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

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

[a] *Language Technology Group, Department of Informatics, Universität Hamburg, Germany*
E-mails: sevgili@informatik.uni-hamburg.de, biemann@informatik.uni-hamburg.de
[b] *Center for Data Intensive Science and Engineering, Skolkovo Institute of Science and Technology, Russia*
E-mails: a.shelmanov@skoltech.ru, a.panchenko@skoltech.ru
[c] *Research Computing Center, Lomonosov Moscow State University, Russia*
[d] *Artificial Intelligence Research Institute, Russia*
[e] *Neural Networks and Deep Learning Laboratory, Moscow Institute of Physics and Technology, Russia*
E-mail: arkhipov@yahoo.com

Editors: First Editor, University or Company name, Country; Second Editor, University or Company name, Country
Solicited reviews: First Solicited Reviewer, University or Company name, Country; Second Solicited Reviewer, University or Company name, Country
Open reviews: First Open Reviewer, University or Company name, Country; Second Open Reviewer, University or Company name, Country

Abstract. In this survey, we provide a comprehensive description of recent neural entity linking (EL) systems developed since 2015 as a result of the "deep learning revolution" in NLP. Our goal is to systemize design features of neural entity linking systems and compare their performance to the prominent classic methods on common benchmarks. We distill generic architectural components of a neural EL system, like candidate generation and entity ranking, and summarize prominent methods for each of them. The vast variety of modifications of this general neural entity linking architecture are grouped by several common themes: joint entity recognition and linking, 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 catch semantic meaning of them, we provide an overview of popular embedding techniques. Finally, we briefly discuss 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

1. Introduction

Knowledge Graphs (KGs), such as Freebase [12], DBpedia [6], and Wikidata [154], 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

[*] Equal contribution. Corresponding author. E-mail: sevgili@informatik.uni-hamburg.de.

[**] Equal contribution. Corresponding author. E-mail: a.shelmanov@skoltech.ru.

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 (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 the neural models and deep learning emerged, pushing the state-of-the-art performance on this task to a new level. The goal of this survey is to provide an overview of this latest wave of models, emerging from 2015 until now.

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 [10, 23, 170]. The state-of-the-art neural entity linking models have shown significant improvements over "classical"[3] machine learning approaches [20, 72, 121] that are based on shallow architectures, e.g. Support Vector Machines, and/or depend mostly on hand-engineered features. Such models often cannot capture all relevant statistical dependencies and interactions [44]. In contrast, deep neural networks are able to learn sophisticated representations within their deep layered architectures. This both reduces the burden of manual feature engineering and enables significant improvements on EL and other tasks.

In this survey, we systematize recently proposed neural models, distilling one generic architecture commonly used by popular neural EL models (illustrated in Figures 2 and 5). We describe the models used in each component of this architecture, e.g. candidate generation or 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 the form of taxonomies (Figures 3 and 6) while notable features of each model are carefully assembled in a tabular form (Table 2).

An important component of neural entity linking systems is entity vector representations and entity encoding methods. It has been shown that encoding the KG structure (entity relationships), entity definitions, or textual information in large annotated corpora in low-dimensional vectors, improves the generalization capabilities of EL models [44, 59]. We summarize novel methods for entity encoding, as well as context/mention encoding techniques.

Many natural language processing systems take advantage of deep pre-trained language models like ELMo [113], BERT [29], and their modifications. EL made its path into these models as a way of introducing information stored in KGs, which helps to adopt word representations to some text processing tasks. We discuss this novel application of EL and how it can be further developed.

As with all surveys, we had to draw the line somewhere. The main criteria for including papers into this survey was that they have been published in or after 2015 and they primarily address the task of EL, i.e. resolving textual mentions to entries in KGs, and its applications. We explicitly exclude related work e.g. on (fine-grained) entity typing (see [3]), which also encompasses a disambiguation task, and work that employs KGs for other tasks than EL. Because of the sheer amount of work, it was not possible for us to try software if available, and to compare approaches on further parameters, such as computational complexity, run-time, and memory requirements.

[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

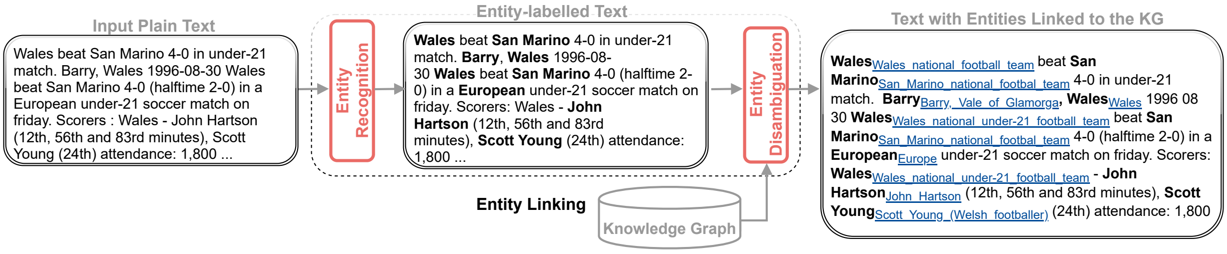
Input Plain Text
Wales beat San Marino 4-0 in under-21 match. Barry, Wales 1996-08-30 Wales beat San Marino 4-0 (halftime 2-0) in a European under-21 soccer match on friday. Scorers : Wales - John Hartson (12th, 56th and 83rd minutes), Scott Young (24th) attendance: 1,800 ...
Entity Recognition
Entity-labelled Text
Wales beat **San Marino** 4-0 in under-21 match. **Barry**, **Wales** 1996-08-30 **Wales** beat **San Marino** 4-0 (halftime 2-0) in a **European** under-21 soccer match on friday. Scorers: Wales - **John Hartson** (12th, 56th and 83rd minutes), **Scott Young** (24th) attendance: 1,800 ...
Entity Disambiguation
Entity Linking
Knowledge Graph
Text with Entities Linked to the KG
WalesWales_national_football_team beat **San Marino**San_Marino_national_footbal_team 4-0 in under-21 match. **Barry**Barry,_Vale_of_Glamorga, **Wales**Wales 1996 08 30 **Wales**Wales_national_under-21_football_team beat **San Marino**San_Marino_national_footbal_team 4-0 (halftime 2-0) in a **European**Europe under-21 soccer match on friday. Scorers: **Wales**Wales_national_under-21_football_team - **John Hartson**John_Hartson (12th, 56th and 83rd minutes), **Scott Young**Scott_Young_(Welsh_footballer) (24th) attendance: 1,800

Fig. 1. **The entity linking task**. EL model takes a raw textual input and enriches it with entity mention links in a KG. Commonly the task is split into entity recognition and entity disambiguation sub-tasks.

1.2. Previous Surveys

One of the first surveys on EL is prepared by Ling et al. [80], in 2015. They aim at providing (1) a standard problem definition to reduce a confusion that appears due to existence of variant similar tasks related to EL (e.g., Wikification [92] and named entity linking [57]), and (2) a clear comparison of models and their various aspects. In the same year, Shen et al. [133] published a survey covering the main approaches to entity linking, its applications, evaluation methods, and future directions.

There are also other surveys, which address a wider scope. The work of Martínez-Rodríguez et al. [88], 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 for information extraction side. In a similar vein, Al-Moslmi et al. [2], released in 2020, overview the research in named entity recognition and named entity disambiguation/linking published between 2014-2019.

Another recent survey paper by Oliveira et al. [105], 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.

Previous surveys on similar topics (a) do not cover many recent publications [80, 133], (b) broadly cover numerous topics [2, 88], or (c) are focused on the specific types of methods [105]. 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 pretrained 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.

1.3. Contributions

More specifically, this article makes the following contributions:

- a survey of state-of-the-art neural entity linking models;
- feature tables for neural EL methods;
- a description 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 task of EL in Section 2. In Section 3.1, the common 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 evaluation results for EL and entity representation models. Section 5 is dedicated to the application of EL by highlighting recently emerged applications for improving neural language models. Finally, Section 6 summarizes the survey and suggests a prominent direction of future work in neural entity linking.

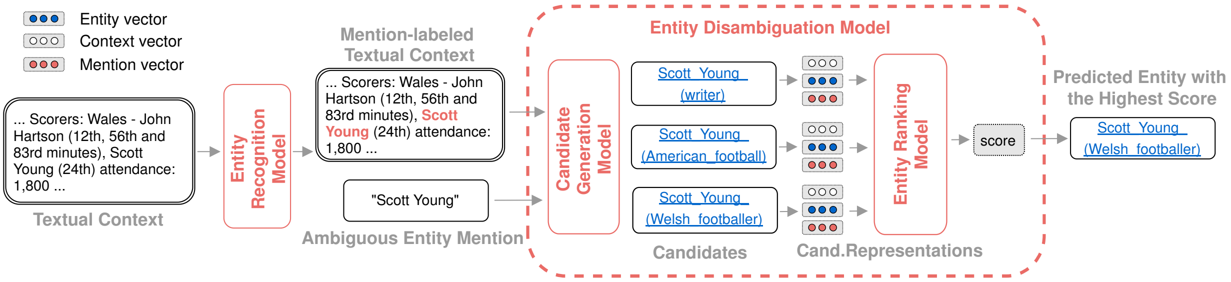
●●● Entity vector
○○○ Context vector
●●● Mention vector
... Scorers: Wales - John Hartson (12th, 56th and 83rd minutes), Scott Young (24th) attendance: 1,800 ...
Textual Context
Entity Recognition Model
Mention-labeled Textual Context
... Scorers: Wales - John Hartson (12th, 56th and 83rd minutes), Scott Young (24th) attendance: 1,800 ...
"Scott Young"
Ambiguous Entity Mention
Entity Disambiguation Model
Candidate Generation Model
Scott_Young_(writer)
Scott_Young_(American_football)
Scott_Young_(Welsh_footballer)
Candidates
Cand.Representations
Entity Ranking Model
score
Predicted Entity with the Highest Score
Scott_Young_(Welsh_footballer)

Fig. 2. **General architecture for neural entity linking.** EL contains two main steps: *Entity Recognition*, mentions in a plain text are distinguished, and *Entity Disambiguation*, a corresponding entity is predicted for the given mention. Entity Disambiguation is further divided into two steps: *Candidate Generation*, possible entities are produced for the mention, and *Entity Ranking*, a score between context/mention and a candidate is computed through the representations.

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 at least refer to an *American football player*, *Welsh football player*, or a *writer*. The EL task is to (1) correctly determine the mention in the text, e.g. determining *Wales* rather than *Wales beat* as a mention, (2) resolve its ambiguity, and ultimately provide a link to a corresponding entity entry in a KG, e.g. providing link for *Scott Young* as a *Welsh footballer*[4] instead of a *writer*[5] in this context. To achieve this goal, commonly the task is decomposed into two stages, as illustrated in Figure 1: Entity Recognition (ER) and Entity Disambiguation (ED).

2.2. *Formal Definition*

2.2.1. *Knowledge Graph (KG)*

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. [65]. Formally, as defined by Färber et al. [37], 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 $l \in L$. This RDF representation can be considered as a multi-relational graph[6] $G = (E, \mathbb{A} = \{A_0, A_1, ..., A_m \subseteq (E \times E)\})$, 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

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

2.2.2. *Entity Recognition (ER)*

The goal of entity recognition is to identify an entity mention span, while entity disambiguation performs linking of found mentions to entries of a KG. We can consider the entity recognition task as determining a ER 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 $(m_1, \dots m_n)$ in this context $m_i \in M$, where M is a set of all possible text spans in the context:[7]

$$\mathsf{ER} : C \to M^n. \tag{2}$$

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 $(c_1, \dots, c_n)$ outputs an entity assignment $(e_1, \dots, e_n), e_i \in E$, where E is a set of entities in a KG:

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

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

[4] https://en.wikipedia.org/wiki/Scott_Young_(Welsh_footballer)

[5] https://en.wikipedia.org/wiki/Scott_Young_(writer)

[6] Multi-relational graphs: https://www.slideshare.net/slidarko/multirelational-graph-structures-from-algebra-to-application-3879972

[7] We adopt and extend notation presented by Ganea et al. [45].

pairs. The size of KGs vary; 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 training set [57]. To deal with this problem, EL models should have wide generalization capabilities.

Despite KGs are usually large, they are incomplete. Therefore, some mentions in 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 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:

$$\mathsf{ED} : (M, C)^n \rightarrow (E \cup \mathsf{NIL})^n. \tag{4}$$

2.3. Terminological Aspects

More or less the same technologies and models are sometimes called differently in the literature. Namely, Wikification [19] and entity disambiguation are considered as subtypes of EL [97]. To be comprehensive in this survey, we assume that the entity linking task encompasses both entity recognition and entity disambiguation. However, only few studies suggest models that perform ER 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 [125] (which may lead to some terminological confusions). Numerous techniques that perform ER only without disambiguation are considered in many previous surveys [48, 96, 132, 161] 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 [6, 12, 154], even in particular domains, like e.g. the scholarly domain [27]. Alas, 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). 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 [58, 65]. 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 structure of pipelines and continue with various specific modifications like joint entity recognition and linking, using global context, domain-independent 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 [68]. 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 entity recognition model identifies the mention boundaries in text, e.g. determining *Scott Young* as a mention but not *attendance* in this context. The next step is to produce a short list 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 entity matching scores. An optional step is to determine unlinkable mentions, for which KGs do 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 [94, 97] as it resolves lexical ambiguity. Yet in WSD, each sense of a word can be clearly defined by WordNet [38], while in EL, KGs do not provide such an exact mapping between mentions and entities [94, 97]. Therefore, a mention potentially can be linked to any entity in a KG, resulting in large search space, e.g. “Big Blue” referring to IBM. To address this issue, candidate generation is performed, which is effectively a preliminary filtering of the entity list.

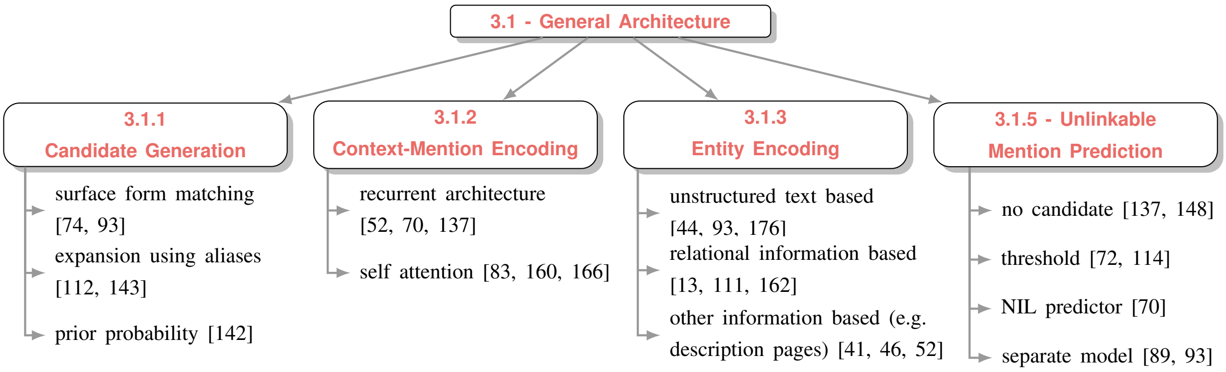
3.1 - General Architecture
3.1.1 Candidate Generation
surface form matching [74, 93]
expansion using aliases [112, 143]
prior probability [142]
3.1.2 Context-Mention Encoding
recurrent architecture [52, 70, 137]
self attention [83, 160, 166]
3.1.3 Entity Encoding
unstructured text based [44, 93, 176]
relational information based [13, 111, 162]
other information based (e.g. description pages) [41, 46, 52]
3.1.5 - Unlinkable Mention Prediction
no candidate [137, 148]
threshold [72, 114]
NIL predictor [70]
separate model [89, 93]

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

Table 1

Candidate generation examples. Five sample candidate entities for the example mention "Big Blue" for different methods. The highlighted are "correct" candidates assuming that given mention refers to the IBM corporation and not a river, e.g. Big_Blue_River_(Kansas).

Method	5 sample candidate entities for the example mention "Big Blue"
surface form matched based on DBpedia names	Big_Blue_Sky, Big_Blue_(Old_Dominion), Big_Blue_Crane_collapse, Dexter_Bexley_and_the_Big_Blue_Beastie, Big_Bluegrass
expansion using aliases on YAGO means	Big_Blue_River_(Indiana), Big_Blue_River_(Kansas), Big_Blue_(crane), Big_Red_(drink), **IBM**
probability + expansion using aliases on [44]:Anchor prob. + CrossWikis + YAGO	**IBM**, Big_Blue_River_(Kansas), The_Big_Blue Big_Blue_River_(Indiana), Big_Blue_(crane)

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

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

Candidates can be generated in several ways, as also discussed in the previous surveys [2, 133]. There are three common candidate generation methods in neural EL: (1) based on surface form matching, (2) based on expansion using aliases, and (3) based on a prior probability computation. In the first approach, a candidate list is composed of entities that match various surface forms of mentions in the text [74, 93, 176]. There are many heuristics for 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 the all Wikipedia articles present in DBpedia[8].

In the second approach, a dictionary of additional aliases is constructed using KG metadata like disambiguation/redirect pages of Wikipedia [36, 176] 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 substrings cannot possibly catch such cases. Pershina et al. [112] expand the given mention to the longest mention in a context found with coreference resolution. The entities are then selected as a candidate if an entity title matches with the longer version of mention or it exists in disambiguation/redirect pages of this mention. This resource is used in many EL models [16, 89, 101, 106, 117, 130, 137, 162]. Another well-known alternative is the YAGO ontology [143] – automatically constructed from Wikipedia and WordNet. Among many other relations, it provides *'means'* relations between mentions and entities, and this mapping is utilized as a candidate generator [44, 57, 130, 137, 162]. In this

[8] http://downloads.dbpedia.org/2016-10/core-i18n/en/labels_en.ttl.bz2

technique, the external information would help to disambiguate “Big Blue” as “IBM”. In Table 1, sample candidate entity sets of the YAGO-means based candidate mapping dataset[9] used in Hoffart et al. [57] are shown.

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 anchor text of the hyperlink determines a mention. Another widely-used option is CrossWikis [142], which is an extensive resource that leverage the frequency of mention-entity links in web crawl data [44, 52].

It is common to apply multiple approaches to candidate generation at once. For example, the resource constructed by Ganea and Hofmann [44] and used in many other EL methods [70, 75, 114, 131, 166] relies on prior probabilities obtained from entity hyperlink count statistics from CrossWikis [142] and Wikipedia, as well as on entity aliases obtained from the “means” relationship of the YAGO ontology Hoffart et al. [57]. The example mention string “Big Blue” can be labeled as its referent entity “IBM” with this methodology, as shown in Table 1.

Recent zero-shot models [46, 83, 160] perform candidate generation without external knowledge. 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 $\boldsymbol{y}_m$ using an encoder network.

$$\mathsf{mENC} : (C, M)^n \rightarrow (\boldsymbol{y}_{m_1}, \boldsymbol{y}_{m_2}, ..., \boldsymbol{y}_{m_n}) \qquad (6)$$

Several early techniques in neural EL utilize a convolutional encoder [41, 103, 141, 144], as well as attention between candidate entity embeddings and embeddings of words surrounding a mention [44, 75]. However, in recent models, two approaches prevail: recurrent networks and self-attention [152].

A recurrent architecture with LSTM cells [56] that has been a backbone model for many NLP applications, is adopted to EL in [36, 52, 70, 74, 89, 137] inter alia. Gupta et al. [52] 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. [137] 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 [22] – is used by Eshel et al. [33] in conjunction with an attention mechanism [7] to encode left and right context of a mention. Kolitsas et al. [70] represent an entity mention as a combination of LSTM hidden states included in the mention span. Le and Titov [74] simply run a bidirectional LSTM network on words complemented with embeddings of word positions relative to a target mention. Shahbazi et al. [131] adopt pre-trained ELMo [113] for mention encoding by averaging mention word vectors.

Encoding methods based on self-attention have recently become ubiquitous. The EL models presented in [83, 114, 160, 166] rely on the outputs from pre-trained BERT layers [29] for context and mention encoding. In Peters et al. [114], a mention 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. [160] reduce a sequence by keeping the representation of the special pooling symbol ‘[CLS]’ inserted at the beginning of a sequence. Logeswaran et al. [83] mark positions of a mention span by summing embeddings of words within the span with a special vector and use the same reduction strategy as Wu et al. [160]. Yamada et al. [166] concatenate text with all mentions in it and jointly encode this sequence via a self-attention model based on pretrained BERT.

3.1.3. Entity Encoding

To make EL systems robust, it is essential to construct distributed vector representations of entity candidates $\boldsymbol{y}_e$ in such a way that they capture semantic relatedness between entities in various aspects.

$$\mathsf{eENC} : E^k \rightarrow (\boldsymbol{y}_{e_1}, \boldsymbol{y}_{e_2}, ..., \boldsymbol{y}_{e_k}) \qquad (7)$$

For instance, in Figure 4, the most similar entities for *Scott Young* in the Scott_Young_(American_football)

[9] http://resources.mpi-inf.mpg.de/yago-naga/aida/download/aida_means.tsv.bz2

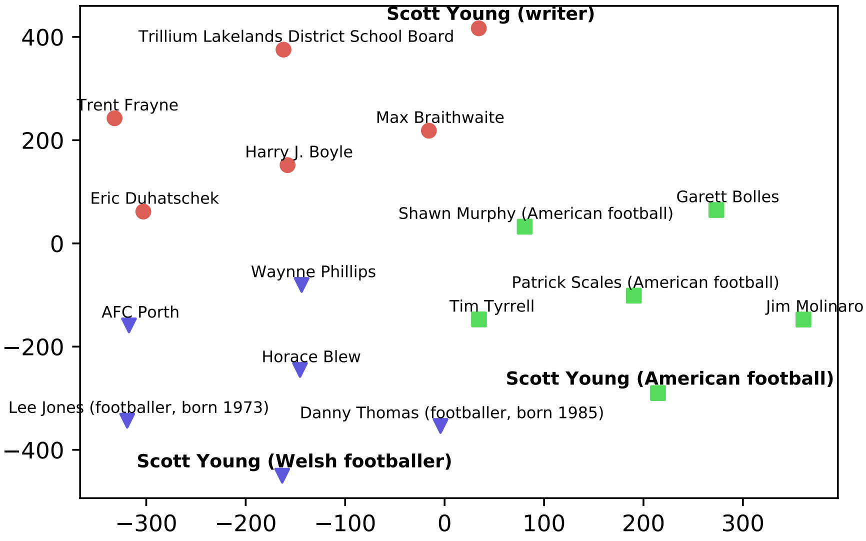

Fig. 4. **Visualization of entity embeddings.** Entity embedding space for entities related to the ambiguous entity mention "Scott Young". Three candidate entities from Wikipedia are illustrated. For each entity, their most similar 5 entities are shown in the same colors. Entity embeddings are visualized with t-SNE, which is utilized to reduce dimensionality (in this sample, to 2D), using pre-trained embeddings provided by Yamada et al. [164].

sense are related to American football, whereas Scott_Young_(writer) sense is in the proximity of writers related entities[10].

There are three common approaches to entity encoding in EL: (1) entity representations learned using unstructured texts and algorithms like word2vec [91] 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 [44] collect entity-word co-occurrences statistics from two sources: entity description pages from Wikipedia; text 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 algorithm, 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 [93, 148, 163, 176]. In [44, 93, 101, 148, 163], entity embeddings are trained in a such a way that entities become embedded in the same semantic space as words. For example, Newman-Griffis et al. [101] 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 and their surface forms (Wikipedia page titles and redirects or terminology from UMLS [11]).

In the second category of entity encoding methods that use relations between entities in a KG, Huang et al. [59] 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 [16, 35, 117, 135, 162, 164], just like the methods from the first category.

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

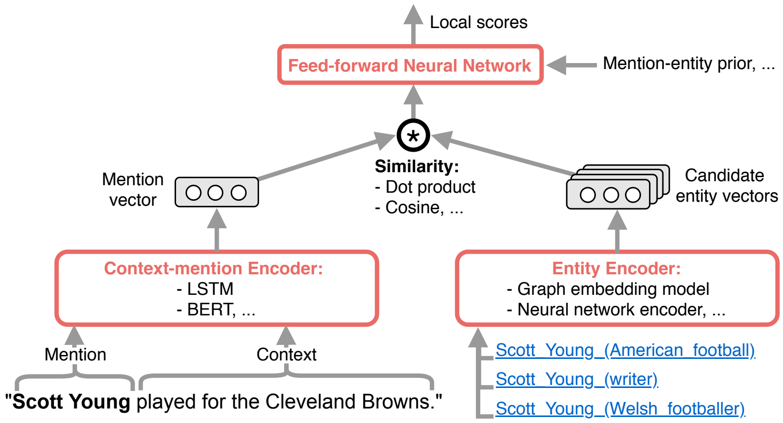

Fig. 5. **Entity ranking**. A generalized entity candidate ranking neural architecture: entity candidates are ranked according their appropriateness for a particular mention in the current context.

Recently, knowledge graph embedding has become a prominent technique and facilitated solving various NLP and data mining tasks [157] from KG completion [13, 98, 159] to entity classification [104]. For entity linking, two major graph embedding algorithms are widely used: DeepWalk [111] and TransE [13].

The goal of the DeepWalk [111] algorithm is to produce embeddings of vertices that preserve their proximity in a graph [49]. It first generates several random walks for each vertex in a graph. The generated walks are used as a training data for the skip-gram algorithm. Like in word2vec for language modelling, given a vertex, the algorithm maximizes probabilities of its neighbors in generated walks. Parravicini et al. [110], Sevgili et al. [129] leverage DeepWalk-based graph embeddings built on DBPedia [6] for entity linking. Parravicini et al. [110] use entity embeddings to compute cosine similarity scores of candidate entities in global entity linking. Sevgili et al. [129] 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 [13] algorithm is to construct embeddings of both vertices and relations in such a way that they are compatible with the facts in a KG [157]. Consider the facts in a KG 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 on Wikidata [154] and Freebase [12] have been used for entity representation in language modeling [174] and in several works on EL [9, 100, 141]. Banerjee et al. [9], Sorokin and Gurevych [141] 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 [141] find that without knowledge graph structure information, their entity linker experiences a big performance drop. Nedelchev et al. [100] integrate knowledge graph embeddings built on 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 embeddings-based method helps to pick correct entity candidates.

There are many other techniques for KG embedding: [28, 50, 104, 147, 159, 167] inter alia and very recent 5*E [99], 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 [15, 49, 127, 157], which we consider as a prominent research direction for future EL approaches.

In the last category, we place methods that produce entity representations using other types of information like entity descriptions and entity types. Often, entity encoder is a full-fledged neural network, which is a part of an entity linking architecture. Sun et al. [144] 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. [41] and Nguyen et al. [103] 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. [52] also include an encoder for fine-grained entity types. Gillick et al. [46] construct entity representations by encoding entity page titles, short entity descriptions, and entity category information with feed-forward networks. Le and Titov [74] use only entity type information from a KG and a simple feed-forward network for entity encoding.

Recent works leverage deep language models like BERT [29] or ELMo [113] for encoding entities. Logeswaran et al. [83] and Wu et al. [160] use BERT to create representations of entities from Wikipedia entity description pages. Yamada et al. [166] propose a masked entity prediction task, where a model based on the BERT architecture learns to predict randomly masked input entities. The proposed task makes the model to learn how to generate entity representations along with standard word representations. Shahbazi et al. [131] introduce E-ELMo that extends the ELMo model [113] with an additional objective. The model is trained in a multi-task fashion: to predict next/previous word, 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.

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.

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

The mention representation $\boldsymbol{y}_m$ generated in the mention encoding step is compared with candidate entity representations $\boldsymbol{y}_{e_i} (i = 1, 2, \ldots, k)$ 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. [166] 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 [44, 52, 70, 114, 160]:

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

or cosine similarity as in [41, 46, 144]:

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

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 [41, 44, 70] or other features $f(e_i, m)$ such as various similarities, a string matching indicator, and entity types [41, 130, 131, 137, 168]. One of the common techniques for that is to use an additional one or two-layer feedforward network $\phi(\cdot, \cdot)$ [41, 44, 131]. 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).

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

$$\Phi(e_i, m) = \phi(P(e_i|m), f(e_i, m)) \tag{12}$$

There are several approaches to frame 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 [83, 160]. However, instead of classes, negative candidates are used:

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

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 [44, 70, 114]:

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

$$\ell(e_i, m) = [\gamma - \Phi(e_*, m) + \Phi(e_i, m)]_+ \tag{15}$$

or

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

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.[11] Therefore, an EL system should be able to predict the absence of a reference if a mention appears in specific contexts, which is known as NIL prediction task.

$$\mathsf{NILp} : (C, M)^n \to \{0, 1\}^n \tag{17}$$

The NIL prediction task is essentially a classification with a reject option [42, 54, 55]. 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 [137, 148]. One can set a threshold for the best linking probability (or a score), below which a mention is considered unlinkable [72, 114]. Some models introduce an additional special 'NIL' entity in the ranking phase, so models can predict it as the best match for the mention [70]. 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.), and makes the final decision about whether a mention is linkable or not [89, 93].

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 Recognition and Disambiguation

While it is common to separate the entity recognition (cf. Equation 2) and entity disambiguation stages (cf. Equation 3) as illustrated in Figure 1, a few systems provide a *joint* solution for entity linking where entity recognition 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:

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

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 [85, 102], we focus on purely neural approaches proposed recently [14, 26, 70, 89, 114, 141].

The main difference of joint models is the necessity to produce also mention candidates. For this purpose, Kolitsas et al. [70] and Peters et al. [114] 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 entity recognition. In a similar fashion, Sorokin and Gurevych [141] treat each token n-gram up to a certain length as a possible mention candidate. Sorokin and Gurevych [141] 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

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

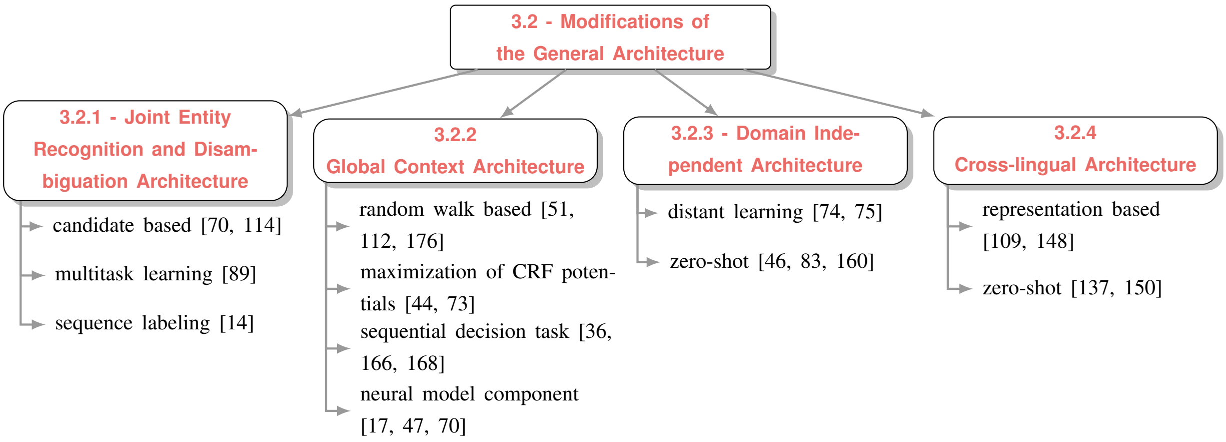
3.2 - Modifications of the General Architecture
3.2.1 - Joint Entity Recognition and Disambiguation Architecture
candidate based [70, 114]
multitask learning [89]
sequence labeling [14]
3.2.2 Global Context Architecture
random walk based [51, 112, 176]
maximization of CRF potentials [44, 73]
sequential decision task [36, 166, 168]
neural model component [17, 47, 70]
3.2.3 - Domain Independent Architecture
distant learning [74, 75]
zero-shot [46, 83, 160]
3.2.4 Cross-lingual Architecture
representation based [109, 148]
zero-shot [137, 150]

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

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 [153] that generates index numbers of potential entities in the input sequence.

Martins et al. [89] describe the approach with a tighter integration between recognition 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 [14] goes further by suggesting a completely end-to-end method that deals with entity recognition and linking jointly without explicitly executing a candidate generation step. They formulate the task 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 [70] but outperforms the baseline, in which candidate generation, entity recognition, and linking are performed independently.

De Cao et al. [26] recently have proposed a generative approach to performing entity recognition and disambiguation jointly. Their model, which is based on BART [79], performs a sequence-to-sequence autoregressive generation of text markup that contains 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 beginning 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 state-of-the-art results for joint ER 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 [160]. The authors also note that the memory footprint of the proposed model is much smaller than that of models based on the standard architecture.

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:

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

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

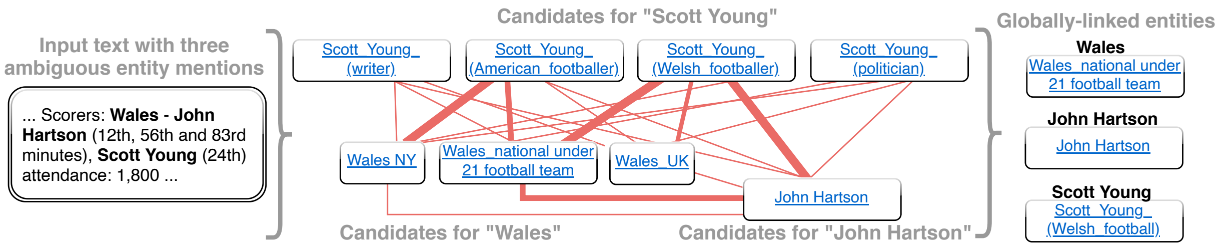
Input text with three ambiguous entity mentions
... Scorers: Wales - John Hartson (12th, 56th and 83rd minutes), Scott Young (24th) attendance: 1,800 ...
Candidates for "Scott Young"
Scott_Young_ (writer)
Scott_Young_ (American_footballer)
Scott_Young_ (Welsh_footballer)
Scott_Young_ (politician)
Wales NY
Wales_national under 21 football team
Wales_UK
John Hartson
Candidates for "Wales"
Candidates for "John Hartson"
Globally-linked entities
Wales
Wales_national under 21 football team
John Hartson
John Hartson
Scott Young
Scott_Young_ (Welsh_football)

Fig. 7. **Global entity disambiguation**. The global entity linking resolves all mentions simultaneously based on entity coherence. Bolder lines indicate higher degrees of entity-entity similarity.

for other entities in a context as illustrated in Figure 7 and Equation 20.

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

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 modelling consistency between entities and the extra information of the global context improves the disambiguation accuracy, the number of possible entity assignments is combinatorial, which results in a high time complexity of disambiguation [44, 168]. 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 [162].

The typical approach to global disambiguation is to generate a graph containing candidate entities of mentions in a context and perform random walk algorithms, e.g. PageRank [108], over it to select highly consistent entities [51, 112, 176]. In this category, Globerson et al. [47] introduce a model with an attention mechanism that takes into account only the subgraph of the target mention, instead of all the mention candidates in a document.

Some works approach global ED by maximizing the Conditional Random Field (CRF) potentials, where the first component Φ represents a local entity-mention score, and the other component Ψ measures coherence among selected candidates [44, 45, 73, 75]:

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

However, model training and its exact inference are NP-hard. Ganea and Hofmann [44] adapt loopy belief propagation [45, 47] with message passing iterations using pairwise entity scores to reduce the complexity. Le and Titov [73] expand it by modelling coreference relations of mentions as latent variables (the mentions are coreferent if they refer to the same entity). Shahbazi et al. [130] develop a greedy beam search strategy, which starts from a locally optimal initial solution and that is improved by searching possible corrections with the focus on the least confident mentions.

Despite the optimizations proposed in the 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 [36]. For example, if two mentions have coherent erroneous candidates, this noisy information can mislead the final global scoring. To resolve this issue, some studies define the 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. [36] 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. [168] also use reinforcement learning to determine ordering for mention disambiguation. They also 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. Yamada et al. [166] iteratively predict entities for yet unresolved mentions with a BERT model, while attending on the previous most confident entity choices. Yamada et al. [162] and Radhakrishnan et al. [117] measure the similarity first based on unambiguous mentions and then predict entities for complex cases.

Many studies rely on the idea of attaching an entity coherence component to the local scoring model and train their parameters jointly. In this case, local models can directly benefit from the pairwise coherence score without a necessity of handling the optimization of the global objective. The coherence component of Kolitsas et al. [70] is an additional feed-forward 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. [35] use the similarity score between the target entity and its surrounding entity candidates in a specified window as feature for the disambiguation model. In the same vein, Yamada et al. [162] and Radhakrishnan et al. [117] treat the global coherence as a feature for the final disambiguation model. Instead of computing entity coherence scores, Tsai and Roth [148] directly use embeddings of previously linked entities as features for the disambiguation model. Distinctively, Cao et al. [17] integrate a graph convolutional network into a disambiguation model that takes advantage of the knowledge provided by a subgraph of candidate entities in a documents. Nguyen et al. [103] use an RNN to 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 computation of the global score.

Another approach that can be considered as global is to make use of a larger context to capture the coherence implicitly instead of explicitly designing an entity coherence component [16, 41, 52, 72, 93, 114, 137].

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 labor. Earlier, this problem is tackled by few domain-independent approaches based on unsupervised [16, 101, 156] and semi-supervised models [72]. Recent studies provide solutions based on distant learning and zero-shot methods.

Le and Titov [74, 75] 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 [74] or mention-entity priors computed from entity hyperlink statistics extracted from Wikipedia [75].

Recently proposed zero-shot techniques [83, 160] 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. [52] (not purely zero-shot because they 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 prebuilt 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 as caching), compute a representation of a mention and calculate its similarity with all the description representations. Precomputed representations of descriptions save a lot of time at the inference stage. Particularly, Logeswaran et al. [83] use the BM25 information retrieval formula [66], 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. [46], 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 comparison of representations. Due to 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, 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. [160] suggest using a BERT-based bi-encoder for candidate generation. Two separate encoders generate representations of mentions and entity descriptions. Similarly to the previous work, the candidate selection is based on the score obtained via a dot-product of the representations.

Zero-shot approaches use descriptions for entity ranking as well. Surprisingly, a very simple embedding-based approach [46] described above shows very competitive scores on the TAC KBP-2010 benchmark, outperforming some complex neural architectures. The recent studies of Logeswaran et al. [83] and Wu et al. [160] 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 [60, 123]. In both studies, cross-encoders achieve superior results compared to bi-encoders and count-based approaches.

Evaluation of zero-shot systems requires data from different domains. Logeswaran et al. [83] proposes the *Zero-shot EL*[12] dataset, constructed from several Wikias[13]. In the proposed setting, training is performed on one set of Wikias while evaluation is performed on others. Gillick et al. [46] 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. [83] further unsupervised pre-training on source data, as well as on the target data is beneficial. Development of better approaches to 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

Abundance of labeled data for EL in English language contrasts with amount of data available in other languages. At the same time, such a unique source of supervision as Wikipedia is available for a variety of languages. However, there is still a big gap between resource-rich Wikipedia languages and low-resource ones.

The cross-lingual EL methods [64] aim at overcoming the lack of annotation for some languages by leveraging supervision coming from their high-resource counterparts. The inter-language links in Wikipedia is one of the most widely used sources of cross-lingual supervision. These links map pages to equivalent pages in another language.

Challenges in cross-lingual EL start at candidate generation and entity recognition steps, since the low-resource language can lack mappings between mention strings and entities. In addition to the standard methods with mention-entity priors [137, 148, 150], candidate generation can be approached by mining a translation dictionary [109], training a translation and alignment model [149], or applying a neural character-level string matching model [124]. The latter relies on training on a high-resource pivot language, similar to the target low-resource one. The neural string matching approach can be further improved with simpler average n-gram encoding and extending entity-entity pairs with mention-entity examples [175]. For entity recognition, the transfer of BiLSTM-CRF with a character encoding network from a similar high-resource pivot language can be applied [24].

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. [109] uses the Abstract Meaning Representation (AMR) [8] statistics in English Wikipedia and mention context for ranking. To train an AMR tagger, pseudo-labeling [76] is used. Tsai and Roth [148] 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 cross-lingual setting by more than 1% for Spanish and Chinese. Sil et al. [137] propose a method for zero-shot transfer from a high-resource language. The

[12] https://github.com/lajanugen/zeshel

[13] https://www.wikia.com

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. [150] argues that the success of zero-shot cross-lingual approaches [137, 148] might be largely originating from a better estimation of mention-entity prior probabilities. Their approach extends [137] with global context information and incorporation 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 high-resource language might be very beneficial for low-resource settings.

Existing 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 recent success of zero-shot multilingual transfer of large pre-trained language models, this might be a motivation to utilize powerful multilingual self-supervised models.

3.3. Summary

We summarize design features for neural EL models in Table 2. The mention encoders have made a shift to self-attention architectures and start 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 recognition 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. It can be a simplistic embedding averaging method or a feature-engineering approach.
- CNN – an encoder based on convolutional layers (usually with pooling).
- Tensor net. – an encoder that uses a tensor network.
- Atten. – means that an encoder uses an attention mechanism.
- GRU – an encoder based on a recurrent neural network and gated recurrent units [22].
- LSTM – an encoder based on a recurrent neural network and long short-term memory cells [56] (might be also bidirectional).
- FFNN – an encoder based on a simple feedforward neural network.
- ELMo – an encoder based on a pre-trained ELMo model [113].
- BERT – an encoder based on a pre-trained BERT model [29].

Note here that, theoretical complexity of various types of encoders is different. As discussed by Vaswani et al. [152], 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 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 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 [152] and specialized literature on this topic, e.g. [82, 107, 138].

The **global** column shows whether a system uses a global solution (see Section 3.2.2). The **recognition** column refers to joint entity recognition and disambiguation models, where recognition 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.

Table 2

Features of neural EL models. Neural entity linking models compared according to their architectural features. (The footnotes in the table are explained in the text.)

Models	**Encoder Type**	**Global**	**Recog-nition**	**NIL Pred.**	**Ent. Encoder Source based on**	**Candidate Generation**	**Learning Type for Disam.**	**Cross-lingual**
Sun et al. (2015) [144]	CNN+Tensor net.				ent. specific info.	surface match+ aliases	supervised	
Francis-Landau et al. (2016) [41]	CNN	✘[4]			ent. specific info.	surface match+prior	supervised	
Fang et al. (2016) [35]	n/a	✘			relational info.	prior[1]	supervised	
Yamada et al. (2016) [162]	n/a	✘			relational info.	aliases	supervised	
Zwicklbauer et al. (2016) [176]	n/a	✘		✘	unstructured text	surface match+prior +nearest neighbors	unsupervised[7]	
Tsai and Roth (2016) [148]	n/a	✘		✘	unstructured text	prior	supervised	✘
Nguyen et al. (2016) [103]	CNN	✘		✘	ent. specific info.	surface match+prior	supervised	
Cao et al. (2017) [16]	n/a	✘			relational info.	aliases	supervised or unsupervised	
Eshel et al. (2017) [33]	GRU+Atten.				unstructured text[2]	aliases	supervised	
Ganea and Hofmann (2017) [44]	Atten.	✘			unstructured text	prior+aliases	supervised	
Moreno et al. (2017) [93]	n/a	✘[4]		✘	unstructured text	surface match+ aliases	supervised	
Gupta et al. (2017) [52]	LSTM	✘[4]			ent. specific info.	prior	supervised[6]	
Sorokin and Gurevych (2018) [141]	CNN	✘	✘		relational info.	surface match	supervised	
Shahbazi et al. (2018) [130]	Atten.	✘			unstructured text	prior or aliases	supervised	
Le and Titov (2018) [73]	Atten.	✘			unstructured text	prior	supervised	
Newman-Griffis et al. (2018) [101]	n/a				unstructured text	aliases	unsupervised	
Radhakrishnan et al. (2018) [117]	n/a	✘			relational info.	aliases	supervised	
Kolitsas et al. (2018) [70]	LSTM	✘	✘		unstructured text	prior+aliases	supervised	
Sil et al. (2018) [137]	LSTM+Tensor net.	✘[4]		✘	ent. specific info.	prior or aliases	supervised[5]	✘
Upadhyay et al. (2018) [150]	CNN	✘			ent. specific info.	prior	supervised[5]	✘
Cao et al. (2018) [17]	FFNN	✘[4]			relational info.	prior+aliases	supervised	
Raiman and Raiman (2018) [119]	n/a	✘			n/a	prior+type classifier	supervised	✘
Mueller and Durrett (2018) [95]	GRU+Atten.+CNN				unstructured text[2]	aliases	supervised	
Shahbazi et al. (2019) [131]	ELMo				unstructured text	prior+aliases or aliases	supervised	
Logeswaran et al. (2019) [83]	BERT				ent. specific info.	BM25	zero-shot	
Gillick et al. (2019) [46]	FFNN				ent. specific info.	nearest neighbors	supervised[6]	
Peters et al. (2019) [114][3]	BERT	✘[4]	✘	✘	unstructured text	prior + aliases	supervised	
Le and Titov (2019) [74]	LSTM				ent. specific info.	surface match	weakly-supervised	
Le and Titov (2019) [75]	Atten.	✘			unstructured text	prior+aliases	weakly-supervised	
Fang et al. (2019) [36]	LSTM	✘			unstructured text	prior+aliases	supervised	
Martins et al. (2019) [89]	LSTM		✘	✘	unstructured text	aliases	supervised	
Yang et al. (2019) [168]	Atten. or CNN	✘			unstructured text	prior	supervised	
Broscheit (2019) [14]	BERT		✘		n/a	n/a	supervised	
Onoe and Durrett (2020) [106]	ELMo+Atten.+CNN				n/a	prior or aliases	supervised	
Wu et al. (2020) [160]	BERT				ent. specific info.	nearest neighbors	zero-shot	
Yamada et al. (2020) [166]	BERT	✘			unstructured text	prior+aliases	supervised	
Banerjee et al. (2020) [9]	n/a		✘		relational info.	surface match	supervised	
De Cao et al. (2021) [26]	BART	✘	✘		n/a	prior	supervised	

Table 3

Evaluation datasets. Descriptive statistics of the evaluation datasets used in this survey to compare the models.

Corpus	Text Type	# of Docs	# of Mentions
AIDA-B [57]	News	231	4485
MSNBC [25]	News	20	656
AQUAINT [92]	News	50	727
ACE2004 [121]	News	36	257
CWEB [43, 51]	ClueWeb & Wikipedia	320	11154
WW [43, 51]	ClueWeb & Wikipedia	320	6821
TAC KBP 2010 [63]	News & Web	1013	1020[1]
TAC KBP 2015 Chinese [64]	News & Forums	166	11066
TAC KBP 2015 Spanish [64]	News & Forums	167	5822

[1] # of mention/entity pairs

- unstructured text means that the entity representations constructed based on unstructured text and approaches based on co-occurrence statistics developed originally for word embeddings like word2vec [91];
- relational info. denotes that the model uses relations between entities in KGs;
- ent. specific info. denotes that the encoder uses other types of entity information, like entity descriptions, types, or categories.

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

- n/a – the solution presented by Broscheit [14] does not have an explicit candidate generation step;
- surface match – surface match 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 [119] filter candidates using a classifier for an automatically learned type system;
- BM25 – Logeswaran et al. [83] 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. [160] train a supplementary model for this purpose.

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).

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

1. In classification, the prior is checked by a threshold. This can be considered as a candidate selection step.
2. These works use only entity description pages, however, they are labeled as in first category (unstructured text) since their training is based on word2vec.
3. The authors provide EL as a subsystem of language modeling.
4. 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.
5. These works are zero-shot in terms of model adaptation to a new language using English annotated data, while the other zero-shot works solve the problem of model adaptation to a new domain without switching the language.
6. These studies are domain-independent as discussed in Section 3.2.3.
7. Zwicklbauer et al. [176] may not be accepted as purely unsupervised since they have some parameters in the disambiguation algorithm.

4. Evaluation

In this section, we present evaluation of the models on the entity linking and entity relatedness tasks over 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 an access to the mention boundaries. The second setup is entity recognition and disambiguation (ER+ED) where the inputs for the systems that perform ER and ED jointly are plain text. We stated their results in separate tables since the scores for the joint models accumulate the errors made during the in entity recognition phase.

Datasets For the purposes of evaluation of the models we used widely-used datasets for evaluation of EL: AIDA [57], TAC KBP 2010 [63], MSNBC [25], AQUAINT [92], ACE2004 [121], CWEB [43, 51], and WW [43, 51]. Among them, CWEB and WW are large datasets that are annotated automatically, while AIDA is also a large dataset, but annotated manually [44]. 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 [64] Spanish (es) and Chinese (zh) datasets. The descriptive statistics of the datasets and their data sources are presented in Table 3 according information reported in [32, 44, 46, 64].

Evaluation Metrics For the ED setting, we report micro F1 or accuracy scores achieved by the systems. 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 [133], and so F1 score equals precision, recall and accuracy score in disambiguation models [133]:

$$F1 = Acc = \frac{\# \, of \ correctly \ disamb. \ mentions}{\# \, of \ total \ mentions} \tag{22}$$

For the ER+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. [133] and Ganea et al. [45]:

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

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

$$F1 = \frac{2 \cdot P \cdot R}{P + R} \tag{25}$$

GERBIL [126] is a benchmarking platform used by multiple papers described in this survey. It implements various experimental settings, e.g. entity disambiguation (ED), denoted as D2KB, combination of entity recognition and disambiguation (ER+ED) denoted as A2KB among other setups. It provides the required evaluation metrics, i.e. micro-macro precision, recall, and F-measure. Besides, it stores the evaluation datasets in the standartized way along with the annotations.

Baseline Models While our goal is to perform a survey of neural entity linking 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) [90], AIDA (2011) [57], Ratinov et al. (2011) [121], WAT (2014) [115], Babelfy (2014) [94], Lazic et al. (2015) [72], Chisholm and Hachey (2015) [20], and PBOH (2016) [45].

For each system, we present the best scores reported by the authors and, for baseline systems, the results are presented as reported in Kolitsas et al. [70] and Ganea and Hofmann [44].

4.1.2. Discussion of Results

Entity Disambiguation Results We start our presentation of results from the disambiguation only models (for which entity boundaries are already provided). Figure 8 shows how performance of the entity disambiguation models improved during the course of the last decade and how the best classic models correspond to the recent neural state-of-the-art models for entity linking. As one may observe the models based on deep learning substantially improve the performance pushing the state of the art by around 10 points. AIDA is the most widely used dataset (but also one of the largest), but we also report results on other datasets in Table 4.

Table 4

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

	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) [90]	0.561	-	0.421	0.518	0.539	-	-	-	-
AIDA (2011) [57]	0.770	-	0.746	0.571	0.798	-	-	-	-
Ratinov et al. (2011) [121]	-	-	0.750	0.830	0.820	0.562	0.672	-	-
WAT (2014) [115]	0.805	-	0.788	0.754	0.796	-	-	-	-
Babelfy (2014) [94]	0.758	-	0.762	0.704	0.619	-	-	-	-
Lazic et al. (2015) [72]	0.864	-	-	-	-	-	-	-	-
Chisholm and Hachey (2015) [20]	0.887	-	-	-	-	-	-	-	-
PBOH (2016) [45]	0.804	-	0.861	0.841	0.832	-	-	-	-
Neural Models									
Sun et al. (2015) [144]	-	0.839	-	-	-	-	-	-	-
Francis-Landau et al. (2016) [41]	0.855	-	-	-	0.899	-	-	-	-
Fang et al. (2016) [35]	-	0.889	0.755	0.852	0.808	-	-	-	-
Yamada et al. (2016) [162]	0.931	0.855	-	-	-	-	-	-	-
Zwicklbauer et al. (2016) [176]	0.784	-	0.911	0.842	0.907	-	-	-	-
Tsai and Roth (2016) [148]	-	-	-	-	-	-	-	0.824	0.851
Nguyen et al. (2016) [103]	0.872	-	-	-	0.897	-	-	-	-
Cao et al. (2017) [16]	0.851	-	-	-	-	-	-	-	-
Eshel et al. (2017) [33]	0.873	-	-	-	-	-	-	-	-
Ganea and Hofmann (2017) [44]	0.922	-	0.937	0.885	0.885	0.779	0.775	-	-
Gupta et al. (2017) [52]	0.829	-	-	-	0.907	-	-	-	-
Shahbazi et al. (2018) [130]	0.944	0.879	-	-	-	-	-	-	-
Le and Titov (2018) [73]	0.931	-	0.939	0.884	0.899	0.775	0.780	-	-
Radhakrishnan et al. (2018) [117]	0.930	0.896	-	-	-	-	-	-	-
Kolitsas et al. (2018) [70]	0.831	-	0.864	0.832	0.855	-	-	-	-
Sil et al. (2018) [137]	0.940	0.874	-	-	-	-	-	0.823	0.844
Upadhyay et al. (2018) [150]	-	-	-	-	-	-	-	0.844	0.860
Cao et al. (2018) [17]	0.800	0.910	-	0.870	0.880	-	0.860	-	-
Raiman and Raiman (2018) [119]	0.949	0.909	-	-	-	-	-	-	-
Shahbazi et al. (2019) [131]	0.962	0.883	-	-	-	-	-	-	-
Gillick et al. (2019) [46]	-	0.870	-	-	-	-	-	-	-
Le and Titov (2019) [74]	0.815	-	-	-	-	-	-	-	-
Le and Titov (2019) [75]	0.897	-	0.922	0.907	0.881	0.782	0.817	-	-
Fang et al. (2019) [36]	0.943	-	0.928	0.875	0.912	0.785	0.828	-	-
Yang et al. (2019) [168]	0.946	-	0.946	0.883	0.901	0.756	0.788	-	-
Onoe and Durrett (2020) [106]	0.859	-	-	-	-	-	-	-	-
Wu et al. (2020) [160]	-	0.940	-	-	-	-	-	-	-
Yamada et al. (2020) [166]	0.950	-	0.963	0.935	0.919	0.789	0.892	-	-
De Cao et al. (2021) [26]	0.933	-	0.943	0.899	0.901	0.773	0.874	-	-

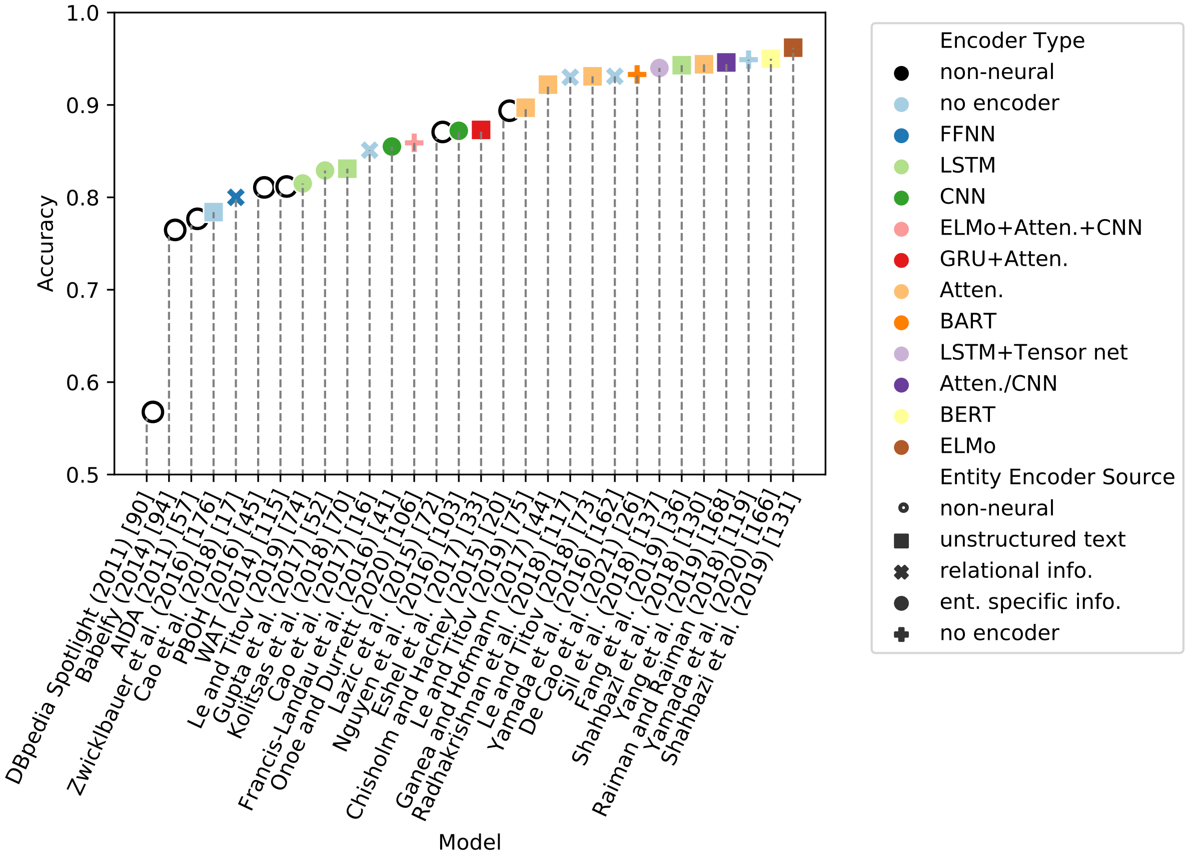

Fig. 8. **Entity disambiguation progress**. Performance of the classic entity linking models (black circle) with the more recent neural models (fill-colored other shapes) on the AIDA dataset shows an improvement (around 10 points of accuracy). The colors refer to the feature of Encoder Type in the Table 2 and the shapes denote another feature, a type of Entity Encoder Source as explained in the Table 2.

Table 5

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

	AIDA-B	MSNBC
Non-Neural Baseline Models		
DBpedia Spotlight (2011) [90]	0.578	0.406
AIDA (2011) [57]	0.728	0.651
WAT (2014) [115]	0.730	0.645
Babelfy (2014) [94]	0.485	0.397
Neural Models		
Kolitsas et al. (2018) [70]	0.824	**0.724**
Martins et al. (2019) [89]	0.819	-
Peters et al. (2019) [114]	0.744	-
Broscheit (2019) [14]	0.793	-
De Cao et al. (2021) [26]	**0.837**	**0.737**

Among local models for disambiguation, the best results are reported by Shahbazi et al. [130] and Wu et al. [160]. It is worth noting that the latter model can be used in a zero-shot setting. Shahbazi et al. [131] has the best score on AIDA-B among other models. However, this is due to the use of less-ambiguous resource of Pershina et al. [112] for candidate generation, while many of other works use the YAGO-based resource provided by Ganea and Hofmann [44], 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 context). The global model of Yamada et al. [166] produce results that are consistently better as compared to other solutions including the results of Shahbazi et al. [131] reported for the YAGO-based resource. The performance improvements are explained by the authors by the novel masked entity prediction objective that helps to fine-tune pre-trained BERT for producing contextualized entity embeddings and the multi-step global disambiguation algorithm.

Joint Entity Recognition and Disambiguation Table 5 presents results of the joint ER and ED models. Only a fraction of the models presented in above is capable of performing both entity recognition and disambiguation; thus, the list of results is much shorter. Among the joint recognition and disambiguation solutions, the top-performing system at the time of writing is De Cao et al. [26]. This system and others that solve also the ER task fall behind the disambiguation-only systems, since they rely on noisy mention boundaries produced by themselves. In the joint setting, the neural models also substantially (up to 10 points) outperform the classic models.

On Effect of Hyperparameter Search As explained above, we report the best scores reported by authors for neural models, in Table 4 and Table 5. In principle, each neural model can be further tuned as shown by Reimers and Gurevych [122], 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 case is RoBERTa [81], 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 on various tasks, while keeping the original architecture.

4.2. Entity Relatedness

In this section, evaluation of entity relatedness is discussed. This evaluation is different from any pipeline in EL and its focus is on entity relatedness only.

4.2.1. Experimental Setup

The evaluation data is provided by Ceccarelli et al. [18] using the dataset of Hoffart et al. [57]. It is in the form of queries, where the first entity is accepted as correctly linked and the second entity is the candidate [44].

Entity representation performance can be evaluated through an entity relatedness task. Namely, the task is to rank entities for the target one, which is usually performed based on a similarity of entity representations except for two studies: Milne and Witten [92] introduce a Wikipedia hyperlink-based measure, known as WLM, and recently, El Vaigh et al. [31] provide a weighted semantic relatedness measure.

The evaluation of ranking quality is performed with a normalized discounted cumulative gain (nDCG) [61] and a mean average precision (MAP) [171]. nDCG is commonly used in information retrieval and provides a fair evaluation by measuring the position impressiveness. Similarly, MAP measures how accurately the model performs for the target entity.

4.2.2. Discussion of Results

In Table 6, the entity relatedness scores are reported. The highest score is reported by Huang et al. [59] and the reason would be that they use different sources of entity information, like entity types [44]. Ganea and Hofmann [44] and Cao et al. [16] achieve good scores, and recently, Shi et al. [135] also present an excellent performance by using various data sources based on textual and KG, like types provided by a category hierarchy of a knowledge graph.

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 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, complexity of medical ontologies such as UMLS [11], and scarcity of annotated resources. There is a long history of development of EL tools for biomedical literature and electronic health record mining applications [5, 71, 84, 128, 140]. These tools have been successfully applied for summarization of clinical reports [87], extraction of drug-disease treatment relationships [69], differential diagnosis [4], patient screening [34], and many other tasks. Besides medical text processing, EL is widely used for mining social networks and news. For example, Twitci-

Table 6

Entity relatedness evaluation. Reported results for entity relatedness evaluation on the dataset of Ceccarelli et al. [18] .

	nDCG@1	nDCG@5	nDCG@10	MAP
Milne and Witten (2008) [92]	0.540	0.520	0.550	0.480
Huang et al. (2015) [59]	0.810	0.730	0.740	0.680
Yamada et al. (2016) [162]	0.590	0.560	0.590	0.520
Ganea and Hofmann (2017) [44]	0.632	0.609	0.641	0.578
Cao et al. (2017) [16]	0.613	0.613	0.654	0.582
El Vaigh et al. (2019) [31]	0.690	0.640	0.580	-
Shi et al. (2020) [135]	0.680	0.814	0.820	-

dent [1] uses the DBpedia Spotlight [90] EL system for mining Twitter messages for small scale incidents. Provatorova et al. [116] leverage a recently proposed EL toolkit REL [151] for mining historical newspapers for people, places, and other entities in the CLEF HIPE 2020 evaluation campaign [30]. Luo et al. [86] 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 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 [62] and Ellis et al. [32] overview various successful systems for knowledge graph population participated in the TAC KBP 2010 and 2015 tasks. Shen et al. [134] 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. [78] use EL to complement the results of a biomedical literature search engine with found entities: genes, diseases, drugs, etc. COVIDASK [77], a real-time question answering system that helps researchers to retrieve information related to coronavirus, uses the BioSyn model [145] 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 usability of the system. Yih et al. [169] 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. [136] develop a fast EL algorithm for pre-processing large corpora for their autonomous debating system [139] 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 of Neural Language Models

In addition to aforementioned applications, 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:

$$\mathcal{L}_{\text{JOINT}} = \mathcal{L}_{\text{BERT}} + \mathcal{L}_{\text{EL-related}} . \qquad (26)$$

Neural entity linkers can be integrated in any other networks. The main novel trend is the use of EL in-

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

formation 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 [114] 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:

$$\mathcal{L}_{\text{BERT}} = \mathcal{L}_{\text{NSP}} + \mathcal{L}_{\text{MLM}}. \quad (27)$$

$$\mathcal{L}_{\text{KnowBert}} = \mathcal{L}_{\text{NSP}} + \mathcal{L}_{\text{MLM}} + \mathcal{L}_{\text{EL}}. \quad (28)$$

The authors adopt the general end-to-end EL architecture of [70] but use only the local context for disambiguation and use 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 prebuilt 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 [174] expands the BERT [29] 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 [13]. EL in this study is performed by an external tool TAGME [39]. 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):

$$\mathcal{L}_{\text{ERNIE}} = \mathcal{L}_{\text{NSP}} + \mathcal{L}_{\text{MLM}} + \mathcal{L}_{\text{dEA}}. \quad (29)$$

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 [155].

Wang et al. [158] train a disambiguation network named KEPLER using the composition of two losses: regular MLM and a Knowledge Embedding (KE) loss based on the TransE [13] objective for encoding graph structures:

$$\mathcal{L}_{\text{KEPLER}} = \mathcal{L}_{\text{MLM}} + \mathcal{L}_{\text{KE}}. \quad (30)$$

In the KE loss, representations of entities are obtained from their textual descriptions encoded with a self-attention network [81], 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 [155], it shows increased performance on a wide range of KB-related tasks such as TACRED [173], FewRel [53], and OpenEntity [21].

Yamada et al. [165] propose a deep pre-trained model called "Language Understanding with Knowledge-based Embeddings" (LUKE). They modify RoBERTa [81] 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 to restore randomly masked entities in an entity-annotated corpus.

$$\mathcal{L}_{\text{LUKE}} = \mathcal{L}_{\text{MLM}} + \mathcal{L}_{\text{MLMe}}. \quad (31)$$

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) [21], TACRED (relation classification) [173], CoNLL-2003 (named entity recognition) [146], ReCoRD (cloze-style question answering) [172], and SQuAD 1.1 (reading comprehension) [120].

Févry et al. [40] 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 [114]. However, in addition to 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 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 [165], 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.

$$\mathcal{L}_{\text{EaE}} = \mathcal{L}_{\text{MLM}} + \mathcal{L}_{\text{NER}} + \mathcal{L}_{\text{EL}}. \tag{32}$$

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 [118] on the TriviaQA task [67]. The authors also show that EAE contains more factual knowledge than a comparably-sized BERT model.

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 a LM while keeping the 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 construction of the future foundation models.

6. Conclusion

In this survey, we have analyzed recently proposed neural entity linking models, which generally perform the task with higher accuracy than classical methods scores. 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. The various modifications of general architecture are grouped into four common directions: (1) joint entity recognition 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 make 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 joint recognition and disambiguation solutions, the leadership is owned by De Cao et al. [26]. Among published local models for disambiguation, the best result is reported by Wu et al. [160]. It is worth noting that this model can be used in a zero-shot setting. The global models outperform the local ones. The work of Yamada et al. [166] reports results that are consistently better in comparison to all other solutions. The performance improvements are attributed to the masked entity prediction mechanism for entity embedding and to the usage of the pre-trained model based on BERT with a multi-step global scoring function.

7. Future Directions

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

1. **End-to-end models including the candidate generation step**: The candidate generation step requires to collect information from a large amount of data, as described in the Section 3.1.1. Al-

though the models could create a domain-independent architecture, they are still based on data from a candidate generator. Therefore, a possible direction would be to handle the candidate generation step without the requisite of external data or directly eliminate this step. There are some studies, which use either the representations [46, 160] or BM25 scores computed from entity descriptions [83] to find out candidates. However, these models do not provide complete end-to-end solutions. Thus, future approaches could tackle the challenge of a complete end-to-end solution without a candidate generator.

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 ER and EL 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 [114, 158, 174] 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 the 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 the 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 straightforwards and may be useful to inject information about entities contained in an EL model into other, possibly specialized, architectures.

Acknowledgements

The work was partially supported by a Deutscher Akademischer Austauschdienst (DAAD) doctoral stipend and the DFG-funded JOIN-T project BI 1544/4. The work of Artem Shelmanov in the current study (preparation of sections related to application of entity linking to neural language models, entity ranking, context-mention encoding, and overall harmonization of the text and results) is supported by the Russian Science Foundation (project 20-11-20166). Finally, this work was partially supported by the joint MTS-Skoltech laboratory.

References

[1] Fabian Abel, Claudia Hauff, Geert-Jan Houben, Richard Stronkman, and Ke Tao. Twitcident: Fighting Fire with Information from Social Web Streams. In *Proceedings of the 21st International Conference on World Wide Web*, WWW '12 Companion, page 305–308, New York, NY, USA, 2012. Association for Computing Machinery. ISBN 9781450312301. . URL https://doi.org/10.1145/2187980.2188035.

[2] Tareq Al-Moslmi, Marc Gallofré Ocaña, Andreas L. Opdahl, and Csaba Veres. Named entity extraction for knowledge graphs: A literature overview. *IEEE Access*, 8:32862–32881, 2020. URL https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8999622.

[3] Rami Aly, Andreas Vlachos, and Ryan 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)*, pages 1516–1528, Online, August 2021. Association for Computational Linguistics. . URL https://aclanthology.org/2021.acl-long.120.

[4] Hadi Amiri, Mitra Mohtarami, and Isaac 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)*, pages 1012–1019, Online, August 2021. Association for Computational Linguistics. . URL https://aclanthology.org/2021.acl-short.128.

[5] Alan R Aronson and François-Michel Lang. An overview of MetaMap: historical perspective and recent advances. *Journal of the American Medical Informatics Association*, 17(3):229–236, 2010.

[6] Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. DBpedia: A nucleus for a web of open data. In *Proceedings of the 6th International The Semantic Web and 2nd Asian Conference on Asian Semantic Web Conference*, ISWC'07/ASWC'07, page 722–735, Berlin, Heidelberg, 2007. Springer-Verlag. ISBN 3540762973.

[7] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua 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. URL http://arxiv.org/abs/1409.0473.

[8] Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. Abstract meaning representation for sembanking. In *Proceedings of the 7th linguistic annotation workshop and interoperability with discourse*, pages 178–186, Sofia, Bulgaria, 2013. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W13-2322/.

[9] Debayan Banerjee, Debanjan Chaudhuri, Mohnish Dubey, and Jens Lehmann. PNEL: Pointer Network Based End-To-End Entity Linking over Knowledge Graphs. In Jeff Z. Pan, Valentina Tamma, Claudia d'Amato, Krzysztof Janowicz, Bo Fu, Axel Polleres, Oshani Seneviratne, and Lalana Kagal, editors, *The Semantic Web – ISWC 2020*, pages 21–38, Cham, 2020. Springer International Publishing. ISBN 978-3-030-62419-4.

[10] Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. A neural probabilistic language model. *J. Mach. Learn. Res.*, 3(null):1137–1155, March 2003. ISSN 1532-4435.

[11] Olivier Bodenreider. The Unified Medical Language System (UMLS): integrating biomedical terminology. *Nucleic Acids Research*, 32(suppl_1):D267–D270, 01 2004. ISSN 0305-1048. . URL https://doi.org/10.1093/nar/gkh061.

[12] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie 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, page 1247–1250, New York, NY, USA, 2008. Association for Computing Machinery. ISBN 9781605581026. URL https://doi.org/10.1145/1376616.1376746.

[13] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In *Advances in neural information processing systems*, volume 26, pages 2787–2795, Stateline, Nevada, USA, 2013. URL https://papers.nips.cc/paper/2013/file/1cecc7a77928ca8133fa24680a88d2f9-Paper.pdf.

[14] Samuel 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)*, pages 677–685, Hong Kong, China, 2019. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/K19-1063.

[15] 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):1616–1637, sep 2018. ISSN 1558-2191. .

[16] Yixin Cao, Lifu Huang, Heng Ji, Xu Chen, and Juanzi 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)*, pages 1623–1633, Vancouver, Canada, 2017. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P17-1149.

[17] Yixin Cao, Lei Hou, Juanzi Li, and Zhiyuan Liu. Neural collective entity linking. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 675–686, Santa Fe, New Mexico, USA, 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/C18-1057.

[18] Diego Ceccarelli, Claudio Lucchese, Salvatore Orlando, Raffaele Perego, and Salvatore Trani. Learning relatedness measures for entity linking. In *Proceedings of the 22Nd ACM International Conference on Information & Knowledge Management*, CIKM '13, pages 139–148, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-2263-8. URL http://doi.acm.org/10.1145/2505515.2505711.

[19] Xiao Cheng and Dan Roth. Relational inference for Wikification. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1787–1796, Seattle, Washington, USA, 2013. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D13-1184.

[20] Andrew Chisholm and Ben Hachey. Entity disambiguation with web links. *Transactions of the Association for Computational Linguistics*, 3:145–156, 2015. URL https://www.aclweb.org/anthology/Q15-1011.

[21] Eunsol Choi, Omer Levy, Yejin Choi, and Luke Zettlemoyer. Ultra-fine entity typing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 87–96, Melbourne, Australia, 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P18-1009.

[22] Junyoung Chung, Caglar Gulcehre, Kyunghyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. In *NIPS 2014 Workshop on Deep Learning*, Montréal, Canada, 2014. URL https://arxiv.org/abs/1412.3555.

[23] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *Journal of machine learning research*, 12(Aug):2493–2537, 2011.

[24] Ryan Cotterell and Kevin 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)*, pages 91–96, Taipei, Taiwan, 2017. Asian Federation of Natural Language Processing. URL https://www.aclweb.org/anthology/I17-2016.

[25] Silviu 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)*, pages 708–716, Prague, Czech Republic, 2007. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D07-1074/.

[26] Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. Autoregressive entity retrieval. In *International Conference on Learning Representations*, 2021.

[27] Danilo Dessì, Francesco Osborne, Diego Reforgiato Recupero, Davide Buscaldi, and Enrico Motta. Generating knowledge graphs by employing natural language processing and machine learning techniques within the scholarly domain. *Future Generation Computer Systems*, 116:253–264, 2021. ISSN 0167-739X. . URL https://www.sciencedirect.com/science/article/pii/S0167739X2033003X.

[28] Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. Convolutional 2D Knowledge Graph Embeddings. In *Proceedings of AAAI Conference on Artificial In-*

telligence, 2018. URL https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/17366.

[29] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina 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)*, pages 4171–4186, Minneapolis, Minnesota, 2019. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/N19-1423.

[30] Maud Ehrmann, Matteo Romanello, Alex Flückiger, and Simon Clematide. Overview of CLEF HIPE 2020: Named entity recognition and linking on historical newspapers. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 288–310. Springer, 2020.

[31] Cheikh Brahim El Vaigh, François Goasdoué, Guillaume Gravier, and Pascale Sébillot. Using knowledge base semantics in context-aware entity linking. In *Proceedings of the ACM Symposium on Document Engineering 2019*, DocEng '19, New York, NY, USA, 2019. ACM. ISBN 9781450368872. . URL https://doi.org/10.1145/3342558.3345393.

[32] Joe Ellis, Jeremy Getman, Dana Fore, Neil Kuster, Zhiyi Song, Ann Bies, and Stephanie 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*, Gaithersburg, Maryland, USA, 2015. NIST. URL https://tac.nist.gov/publications/2015/additional.papers/TAC2015.KBP_resources_overview.proceedings.pdf.

[33] Yotam Eshel, Noam Cohen, Kira Radinsky, Shaul Markovitch, Ikuya Yamada, and Omer Levy. Named entity disambiguation for noisy text. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 58–68, Vancouver, Canada, 2017. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/K17-1008.

[34] Hannah Eyre, Alec B Chapman, Kelly S Peterson, Jianlin Shi, Patrick R Alba, Makoto M Jones, Tamara L Box, Scott L DuVall, and Olga V Patterson. Launching into clinical space with medspaCy: a new clinical text processing toolkit in Python. *arXiv preprint arXiv:2106.07799*, 2021.

[35] Wei Fang, Jianwen Zhang, Dilin Wang, Zheng Chen, and Ming Li. Entity disambiguation by knowledge and text jointly embedding. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 260–269, Berlin, Germany, 2016. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/K16-1026.

[36] Zheng Fang, Yanan Cao, Qian Li, Dongjie Zhang, Zhenyu Zhang, and Yanbing Liu. Joint entity linking with deep reinforcement learning. In *The World Wide Web Conference*, WWW '19, pages 438–447, New York, NY, USA, 2019. ACM. ISBN 978-1-4503-6674-8. URL http://doi.acm.org/10.1145/3308558.3313517.

[37] Michael Färber, Frederic Bartscherer, Carsten Menne, and Achim Rettinger. Linked Data Quality of DBpedia, Freebase, OpenCyc, Wikidata, and YAGO. *Semantic Web*, 9(1):77–129, 2018.

[38] Christiane Fellbaum, editor. *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, MA, 1998. ISBN 978-0-262-06197-1.

[39] Paolo Ferragina and Ugo 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, page 1625–1628, New York, NY, USA, 2010. ACM. ISBN 9781450300995. URL https://doi.org/10.1145/1871437.1871689.

[40] Thibault Févry, Livio Baldini Soares, Nicholas FitzGerald, Eunsol Choi, and Tom Kwiatkowski. Entities as Experts: Sparse Memory Access with Entity Supervision. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4937–4951, Online, November 2020. Association for Computational Linguistics. . URL https://aclanthology.org/2020.emnlp-main.400.

[41] Matthew Francis-Landau, Greg Durrett, and Dan 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*, pages 1256–1261, San Diego, California, USA, 2016. URL https://www.aclweb.org/anthology/N16-1150.

[42] Giorgio Fumera, Fabio Roli, and Giorgio Giacinto. Reject option with multiple thresholds. *Pattern recognition*, 33(12): 2099–2101, 2000.

[43] Evgeniy Gabrilovich, Michael Ringgaard, and Amarnag 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/.

[44] Octavian-Eugen Ganea and Thomas Hofmann. Deep joint entity disambiguation with local neural attention. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2619–2629, Copenhagen, Denmark, 2017. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D17-1277.

[45] Octavian-Eugen Ganea, Marina Ganea, Aurelien Lucchi, Carsten Eickhoff, and Thomas Hofmann. Probabilistic bag-of-hyperlinks model for entity linking. In *Proceedings of the 25th International Conference on World Wide Web*, WWW '16, pages 927–938, Republic and Canton of Geneva, Switzerland, 2016. International World Wide Web Conferences Steering Committee. ISBN 978-1-4503-4143-1. URL https://doi.org/10.1145/2872427.2882988.

[46] Daniel Gillick, Sayali Kulkarni, Larry Lansing, Alessandro Presta, Jason Baldridge, Eugene Ie, and Diego Garcia-Olano. Learning dense representations for entity retrieval. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 528–537, Hong Kong, China, 2019. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/K19-1049.

[47] Amir Globerson, Nevena Lazic, Soumen Chakrabarti, Amarnag Subramanya, Michael Ringgaard, and Fernando 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)*, pages 621–631, Berlin, Germany, 2016. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P16-1059.

[48] Archana Goyal, Vishal Gupta, and Manish Kumar. Recent Named Entity Recognition and Classification techniques: A systematic review. *Computer Science Review*, 29:21–43, 2018. ISSN 1574-0137. . URL https://www.sciencedirect.com/science/article/pii/S1574013717302782.

[49] Palash Goyal and Emilio Ferrara. Graph embedding techniques, applications, and performance: A survey. *Knowledge-Based Systems*, 2018. ISSN 0950-7051. . URL http://www.sciencedirect.com/science/article/pii/S0950705118301540.

[50] Aditya Grover and Jure Leskovec. Node2vec: Scalable Feature Learning for Networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 855–864, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450342322. . URL https://doi.org/10.1145/2939672.2939754.

[51] Zhaochen Guo and Denilson Barbosa. Robust named entity disambiguation with random walks. *Semantic Web*, 9(4): 459 – 479, 2018. URL https://content.iospress.com/articles/semantic-web/sw273.

[52] Nitish Gupta, Sameer Singh, and Dan Roth. Entity linking via joint encoding of types, descriptions, and context. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2681–2690, Copenhagen, Denmark, 2017. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D17-1284.

[53] Xu Han, Hao Zhu, Pengfei Yu, Ziyun Wang, Yuan Yao, Zhiyuan Liu, and Maosong 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*, pages 4803–4809, Brussels, Belgium, 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D18-1514.

[54] Martin E Hellman. The nearest neighbor classification rule with a reject option. *IEEE Transactions on Systems Science and Cybernetics*, 6(3):179–185, 1970.

[55] Radu Herbei and Marten H Wegkamp. Classification with reject option. *The Canadian Journal of Statistics/La Revue Canadienne de Statistique*, pages 709–721, 2006.

[56] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[57] Johannes Hoffart, Mohamed Amir Yosef, Ilaria Bordino, Hagen Fürstenau, Manfred Pinkal, Marc Spaniol, Bilyana Taneva, Stefan Thater, and Gerhard Weikum. Robust disambiguation of named entities in text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 782–792. Association for Computational Linguistics, 2011. ISBN 978-1-937284-11-4. URL http://dl.acm.org/citation.cfm?id=2145432.2145521.

[58] Aidan Hogan, Eva Blomqvist, Michael Cochez, Claudia d'Amato, Gerard de Melo, Claudio Gutierrez, José Emilio Labra Gayo, Sabrina Kirrane, Sebastian Neumaier, Axel Polleres, Roberto Navigli, Axel-Cyrille Ngonga Ngomo, Sabbir M. Rashid, Anisa Rula, Lukas Schmelzeisen, Juan Sequeda, Steffen Staab, and Antoine Zimmermann. Knowledge graphs, 2021. URL https://arxiv.org/abs/2003.02320.

[59] Hongzhao Huang, Larry Heck, and Heng Ji. Leveraging deep neural networks and knowledge graphs for entity disambiguation. *arXiv preprint arXiv:1504.07678*, 2015. URL https://arxiv.org/abs/1504.07678.

[60] Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux, and Jason Weston. Poly-encoders: Architectures and pre-training strategies for fast and accurate multi-sentence scoring. In *International Conference on Learning Representations*, 2019.

[61] Kalervo Järvelin and Jaana Kekäläinen. Cumulated Gain-Based Evaluation of IR Techniques. *ACM Trans. Inf. Syst.*, 20(4):422–446, October 2002. ISSN 1046-8188. URL https://doi.org/10.1145/582415.582418.

[62] Heng Ji and Ralph Grishman. Knowledge base population: Successful approaches and challenges. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pages 1148–1158, 2011.

[63] Heng Ji, Ralph Grishman, Hoa Trang Dang, Kira Griffitt, and Joe Ellis. Overview of the TAC 2010 knowledge base population track. In *Third Text Analysis Conference (TAC)*, Gaithersburg, Maryland, USA, 2010. URL https://blender.cs.illinois.edu/paper/kbp2010overview.pdf.

[64] Heng Ji, Joel Nothman, Ben Hachey, and Radu Florian. Overview of TAC-KBP2015 tri-lingual entity discovery and linking. In *Proceedings of the 2015 Text Analysis Conference, TAC 2015*, pages 16–17, Gaithersburg, Maryland, USA, 2015. NIST. URL https://tac.nist.gov/publications/2015/additional.papers/TAC2015.KBP_Trilingual_Entity_Discovery_and_Linking_overview.proceedings.pdf.

[65] Shaoxiong Ji, Shirui Pan, Erik Cambria, Pekka Marttinen, and Philip S. Yu. A survey on knowledge graphs: representation, acquisition, and applications. *IEEE Transactions on Neural Networks and Learning Systems*, April 2021. ISSN 2162-237X. . Publisher Copyright: IEEE Copyright: Copyright 2021 Elsevier B.V., All rights reserved.

[66] Karen Spärck Jones, Shelia Walker, and Stephen E. Robertson. A probabilistic model of information retrieval: Development and comparative experiments part 2. *Information Processing & Management*, 36(6):809–840, 2000. ISSN 0306-4573. . URL https://doi.org/10.1016/S0306-4573(00)00016-9.

[67] Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke 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)*, pages 1601–1611, Vancouver, Canada, July 2017. Association for Computational Linguistics. . URL https://aclanthology.org/P17-1147.

[68] Rijula Kar, Susmija Reddy, Sourangshu Bhattacharya, Anirban Dasgupta, and Soumen Chakrabarti. Task-specific representation learning for web-scale entity disambiguation. In *The Thirty-Second AAAI Conference on Artificial Intelligence*, pages 5812–5819, New Orleans, Louisiana, USA, 2018. AAAI Press. URL https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/viewFile/17281/16144.

[69] Ritu Khare, Jiao Li, and Zhiyong Lu. LabeledIn: cataloging labeled indications for human drugs. *Journal of biomedical informatics*, 52:448–456, 2014. .

[70] Nikolaos Kolitsas, Octavian-Eugen Ganea, and Thomas Hofmann. End-to-End Neural Entity Linking. In *Proceedings of the 22nd Conference on Computational Natural Language Learning*, pages 519–529, Brussels, Belgium, 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/K18-1050.

[71] Zeljko Kraljevic, Thomas Searle, Anthony Shek, Lukasz Roguski, Kawsar Noor, Daniel Bean, Aurelie Mascio, Leilei Zhu, Amos A. Folarin, Angus Roberts, Rebecca Bendayan, Mark P. Richardson, Robert Stewart, Anoop D. Shah, Wai Keong Wong, Zina Ibrahim, James T. Teo, and Richard J.B. Dobson. Multi-domain clinical natural language processing with MedCAT: The Medical Concept Annotation Toolkit. *Artificial Intelligence in Medicine*, 117:102083, 2021. ISSN 0933-3657. . URL https://www.sciencedirect.com/science/article/pii/S0933365721000762.

[72] Nevena Lazic, Amarnag Subramanya, Michael Ringgaard, and Fernando Pereira. Plato: A selective context model for entity resolution. *Transactions of the Association for Computational Linguistics*, 3:503–515, 2015. URL https://www.aclweb.org/anthology/Q15-1036.

[73] Phong Le and Ivan 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)*, pages 1595–1604, Melbourne, Australia, 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P18-1148.

[74] Phong Le and Ivan Titov. Distant learning for entity linking with automatic noise detection. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4081–4090, Florence, Italy, 2019. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P19-1400.

[75] Phong Le and Ivan Titov. Boosting entity linking performance by leveraging unlabeled documents. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1935–1945, Florence, Italy, 2019. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P19-1187.

[76] Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, page 2, Atlanta, USA, 2013. JMLR. URL http://deeplearning.net/wp-content/uploads/2013/03/pseudo_label_final.pdf.

[77] Jinhyuk Lee, Sean S. Yi, Minbyul Jeong, Mujeen Sung, Won-Jin Yoon, Yonghwa Choi, Miyoung Ko, and Jaewoo 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*, Online, December 2020. Association for Computational Linguistics. . URL https://www.aclweb.org/anthology/2020.nlpcovid19-2.1.

[78] Sunwon Lee, Donghyeon Kim, Kyubum Lee, Jaehoon Choi, Seongsoon Kim, Minji Jeon, Sangrak Lim, Donghee Choi, Sunkyu Kim, Aik-Choon Tan, and Jaewoo Kang. BEST: Next-Generation Biomedical Entity Search Tool for Knowledge Discovery from Biomedical Literature. *PLOS ONE*, 11 (10):1–16, 10 2016. . URL https://doi.org/10.1371/journal.pone.0164680.

[79] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke 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*, pages 7871–7880, Online, July 2020. Association for Computational Linguistics. . URL https://aclanthology.org/2020.acl-main.703.

[80] Xiao Ling, Sameer Singh, and Daniel S. Weld. Design challenges for entity linking. *Transactions of the Association for Computational Linguistics*, 3:315–328, 2015. URL https://www.aclweb.org/anthology/Q15-1023/.

[81] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019. URL https://arxiv.org/abs/1907.11692.

[82] Roi Livni, Shai Shalev-Shwartz, and Ohad Shamir. On the computational efficiency of training neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'14, page 855–863, Cambridge, MA, USA, 2014. MIT Press.

[83] Lajanugen Logeswaran, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, Jacob Devlin, and Honglak Lee. Zero-shot entity linking by reading entity descriptions. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3449–3460, Florence, Italy, 2019. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P19-1335.

[84] Daniel Loureiro and Alípio Mário Jorge. Medlinker: Medical entity linking with neural representations and dictionary matching. In Joemon M. Jose, Emine Yilmaz, João Magalhães, Pablo Castells, Nicola Ferro, Mário J. Silva, and Flávio Martins, editors, *Advances in Information Retrieval*, pages 230–237, Cham, 2020. Springer International Publishing. ISBN 978-3-030-45442-5.

[85] Gang Luo, Xiaojiang Huang, Chin-Yew Lin, and Zaiqing Nie. Joint entity recognition and disambiguation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 879–888, Lisbon, Portugal, 2015. URL https://www.aclweb.org/anthology/D15-1104/.

[86] Grace Luo, Trevor Darrell, and Anna Rohrbach. NewsCLIPpings: Automatic Generation of Out-of-Context Multimodal Media. *arXiv preprint arXiv:2104.05893*, 2021.

[87] Sean MacAvaney, Sajad Sotudeh, Arman Cohan, Nazli Goharian, Ish Talati, and Ross 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, page 1013–1016, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450361729. . URL https://doi.org/10.1145/3331184.3331319.

[88] José L. Martínez-Rodríguez, A. Hogan, and I. López-Arévalo. Information extraction meets the Semantic Web: A survey. *Semantic Web*, 11(2):255–335, 2020.

[89] Pedro Henrique Martins, Zita Marinho, and André 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*, pages 190–196, Florence, Italy, 2019. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P19-2026.

[90] Pablo N. Mendes, Max Jakob, Andrés García-Silva, and Christian Bizer. DBpedia Spotlight: Shedding Light on the Web of Documents. In *Proceedings of the 7th International Conference on Semantic Systems*, I-Semantics '11, pages 1–8,

New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0621-8. URL http://doi.acm.org/10.1145/2063518.2063519.

[91] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S. Corrado, and Jeffrey 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, page 3111–3119, Red Hook, NY, USA, 2013. Curran Associates Inc. URL https://papers.nips.cc/paper/2013/file/9aa42b31882ec039965f3c4923ce901b-Paper.pdf.

[92] David Milne and Ian H. Witten. Learning to link with Wikipedia. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, CIKM '08, pages 509–518, New York, NY, USA, 2008. ACM. ISBN 978-1-59593-991-3. URL http://doi.acm.org/10.1145/1458082.1458150.

[93] Jose G. Moreno, Romaric Besançon, Romain Beaumont, Eva D'hondt, Anne-Laure Ligozat, Sophie Rosset, Xavier Tannier, and Brigitte Grau. Combining word and entity embeddings for entity linking. In *Extended Semantic Web Conference (1)*, volume 10249 of *Lecture Notes in Computer Science*, pages 337–352, 2017. URL https://perso.limsi.fr/bg/fichiers/2017/combining-word-entity-eswc2017.pdf.

[94] Andrea Moro, Alessandro Raganato, and Roberto Navigli. Entity Linking meets Word Sense Disambiguation: a Unified Approach. *Transactions of the Association for Computational Linguistics*, 2:231–244, 2014. URL https://www.aclweb.org/anthology/Q14-1019/.

[95] David Mueller and Greg Durrett. Effective use of context in noisy entity linking. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1024–1029, Brussels, Belgium, 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D18-1126.

[96] David Nadeau and Satoshi Sekine. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26, 2007. URL https://nlp.cs.nyu.edu/sekine/papers/li07.pdf.

[97] Roberto Navigli. Word sense disambiguation: A survey. *ACM Comput. Surv.*, 41(2):10:1–10:69, 2009. ISSN 0360-0300. URL http://doi.acm.org/10.1145/1459352.1459355.

[98] Mojtaba Nayyeri, Sahar Vahdati, Jens Lehmann, and Hamed Shariat Yazdi. Soft marginal transe for scholarly knowledge graph completion. *CoRR*, abs/1904.12211, 2019. URL http://arxiv.org/abs/1904.12211.

[99] Mojtaba Nayyeri, Sahar Vahdati, Can Aykul, and Jens Lehmann. 5* Knowledge Graph Embeddings with Projective Transformations. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(10):9064–9072, May 2021. URL https://ojs.aaai.org/index.php/AAAI/article/view/17095.

[100] Rostislav Nedelchev, Debanjan Chaudhuri, Jens Lehmann, and Asja Fischer. End-to-End Entity Linking and Disambiguation leveraging Word and Knowledge Graph Embeddings. *CoRR*, abs/2002.11143, 2020. URL https://arxiv.org/abs/2002.11143.

[101] Denis Newman-Griffis, Albert M. Lai, and Eric Fosler-Lussier. Jointly embedding entities and text with distant supervision. In *Proceedings of The Third Workshop on Representation Learning for NLP*, pages 195–206, Melbourne, Australia, 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W18-3026.

[102] Dat Ba Nguyen, Martin Theobald, and Gerhard Weikum. J-NERD: joint named entity recognition and disambiguation with rich linguistic features. *Transactions of the Association for Computational Linguistics*, 4:215–229, 2016. URL https://www.aclweb.org/anthology/Q16-1016/.

[103] Thien Huu Nguyen, Nicolas Fauceglia, Mariano Rodriguez Muro, Oktie Hassanzadeh, Alfio Massimiliano Gliozzo, and Mohammad 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*, pages 2310–2320, Osaka, Japan, 2016. The COLING 2016 Organizing Committee. URL https://www.aclweb.org/anthology/C16-1218.

[104] Maximilian Nickel, Volker Tresp, and Hans-Peter 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, page 809–816, Madison, WI, USA, 2011. Omnipress. ISBN 9781450306195.

[105] Italo L. Oliveira, Renato Fileto, René Speck, Luís P.F. Garcia, Diego Moussallem, and Jens Lehmann. Towards holistic entity linking: Survey and directions. *Information Systems*, 95:101624, 2021. ISSN 0306-4379. URL http://www.sciencedirect.com/science/article/pii/S0306437920300958.

[106] Yasumasa Onoe and Greg Durrett. Fine-grained entity typing for domain independent entity linking. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8576–8583, Apr. 2020. . URL https://ojs.aaai.org/index.php/AAAI/article/view/6380.

[107] Pekka Orponen. Computational Complexity of Neural Networks: A Survey. *Nordic J. of Computing*, 1(1):94–110, March 1994. ISSN 1236-6064.

[108] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The PageRank Citation Ranking: Bringing Order to the Web. Technical Report 1999-66, Stanford InfoLab, November 1999. URL http://ilpubs.stanford.edu:8090/422/. Previous number = SIDL-WP-1999-0120.

[109] Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng 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)*, pages 1946–1958, Vancouver, Canada, 2017. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P17-1178/.

[110] Alberto Parravicini, Rhicheek Patra, Davide B. Bartolini, and Marco 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, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450367899. . URL https://doi.org/10.1145/3327964.3328499.

[111] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. DeepWalk: Online Learning of Social Representations. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 701–710, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2956-9. . URL http://doi.acm.org/10.1145/2623330.2623732.

[112] Maria Pershina, Yifan He, and Ralph 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*, pages 238–243, Denver, Colorado, USA, 2015. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/N15-1026.

[113] Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *The Thirty-Second AAAI Conference on Artificial Intelligence*, pages 2227–2237, New Orleans, Louisiana, USA, 2018. AAAI Press. URL https://arxiv.org/abs/1802.05365.

[114] Matthew E. Peters, Mark Neumann, Robert Logan, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah 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)*, pages 43–54, Hong Kong, China, 2019. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D19-1005.

[115] Francesco Piccinno and Paolo Ferragina. From TagME to WAT: A New Entity Annotator. In *Proceedings of the First International Workshop on Entity Recognition |& Disambiguation*, ERD '14, pages 55 – 62, New York, NY, USA, 2014. Association for Computing Machinery. ISBN 9781450330237. URL https://doi.org/10.1145/2633211.2634350.

[116] Vera Provatorova, Svitlana Vakulenko, Evangelos Kanoulas, Koen Dercksen, and Johannes M van Hulst. Named Entity Recognition and Linking on Historical Newspapers: UvA. ILPS & REL at CLEF HIPE 2020. In *CLEF (Working Notes)*, 2020.

[117] Priya Radhakrishnan, Partha Talukdar, and Vasudeva 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)*, pages 1844–1853, New Orleans, Louisiana, 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/N18-1167.

[118] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020. URL http://jmlr.org/papers/v21/20-074.html.

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

[120] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas, November 2016. Association for Computational Linguistics. . URL https://aclanthology.org/D16-1264.

[121] Lev Ratinov, Dan Roth, Doug Downey, and Mike 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, pages 1375–1384, Portland, Oregon, USA, 2011. Association for Computational Linguistics. ISBN 978-1-932432-87-9. URL http://dl.acm.org/citation.cfm?id=2002472.2002642.

[122] Nils Reimers and Iryna 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*, pages 338–348, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. . URL https://aclanthology.org/D17-1035.

[123] Nils Reimers and Iryna 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)*, pages 3982–3992, Hong Kong, China, November 2019. Association for Computational Linguistics. . URL https://aclanthology.org/D19-1410.

[124] Shruti Rijhwani, Jiateng Xie, Graham Neubig, and Jaime Carbonell. Zero-shot neural transfer for cross-lingual entity linking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6924–6931, Honolulu, Hawaii, USA, 2019. URL https://ojs.aaai.org/index.php/AAAI/article/download/4670/4548.

[125] Giuseppe Rizzo, Marieke van Erp, and Raphaël 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)*, pages 4593–4600, Reykjavik, Iceland, 2014. European Language Resources Association (ELRA). URL http://www.lrec-conf.org/proceedings/lrec2014/pdf/176_Paper.pdf.

[126] Michael Röder, Ricardo Usbeck, and Axel-Cyrille Ngonga Ngomo. GERBIL - Benchmarking Named Entity Recognition and Linking consistently. *Semantic Web*, 9(5):605–625, 2018. . URL http://www.semantic-web-journal.net/system/files/swj1671.pdf.

[127] Daniel Ruffinelli, Samuel Broscheit, and Rainer Gemulla. You CAN Teach an Old Dog New Tricks! On Training Knowledge Graph Embeddings. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=BkxSmlBFvr.

[128] Guergana K Savova, James J Masanz, Philip V Ogren, Jiaping Zheng, Sunghwan Sohn, Karin C Kipper-Schuler, and Christopher 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):507–513, 2010.

[129] Özge Sevgili, Alexander Panchenko, and Chris Biemann. Improving neural entity disambiguation with graph embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 315–322, Florence, Italy, 2019. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P19-2044/.

[130] Hamed Shahbazi, Xiaoli Fern, Reza Ghaeini, Chao Ma, Rasha Mohammad Obeidat, and Prasad Tadepalli. Joint neural entity disambiguation with output space search. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2170–2180, Santa Fe, New Mexico, USA, August 2018. Association for Computational Linguistics. URL https://aclanthology.org/C18-1184.

[131] Hamed Shahbazi, Xiaoli Z Fern, Reza Ghaeini, Rasha Obeidat, and Prasad Tadepalli. Entity-aware ELMo: Learning Contextual Entity Representation for Entity Disambiguation. *arXiv preprint arXiv:1908.05762*, 2019. URL https://arxiv.org/abs/1908.05762.

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

[133] Wei Shen, Jianyong Wang, and Jiawei Han. Entity Linking with a Knowledge Base: Issues, Techniques, and Solutions. *Transactions on Knowledge & Data Engineering*, 27(2):443–460, 2015. URL http://www.computer.org/csdl/trans/tk/2015/02/06823700-abs.html.

[134] Wei Shen, Jiawei Han, Jianyong Wang, Xiaojie Yuan, and Zhenglu Yang. SHINE+: A General Framework for Domain-Specific Entity Linking with Heterogeneous Information Networks. *IEEE Transactions on Knowledge and Data Engineering*, 30(2):353–366, 2018. .

[135] Wei Shi, Siyuan Zhang, Zhiwei Zhang, Hong Cheng, and Jeffrey Xu Yu. Joint embedding in named entity linking on sentence level. *arXiv preprint arXiv:2002.04936*, 2020. URL https://arxiv.org/abs/2002.04936.

[136] Ilya Shnayderman, Liat Ein-Dor, Yosi Mass, Alon Halfon, Benjamin Sznajder, Artem Spector, Yoav Katz, Dafna Sheinwald, Ranit Aharonov, and Noam Slonim. Fast End-to-End Wikification. *arXiv preprint arXiv:1908.06785*, 2019.

[137] Avirup Sil, Gourab Kundu, Radu Florian, and Wael Hamza. Neural cross-lingual entity linking. In *The Thirty-Second AAAI Conference on Artificial Intelligence*, New Orleans, Louisiana, USA., 2018. AAAI Press. URL https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16501/16101.

[138] Jiří Šíma and Pekka Orponen. General-purpose computation with neural networks: A survey of complexity theoretic results. *Neural Computation*, 15(12):2727–2778, 2003.

[139] Noam Slonim, Yonatan Bilu, Carlos Alzate, Roy Bar-Haim, Ben Bogin, Francesca Bonin, Leshem Choshen, Edo Cohen-Karlik, Lena Dankin, Lilach Edelstein, et al. An autonomous debating system. *Nature*, 591(7850):379–384, 2021.

[140] Luca Soldaini and Nazli Goharian. QuickUMLS: a fast, unsupervised approach for medical concept extraction. In *MedIR workshop, SIGIR*, pages 1–4, 2016.

[141] Daniil Sorokin and Iryna 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*, pages 65–75, 2018. URL https://www.aclweb.org/anthology/S18-2007/.

[142] Valentin I. Spitkovsky and Angel X. Chang. A cross-lingual dictionary for English Wikipedia concepts. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 3168–3175, Istanbul, Turkey, 2012. European Language Resources Association (ELRA). URL http://www.lrec-conf.org/proceedings/lrec2012/pdf/266_Paper.pdf.

[143] Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. YAGO: A Core of Semantic Knowledge. In *Proceedings of the 16th International Conference on World Wide Web*, WWW '07, page 697–706, New York, NY, USA, 2007. ACM. ISBN 9781595936547. . URL https://doi.org/10.1145/1242572.1242667.

[144] Yaming Sun, Lei Lin, Duyu Tang, Nan Yang, Zhenzhou Ji, and Xiaolong Wang. Modeling mention, context and entity with neural networks for entity disambiguation. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, pages 1333–1339. AAAI Press, 2015. ISBN 978-1-57735-738-4. URL http://dl.acm.org/citation.cfm?id=2832415.2832435.

[145] Mujeen Sung, Hwisang Jeon, Jinhyuk Lee, and Jaewoo Kang. Biomedical entity representations with synonym marginalization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3641–3650, Online, July 2020. Association for Computational Linguistics. . URL https://www.aclweb.org/anthology/2020.acl-main.335.

[146] Erik F. Tjong Kim Sang and Fien 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*, pages 142–147, 2003. URL https://aclanthology.org/W03-0419.

[147] Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. Complex embeddings for simple link prediction. In *International Conference on Machine Learning (ICML)*, volume 48, pages 2071–2080, 2016.

[148] Chen-Tse Tsai and Dan 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 Technologies*, pages 589–598, San Diego, California, USA, 2016. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/N16-1072.

[149] Chen-Tse Tsai and Dan Roth. Learning better name translation for cross-lingual Wikification. In *Thirty-Second AAAI Conference on Artificial Intelligence*, New Orleans, Louisiana, USA, 2018. AAAI Press. URL https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/17318/16109.

[150] Shyam Upadhyay, Nitish Gupta, and Dan Roth. Joint multilingual supervision for cross-lingual entity linking. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2486–2495, Brussels, Belgium, 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D18-1270.

[151] Johannes M. van Hulst, Faegheh Hasibi, Koen Dercksen, Krisztian Balog, and Arjen 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*, SIGIR '20. ACM, 2020.

[152] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN

9781510860964. URL https://papers.nips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[153] Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. Pointer networks. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 2692–2700. Curran Associates, Inc., 2015. URL http://papers.nips.cc/paper/5866-pointer-networks.pdf.

[154] Denny Vrandečić and Markus Krötzsch. Wikidata: A free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85, September 2014. ISSN 0001-0782. URL https://doi.org/10.1145/2629489.

[155] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel 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*, pages 353–355, Brussels, Belgium, 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W18-5446.

[156] Han Wang, Jin Guang Zheng, Xiaogang Ma, Peter Fox, and Heng Ji. Language and domain independent entity linking with quantified collective validation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 695–704, Lisbon, Portugal, 2015. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D15-1081.

[157] Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering*, 29(12):2724–2743, 2017. .

[158] Xiaozhi Wang, Tianyu Gao, Zhaocheng Zhu, Zhengyan Zhang, Zhiyuan Liu, Juanzi Li, and Jian Tang. KEPLER: A unified model for knowledge embedding and pre-trained language representation. *Transactions of the Association for Computational Linguistics*, 9:176–194, 2021.

[159] Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. Knowledge Graph Embedding by Translating on Hyperplanes. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, AAAI'14, page 1112–1119. AAAI Press, 2014.

[160] Ledell Wu, Fabio Petroni, Martin Josifoski, Sebastian Riedel, and Luke Zettlemoyer. Scalable zero-shot entity linking with dense entity retrieval. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6397–6407, Online, November 2020. Association for Computational Linguistics. . URL https://aclanthology.org/2020.emnlp-main.519.

[161] Vikas Yadav and Steven Bethard. A survey on recent advances in named entity recognition from deep learning models. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2145–2158, Santa Fe, New Mexico, NM, USA, 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/C18-1182.

[162] Ikuya Yamada, Hiroyuki Shindo, Hideaki Takeda, and Yoshiyasu 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*, pages 250–259, Berlin, Germany, 2016. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/K16-1025.

[163] Ikuya Yamada, Hiroyuki Shindo, Hideaki Takeda, and Yoshiyasu Takefuji. Learning distributed representations of texts and entities from knowledge base. *Transactions of the Association for Computational Linguistics*, 5:397–411, 2017. . URL https://aclanthology.org/Q17-1028.

[164] Ikuya Yamada, Akari Asai, Jin Sakuma, Hiroyuki Shindo, Hideaki Takeda, Yoshiyasu Takefuji, and Yuji 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*, pages 23–30, Online, October 2020. Association for Computational Linguistics. . URL https://aclanthology.org/2020.emnlp-demos.4.

[165] Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji 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)*, pages 6442–6454, Online, November 2020. Association for Computational Linguistics. . URL https://aclanthology.org/2020.emnlp-main.523.

[166] Ikuya Yamada, Koki Washio, Hiroyuki Shindo, and Yuji Matsumoto. Global entity disambiguation with pretrained contextualized embeddings of words and entities. *arXiv preprint arXiv:1909.00426v2*, 2020. URL https://arxiv.org/abs/1909.00426v2.

[167] Bishan Yang, Scott Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. Embedding entities and relations for learning and inference in knowledge bases. In *Proceedings of the International Conference on Learning Representations (ICLR) 2015*, May 2015. URL https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/ICLR2015_updated.pdf.

[168] Xiyuan Yang, Xiaotao Gu, Sheng Lin, Siliang Tang, Yueting Zhuang, Fei Wu, Zhigang Chen, Guoping Hu, and Xiang 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)*, pages 271–281, Hong Kong, China, 2019. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D19-1026.

[169] Wen-tau Yih, Ming-Wei Chang, Xiaodong He, and Jianfeng 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)*, pages 1321–1331, Beijing, China, 2015. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P15-1128/.

[170] Tom Young, Devamanyu Hazarika, Soujanya Poria, and Erik Cambria. Recent trends in deep learning based natural language processing. *IEEE Computational Intelligence Magazine*, 13(3):55–75, 2018. URL https://ieeexplore.ieee.org/document/8416973.

[171] Yisong Yue, Thomas Finley, Filip Radlinski, and Thorsten Joachims. A support vector method for optimizing average precision. In *Proceedings of the 30th Annual Interna-*

tional ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '07, page 271–278, New York, NY, USA, 2007. Association for Computing Machinery. ISBN 9781595935977. URL https://doi.org/10.1145/1277741.1277790.

[172] Sheng Zhang, Xiaodong Liu, Jingjing Liu, Jianfeng Gao, Kevin Duh, and Benjamin Van Durme. ReCoRD: Bridging the Gap between Human and Machine Commonsense Reading Comprehension. *CoRR*, abs/1810.12885, 2018. URL http://arxiv.org/abs/1810.12885.

[173] Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D. Manning. Position-aware attention and supervised data improve slot filling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP 2017)*, pages 35–45, Copenhagen, Denmark, 2017. ACL. URL https://nlp.stanford.edu/pubs/zhang2017tacred.pdf.

[174] Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. ERNIE: Enhanced language representation with informative entities. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1441–1451, Florence, Italy, 2019. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P19-1139.

[175] Shuyan Zhou, Shruti Rijhwani, John Wieting, Jaime Carbonell, and Graham Neubig. Improving candidate generation for low-resource cross-lingual entity linking. *Transactions of the Association for Computational Linguistics*, 8: 109–124, 2020. URL https://www.mitpressjournals.org/doi/full/10.1162/tacl_a_00303.

[176] Stefan Zwicklbauer, Christin Seifert, and Michael 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, pages 425–434, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-4069-4. URL http://doi.acm.org/10.1145/2911451.2911535.