G. Rizzo, M. van Erp and R. Troncy, Benchmarking the Extraction and Disambiguation of Named Entities on the Semantic Web, in: *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, European Language Resources Association (ELRA), Reykjavik, Iceland, 2014, pp. 4593–4600. http://www.lrec-conf.org/proceedings/lrec2014/pdf/176_Paper.pdf.

[153] M. Röder, R. Usbeck and A.N. Ngomo, GERBIL - Benchmarking Named Entity Recognition and Linking consistently, *Semantic Web* **9**(5) (2018), 605 – 625–. doi:10.3233/SW-170286. https://content.iospress.com/articles/semantic-web/sw286.

[154] D. Ruffinelli, S. Broscheit and R. Gemulla, You CAN Teach an Old Dog New Tricks! On Training Knowledge Graph Embeddings, in: *International Conference on Learning Representations*, 2020. https://openreview.net/forum?id=BkxSmlBFvr.

[155] G.K. Savova, J.J. Masanz, P.V. Ogren, J. Zheng, S. Sohn, K.C. Kipper-Schuler and C.G. Chute, Mayo clinical Text Analysis and Knowledge Extraction System (cTAKES): architecture, component evaluation and applications, *Journal of the American Medical Informatics Association* **17**(5) (2010), 507–513. doi:10.1136/jamia.2009.001560.

[156] Ö. Sevgili, A. Panchenko and C. Biemann, Improving Neural Entity Disambiguation with Graph Embeddings, in: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, Association for Computational Linguistics, Florence, Italy, 2019, pp. 315–322. doi:10.18653/v1/P19-2044. https://aclanthology.org/P19-2044.

[157] H. Shahbazi, X. Fern, R. Ghaeini, C. Ma, R.M. Obeidat and P. Tadepalli, Joint Neural Entity Disambiguation with Output Space Search, in: *Proceedings of the 27th International Conference on Computational Linguistics*, Association for Computational Linguistics, Santa Fe, New Mexico, USA, 2018, pp. 2170–2180. https://aclanthology.org/C18-1184.

[158] H. Shahbazi, X.Z. Fern, R. Ghaeini, R. Obeidat and P. Tadepalli, Entity-aware ELMo: Learning Contextual Entity Representation for Entity Disambiguation, *arXiv preprint arXiv:1908.05762* (2019). https://arxiv.org/abs/1908.05762.

[159] R. Sharnagat, Named entity recognition: A literature survey, *Center For Indian Language Technology* (2014). http://www.cfilt.iitb.ac.in/resources/surveys/rahul-ner-survey.pdf.

[160] W. Shen, J. Wang and J. Han, Entity Linking with a Knowledge Base: Issues, Techniques, and Solutions, *IEEE Transactions on Knowledge & Data Engineering* **27**(02) (2015), 443–460. doi:10.1109/TKDE.2014.2327028.

[161] W. Shen, J. Han, J. Wang, X. Yuan and Z. Yang, SHINE+: A General Framework for Domain-Specific Entity Linking with Heterogeneous Information Networks, *IEEE Transactions on Knowledge and Data Engineering* **30**(2) (2018), 353–366. doi:10.1109/TKDE.2017.2730862.

[162] W. Shi, S. Zhang, Z. Zhang, H. Cheng and J.X. Yu, Joint Embedding in Named Entity Linking on Sentence Level, *arXiv preprint arXiv:2002.04936* (2020). https://arxiv.org/abs/2002.04936.

[163] I. Shnayderman, L. Ein-Dor, Y. Mass, A. Halfon, B. Sznajder, A. Spector, Y. Katz, D. Sheinwald, R. Aharonov and N. Slonim, Fast End-to-End Wikification, *arXiv preprint arXiv:1908.06785* (2019).

[164] A. Sil, G. Kundu, R. Florian and W. Hamza, Neural Cross-Lingual Entity Linking, *Proceedings of the AAAI Conference on Artificial Intelligence* **32**(1) (2018). https://ojs.aaai.org/index.php/AAAI/article/view/11964.

[165] J. Šíma and P. Orponen, General-Purpose Computation with Neural Networks: A Survey of Complexity Theoretic Results, *Neural Computation* **15**(12) (2003), 2727–2778. doi:10.1162/089976603322518731.

[166] N. Slonim, Y. Bilu, C. Alzate, R. Bar-Haim, B. Bogin, F. Bonin, L. Choshen, E. Cohen-Karlik, L. Dankin, L. Edelstein et al., An autonomous debating system, *Nature* **591**(7850) (2021), 379–384. doi:10.1038/s41586-021-03215-w.

[167] L. Soldaini and N. Goharian, QuickUMLS: a fast, unsupervised approach for medical concept extraction, in: *MedIR workshop, SIGIR*, 2016, pp. 1–4. http://medir2016.imag.fr/data/MEDIR_2016_paper_16.pdf.

[168] D. Sorokin and I. Gurevych, Mixing Context Granularities for Improved Entity Linking on Question Answering Data across Entity Categories, in: *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, Association for Computational Linguistics, New Orleans, Louisiana, 2018, pp. 65–75. doi:10.18653/v1/S18-2007. https://aclanthology.org/S18-2007.

[169] V.I. Spitkovsky and A.X. Chang, A Cross-Lingual Dictionary for English Wikipedia Concepts, in: *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, European Language Resources Association (ELRA), Istanbul, Turkey, 2012, pp. 3168–3175. http://www.lrec-conf.org/proceedings/lrec2012/pdf/266_Paper.pdf.

[170] F.M. Suchanek, G. Kasneci and G. Weikum, YAGO: A Core of Semantic Knowledge, in: *Proceedings of the 16th International Conference on World Wide Web*, WWW '07, Association for Computing Machinery, New York, NY, USA, 2007, pp. 697–706–. ISBN 9781595936547. doi:10.1145/1242572.1242667.

[171] Y. Sun, L. Lin, D. Tang, N. Yang, Z. Ji and X. Wang, Modeling Mention, Context and Entity with Neural Networks for Entity Disambiguation, in: *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, AAAI Press, 2015, pp. 1333–1339–. ISBN 9781577357384.

[172] M. Sung, H. Jeon, J. Lee and J. Kang, Biomedical Entity Representations with Synonym Marginalization, in: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, Online, 2020, pp. 3641–3650. doi:10.18653/v1/2020.acl-main.335. https://www.aclweb.org/anthology/2020.acl-main.335.

[173] H. Tang, X. Sun, B. Jin and F. Zhang, A Bidirectional Multi-paragraph Reading Model for Zero-shot Entity Linking, 2021, pp. 13889–13897. https://ojs.aaai.org/index.php/AAAI/article/view/17636.

[174] E.F. Tjong Kim Sang and F. De Meulder, Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition, in: *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, 2003, pp. 142–147. https://aclanthology.org/W03-0419.

[175] T. Trouillon, J. Welbl, S. Riedel, E. Gaussier and G. Bouchard, Complex Embeddings for Simple Link Prediction, in: *Proceedings of The 33rd International Conference on Machine Learning*, M.F. Balcan and K.Q. Weinberger, eds, Proceedings of Machine Learning Research, Vol. 48, PMLR, New York, New York, USA, 2016, pp. 2071–2080. https://proceedings.mlr.press/v48/trouillon16.html.

[176] C.-T. Tsai and D. Roth, Cross-lingual Wikification Using Multilingual Embeddings, in: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technolo-*gies, Association for Computational Linguistics, San Diego, California, 2016, pp. 589–598. doi:10.18653/v1/N16-1072. https://aclanthology.org/N16-1072.

[177] C.-T. Tsai and D. Roth, Learning Better Name Translation for Cross-Lingual Wikification, *Proceedings of the AAAI Conference on Artificial Intelligence* **32**(1) (2018). https://ojs.aaai.org/index.php/AAAI/article/view/12018.

[178] E. Tutubalina, A. Kadurin and Z. Miftahutdinov, Fair Evaluation in Concept Normalization: a Large-scale Comparative Analysis for BERT-based Models, in: *Proceedings of the 28th International Conference on Computational Linguistics*, International Committee on Computational Linguistics, Barcelona, Spain (Online), 2020, pp. 6710–6716. doi:10.18653/v1/2020.coling-main.588. https://aclanthology.org/2020.coling-main.588.

[179] S. Upadhyay, N. Gupta and D. Roth, Joint Multilingual Supervision for Cross-lingual Entity Linking, in: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Brussels, Belgium, 2018a, pp. 2486–2495. doi:10.18653/v1/D18-1270. https://aclanthology.org/D18-1270.

[180] S. Upadhyay, J. Kodner and D. Roth, Bootstrapping Transliteration with Constrained Discovery for Low-Resource Languages, in: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Brussels, Belgium, 2018b, pp. 501–511. doi:10.18653/v1/D18-1046. https://aclanthology.org/D18-1046.

[181] J.M. van Hulst, F. Hasibi, K. Dercksen, K. Balog and A.P. de Vries, *REL: An Entity Linker Standing on the Shoulders of Giants*, in: *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, Association for Computing Machinery, New York, NY, USA, 2020, pp. 2197–2200–. ISBN 9781450380164. https://doi.org/10.1145/3397271.3401416.

[182] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser and I. Polosukhin, Attention is All You Need, in: *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, Curran Associates Inc., Red Hook, NY, USA, 2017, pp. 6000–6010–. ISBN 9781510860964. https://papers.nips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[183] O. Vinyals, M. Fortunato and N. Jaitly, Pointer Networks, in: *Advances in Neural Information Processing Systems 28*, C. Cortes, N.D. Lawrence, D.D. Lee, M. Sugiyama and R. Garnett, eds, Curran Associates, Inc., 2015, pp. 2692–2700. http://papers.nips.cc/paper/5866-pointer-networks.pdf.

[184] D. Vrandečić and M. Krötzsch, Wikidata: A Free Collaborative Knowledgebase, *Communications of the ACM* **57**(10) (2014), 78–85–. doi:10.1145/2629489.

[185] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy and S. Bowman, GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding, in: *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, Association for Computational Linguistics, Brussels, Belgium, 2018, pp. 353–355. doi:10.18653/v1/W18-5446. https://aclanthology.org/W18-5446.

[186] H. Wang, J.G. Zheng, X. Ma, P. Fox and H. Ji, Language and Domain Independent Entity Linking with Quantified Collective Validation, in: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Lisbon, Portugal, 2015, pp. 695–704. doi:10.18653/v1/D15-1081. https://aclanthology.org/D15-1081.

[187] Q. Wang, Z. Mao, B. Wang and L. Guo, Knowledge Graph Embedding: A Survey of Approaches and Applications, *IEEE Transactions on Knowledge and Data Engineering* **29**(12) (2017), 2724–2743. doi:10.1109/TKDE.2017.2754499.

[188] X. Wang, T. Gao, Z. Zhu, Z. Zhang, Z. Liu, J. Li and J. Tang, KEPLER: A Unified Model for Knowledge Embedding and Pre-trained Language Representation, *Transactions of the Association for Computational Linguistics* **9** (2021), 176–194. doi:10.1162/tacl_a_00360. https://doi.org/10.1162/tacl_a_00360.

[189] Z. Wang, J. Zhang, J. Feng and Z. Chen, Knowledge Graph Embedding by Translating on Hyperplanes, *Proceedings of the AAAI Conference on Artificial Intelligence* **28**(1) (2014). https://ojs.aaai.org/index.php/AAAI/article/view/8870.

[190] J. Wu, R. Zhang, Y. Mao, H. Guo, M. Soflaei and J. Huai, Dynamic Graph Convolutional Networks for Entity Linking, in: *Proceedings of The Web Conference 2020*, WWW '20, Association for Computing Machinery, New York, NY, USA, 2020a, pp. 1149–1159–. ISBN 9781450370233. doi:10.1145/3366423.3380192.

[191] L. Wu, F. Petroni, M. Josifoski, S. Riedel and L. Zettlemoyer, Scalable Zero-shot Entity Linking with Dense Entity Retrieval, in: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Association for Computational Linguistics, Online, 2020b, pp. 6397–6407. doi:10.18653/v1/2020.emnlp-main.519. https://aclanthology.org/2020.emnlp-main.519.

[192] M. Xue, W. Cai, J. Su, L. Song, Y. Ge, Y. Liu and B. Wang, Neural Collective Entity Linking Based on Recurrent Random Walk Network Learning, in: *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, International Joint Conferences on Artificial Intelligence Organization, 2019, pp. 5327–5333. doi:10.24963/ijcai.2019/740.

[193] V. Yadav and S. Bethard, A Survey on Recent Advances in Named Entity Recognition from Deep Learning models, in: *Proceedings of the 27th International Conference on Computational Linguistics*, Association for Computational Linguistics, Santa Fe, New Mexico, NM, USA, 2018, pp. 2145–2158. https://www.aclweb.org/anthology/C18-1182.

[194] I. Yamada, H. Shindo, H. Takeda and Y. Takefuji, Joint Learning of the Embedding of Words and Entities for Named Entity Disambiguation, in: *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, Association for Computational Linguistics, Berlin, Germany, 2016, pp. 250–259. doi:10.18653/v1/K16-1025. https://aclanthology.org/K16-1025.

[195] I. Yamada, H. Shindo, H. Takeda and Y. Takefuji, Learning Distributed Representations of Texts and Entities from Knowledge Base, *Transactions of the Association for Computational Linguistics* **5** (2017), 397–411. doi:10.1162/tacl_a_00069. https://aclanthology.org/Q17-1028.

[196] I. Yamada, A. Asai, H. Shindo, H. Takeda and Y. Matsumoto, LUKE: Deep Contextualized Entity Representations with Entity-aware Self-attention, in: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Association for Computational Linguistics, Online, 2020a, pp. 6442–6454. doi:10.18653/v1/2020.emnlp-main.523. https://aclanthology.org/2020.emnlp-main.523.

[197] I. Yamada, A. Asai, J. Sakuma, H. Shindo, H. Takeda, Y. Takefuji and Y. Matsumoto, Wikipedia2Vec: An Efficient Toolkit for Learning and Visualizing the Embeddings of Words and Entities from Wikipedia, in: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, Association for Computational Linguistics, Online, 2020b, pp. 23–30. doi:10.18653/v1/2020.emnlp-demos.4. https://aclanthology.org/2020.emnlp-demos.4.

[198] I. Yamada, K. Washio, H. Shindo and Y. Matsumoto, Global Entity Disambiguation with Pretrained Contextualized Embeddings of Words and Entities, *arXiv preprint arXiv:1909.00426v3* (2021). https://arxiv.org/abs/1909.00426v3.

[199] B. Yang, S.W.-t. Yih, X. He, J. Gao and L. Deng, Embedding Entities and Relations for Learning and Inference in Knowledge Bases, in: *Proceedings of the International Conference on Learning Representations (ICLR) 2015*, Proceedings of the international conference on learning representations (iclr) 2015 edn, 2015. https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/ICLR2015_updated.pdf.

[200] X. Yang, X. Gu, S. Lin, S. Tang, Y. Zhuang, F. Wu, Z. Chen, G. Hu and X. Ren, Learning Dynamic Context Augmentation for Global Entity Linking, in: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Association for Computational Linguistics, Hong Kong, China, 2019, pp. 271–281. doi:10.18653/v1/D19-1026. https://aclanthology.org/D19-1026.

[201] Z. Yao, L. Cao and H. Pan, 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, 2020, pp. 2517–2522. doi:10.18653/v1/2020.findings-emnlp.228. https://aclanthology.org/2020.findings-emnlp.228.

[202] W.-t. Yih, M.-W. Chang, X. He and J. Gao, Semantic Parsing via Staged Query Graph Generation: Question Answering with Knowledge Base, in: *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Association for Computational Linguistics, Beijing, China, 2015, pp. 1321–1331. doi:10.3115/v1/P15-1128. https://aclanthology.org/P15-1128.

[203] T. Young, D. Hazarika, S. Poria and E. Cambria, Recent Trends in Deep Learning Based Natural Language Processing [Review Article], *IEEE Computational Intelligence Magazine* **13**(3) (2018), 55–75. doi:10.1109/MCI.2018.2840738.

[204] S. Zhang, X. Liu, J. Liu, J. Gao, K. Duh and B.V. Durme, ReCoRD: Bridging the Gap between Human and Machine Commonsense Reading Comprehension, *CoRR* **abs/1810.12885** (2018). http://arxiv.org/abs/1810.12885.

[205] Y. Zhang, V. Zhong, D. Chen, G. Angeli and C.D. Manning, Position-aware Attention and Supervised Data Improve Slot Filling, in: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Copenhagen, Denmark, 2017, pp. 35–45. doi:10.18653/v1/D17-1004. https://aclanthology.org/D17-1004.

[206] Z. Zhang, X. Han, Z. Liu, X. Jiang, M. Sun and Q. Liu, ERNIE: Enhanced Language Representation with Informative Entities, in: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, Florence, Italy, 2019, pp. 1441–1451. doi:10.18653/v1/P19-1139. https://aclanthology.org/P19-1139.

[207] S. Zhou, S. Rijhwani and G. Neubig, Towards Zero-resource Cross-lingual Entity Linking, in: *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019)*, Association for Computational Linguistics, Hong Kong, China, 2019, pp. 243–252. doi:10.18653/v1/D19-6127. https://aclanthology.org/D19-6127.

[208] S. Zhou, S. Rijhwani, J. Wieting, J. Carbonell and G. Neubig, Improving Candidate Generation for Low-resource Cross-lingual Entity Linking, *Transactions of the Association for Computational Linguistics* **8** (2020), 109–124. doi:10.1162/tacl_a_00303. https://doi.org/10.1162/tacl_a_00303.

[209] M. Zhu, B. Celikkaya, P. Bhatia and C.K. Reddy, LATTE: Latent Type Modeling for Biomedical Entity Linking, *Proceedings of the AAAI Conference on Artificial Intelligence* **34**(05) (2020), 9757–9764. doi:10.1609/aaai.v34i05.6526. https://ojs.aaai.org/index.php/AAAI/article/view/6526.

[210] S. Zwicklbauer, C. Seifert and M. Granitzer, DoSeR - A Knowledge-Base-Agnostic Framework for Entity Disambiguation Using Semantic Embeddings, in: *The Semantic Web. Latest Advances and New Domains*, H. Sack, E. Blomqvist, M. d'Aquin, C. Ghidini, S.P. Ponzetto and C. Lange, eds, Springer International Publishing, Cham, 2016a, pp. 182–198. ISBN 978-3-319-34129-3. doi:10.1007/978-3-319-34129-3_12.

[211] S. Zwicklbauer, C. Seifert and M. Granitzer, Robust and Collective Entity Disambiguation through Semantic Embeddings, in: *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '16, Association for Computing Machinery, New York, NY, USA, 2016b, pp. 425–434–. ISBN 9781450340694. doi:10.1145/2911451.2911535.