

Implicit Entity Linking in Tweets: an Ad-hoc Retrieval Approach

Hawre Hosseini [a,*], Tam T. Nguyen [a], Jimmy Wu [a] and Ebrahim Bagheri [a]
[a] *Laboratory for Systems, Software and Semantics (LS^3), Ryerson University, Toronto, Canada*
E-mails: hawre.hosseini@ryerson.ca, nthanhtam@gmail.com, jimmy1.wu@ryerson.ca, bagheri@ryerson.ca

Abstract. Within the context of Twitter analytics, the notion of *implicit entity linking* has recently been introduced to refer to the identification of a named entity, which is central to the topic of the tweet, but whose surface form is not present in the tweet itself. Compared to traditional forms of entity linking where the linking process revolves around an identified surface form of a potential entity, implicit entity linking relies on contextual clues to determine whether an implicit entity is present within a given tweet and if so, which entity is being referenced. The objective of this paper, while introducing and publicly sharing a comprehensive gold standard dataset for implicit entity linking, is to perform the task of implicit entity linking. The dataset consists of 7,870 tweets, which are classified as either containing implicit entities, explicit entities, both, or neither. The implicit entities are then linked to three levels of entities on Wikipedia, namely coarse-grained level, e.g., *Person*, Fine-grained level, e.g., *Comedian*, and the actual entity, e.g., *Seinfeld*. The proposed model in this work formulates the problem of implicit entity linking as an *ad-hoc document retrieval* process where the input query is the tweet, which needs to be implicitly linked and the document space is the set of textual descriptions of entities in the knowledge base. The novel contributions of our work include: 1) designing and collecting a gold standard dataset for the task of implicit entity linking; 2) defining the implicit entity linking process as an ad-hoc document retrieval task; and 3) proposing a neural embedding-based feature function that is interpolated with prior term dependency and entity-based feature functions to enhance implicit entity linking. We systematically compare our work with existing work in this area and show that our method is able to provide improvements on a number of retrieval measures.

Keywords: Implicit entity linking, Semantic retrieval, DBpedia, Knowledge graph

1. Introduction

The task of recognizing mentions of entities in a text and linking them to a knowledge base, e.g., DBpedia, is referred to as entity linking, which is now extensively studied for textual content of various types (Shen et al., 2015) and is an important building block in a variety of downstream applications (Basile et al., 2015). The main objective of this task is to connect between the entities' surface form in the text, i.e., their explicit mentions, and their corresponding knowledge base representations. Within the context of tweets, the core idea being discussed in a given tweet may not be easily recognizable. For instance, a tweet such as '`Then there's Ethan Hawke and Patricia Arquette, easily the best characters and best performances of the movie`' is referring to the *Boyhood* movie while the movie itself is not explicitly mentioned in the tweet; for this reason, an entity linking system will not link any of the phrases in the tweet to the *Boyhood* entity, since the majority of the

[*]Corresponding author. E-mail: hawre.hosseini@ryerson.ca.

state-of-the-art entity linking approaches only focus on explicitly mentioned entities (Chang et al., 2014; Derczynski et al., 2015a).

Linking tweets to entities was examined in the early work by Meij et al. (2012) who proposed to link explicit mentions of entities in tweets to their corresponding knowledge base entries. Differently, however, the notion of *implicit entity linking* was introduced and formulated by Perera et al. (2016) for tweets as the task of linking a tweet to an entity, which is not explicitly mentioned in the text but is core to its understanding. This definition has been adopted in our prior work as well (Hosseini et al., 2018). Implicit entities are often omitted by the users as they are taken for granted to be understood by the intended audience. As a first attempt to formalize and define this task, Perera et al.'s definition of the implicit linking process is as follows: given a Tweet *twt*, and the type of the implicit entity that is being sought ϑ, e.g., `wikidata:Film`, the objective of *implicit entity linking* is to retrieve an entity from DBpedia of type ϑ that would qualify as the implicit entity.

Specifically for Twitter content, implicit entity linking is an important task, as researchers have shown that a noticeable number of tweets, more specifically 21% of tweets about movies and 40% of tweets about books, contain *implicit* references to the entity being discussed (Perera et al., 2016). Also, in the dataset collected and presented as a part of this paper, 15% of the tweets discussed implicit mentions of entities. From a technical perspective, implicit entity linking can be seen as an unsupervised tweet classification problem with an unbound number of classes, i.e., the number of classes are equivalent to the number of possible entities in the knowledge base. Given the large number of possible entities, it is impractical, even if at all possible, to employ classical text classification methods to perform implicit entity linking.

Perera et al. (2016), as the main prior work, has made extensive use of the knowledge graph for building tweet representations. In so doing, they have leveraged explicit mentions of entities within the input tweet. Their main hypothesis has been that observable explicit entities within the tweet itself, as well as other temporally relevant tweets, are indicative of the implicit entities that are being discussed.

In this paper, while building on Perera et al. (2016), we develop a model to perform the task of implicit entity linking through ad-hoc document retrieval. In this model, we view the input tweet containing an implicit mention as the query and try to search for entities which are most probably referred to by the tweet. We represent each candidate entity by its textual content acquired from its corresponding Wikipedia page. As such, we rank the textual documents based on their relevance to the input tweet. The top ranked document would ideally belong to the target entity to be linked to the tweet. Given this work is among the first to address the problem of implicit entity recognition, our work in this paper operates under two main assumptions: 1) the high-level domain of the implicit entity is known a priori. For instance, for a tweet such as the earlier mentioned tweet about the Boyhood movie, it is known that the implicit entity referred to in this tweet is from the broad space of *Films*, and 2) that the tweet that is being processed does in fact talk about an implicit entity. In other words, it is assumed that the tweet will relate to one implicit entity, which is a limitation of both our and the baseline approach.

More concretely, the contributions and novelties of our work can be enumerated as follows:

(1) We present a gold standard for benchmarking implicit entity linking techniques. We collect, manually label, and make public this gold standard dataset which consists of 7,870 tweets;
(2) We formalize and establish that implicit entity linking of tweets can be viewed as an ad-hoc document retrieval process whereby tweets are linked to documents as opposed to being directly linked to knowledge base entries;
(3) We show how state-of-the-art ad-hoc retrieval methods that rely on term dependency as well as explicit entity mentions can be used within the process of implicit entity linking;

(4) We propose a novel neural embedding-based feature that is incorporated into a Markov Random Field retrieval framework to augment term dependency and entity features to enhance implicit entity linking performance; and
(5) We compare our work on our gold standard dataset with the state-of-the-art and while showing improved performance; discuss the impact of various feature types on the implicit entity linking process. To this aim, we implement Perera et al. (2016)'s model to measure its performance on our gold standard.

The rest of this paper is organized as follows. Next section covers related work on NERC and entity linking as well as implicit entity linking for tweets. The details of our gold standard dataset is presented in Section 3. Our proposed approach is elaborated on in Section 4. We provide the details of our evaluations and our experimental findings in Section 5 and the paper is concluded in Section 6.

2. Related Work

The relevant literature to our work fall into four categories. First, we review work that introduce NERC for non-Twitter content types. These studies are specifically significant as they provide insight into how gold standard datasets should be developed for NERC. Additionally, we draw upon studies which focus on NERC for tweets and more specifically on different datasets that have been developed for this task. As the second category of related work, we review named entity linking in tweets. Most relevant to our work, the third category of related work encompasses those which focus on the identification of implicit entities. Last but not least, we review the entity embedding work that has recently been published. A summary of studies that will be reviewed in the following sections is presented in Table 1.

2.1. Traditional Named Entity Recognition and Classification

One of the relevant areas to our work addresses named entity hierarchies that have been developed and evolved to form the basis for gold standard dataset development and annotation for the task of NERC. A named entity hierarchy acts as a taxonomy of classes for different categories of named entities. Such hierarchies were initially developed with simple and shallow hierarchies including as few as 7 types of entities, namely organization, location, person, date, time, money and percent expressions (Grishman and Sundheim, 1996). In later studies, more specific entity types such as products and books (Satoshi and Hitoshi, 2000) were also introduced. One of the earliest comprehensive named entity hierarchies was developed in the work by Sekine and Nobata (2004). These authors proposed an *extended named entity hierarchy* containing 200 categories. Categories are rendered in a hierarchy containing parent and child nodes and 130,000 instances of each category are manually labelled. They also developed a named entity tagger using the labelled dataset and a dictionary of common noun phrases containing 50,000 entries, which were all manually labelled.

There have been attempts to benchmark NERC for both Twitter content and other content types. We draw upon CoNLL 2003 (Tjong Kim Sang and De Meulder, 2003) and NLPBA (Song et al., 2004) as examples on non-Twitter type content. CoNLL 2003 defines named entity as a phrase that contains a name of one of the three types of person, organization, and location. The task of CoNLL 2003 is to recognize named entities independent of language, i.e., for both English and German. A dataset containing Newswire content from Reuters and annotated with four types of entities, i.e., person (PER), location (LOC), organization (ORG), and miscellaneous (MISC) is provided. The dataset is comprised of three

Table 1

Summary of related work.

Study	Proposing Taxonomy	Providing Dataset		Perform NER, Classification, and/or Linking	Support for Implicit Entity Linking	Exploitation of Contextual Knowledge	Joint NERC	Exploitation of Neural Network Methods	Entity Embedding
		Microblog Content	Other Content						
(Grishman and Sundheim, 1996), Satoshi and Hitoshi (2000), Rizzo et al. (2017)	✓	-	✓	✓	-	-	-	-	-
(Tjong Kim Sang and De Meulder, 2003), (Tjong Kim Sang and De Meulder, 2003), (Kim et al., 2004)	-	-	✓	✓	-	-	-	-	-
(Ritter et al., 2011), (Cano Basave et al., 2013)	-	✓	-	✓	-	-	-	-	-
(Finin et al., 2010), Derczynski et al. (2015a)	-	✓	-	-	-	-	-	-	-
Liu et al. (2013), Shen et al. (2013), Hua et al. (2015), Fang and Chang (2014), Ibrahim et al. (2014), Feng et al. (2018),	-	-	-	✓	-	✓	-	-	-
Guo et al. (2013), Chang et al. (2014), ter Horst et al. (2017)	-	-	-	✓	-	-	✓	-	-
Waitelonis and Sack (2016), Torres-Tramón et al. (2016), Greenfield et al. (2016)	-	-	-	✓	-	-	-	-	-
Yamada et al. (2018), Li et al. (2016)	-	-	-	-	-	-	-	-	✓
Yamada et al. (2017)	-	-	-	✓	-	-	-	-	✓
Perera et al. (2016),	-	✓	-	✓	✓	✓	-	-	-
Our approach	✓	✓	-	✓	✓	✓	-	✓	✓

sets of pre-processed and tagged documents consisting of training, development, and test sets and one large set of unprocessed and unannotated data. The task of NERC has been performed for other text genres as medical and biological text, such as JNLPBA task (Kim et al., 2004). NLPBA is a dataset introduced at the aforementioned task for biomedical NER. This dataset is comprised of PubMed abstracts annotated with such named entity types as protein, DNA, RNA, cell line, and cell type. More recently, Jovanović and Bagheri (2017) investigate the annotation of biomedical unstructured textual content. Their work focuses on reviewing applications of annotation of entity mentions in biomedical texts. According to authors, semantic annotation of biomedical texts can facilitate the task of information extraction; therefore, they review the applications and benefits of semantic annotators.

Furthermore, we review work that attempt at providing benchmarking datasets for NERC on tweets. Despite abundance of Twitter data and information-richness of this form of user-generated content, there are not many available gold standard datasets for this task on tweets. Ritter et al. (2011) developed a dataset containing 2,400 tweets tagged with 10 tags *prevalent in Freebase and popular on Twitter* for evaluation purposes: person, geo-location, company, product, facility, TV-show, movie, sportstream, band, and others. Relying on crowdsourcing for annotation, Finin et al. (2010) developed a gold standard dataset containing 441 tweets with a total of 7,037 tokens. Another major dataset was developed by Cano Basave et al. (2013) for the purpose of *Concept Extraction Challenge*. The dataset contains a labelled set as the training set and an unlabelled one as the test set, with a total of 4,341 tweets. Entity types used in this dataset are person, organization, location, and miscellaneous. Derczynski et al. (2015a), while examining the performance of a number of state-of-the-art named entity recognition and linking systems, provide a new gold standard dataset for the task of entity linking. They systematically explore the challenges of NER and linking for microblogs through error analysis. Additionally, they examine problems that need to be tackled in order to improve the aforementioned text for microblog genre.

2.2. *Named Entity Linking in Tweets*

Entity linking in tweets has gained more attention in the recent years as the number of active Twitter users has increased and the accumulated social data has been successfully used in many upstream application domains (Liu et al., 2013; Rizzo et al., 2017; Edouard et al., 2017). Most Twitter entity linking work focus on *mention-level* entity linking, rather than whole tweet entity linking, and as such identify a word or a sub-sequence of words that can be linked to a knowledge base entry. Derczynski et al. (2015b) investigate the tasks of named entity recognition (NER) and named entity linking (NEL) for tweets and report on the state-of-the-art methods and systems developed for those tasks, as well as their main challenges and sources of error. They also provide a named entity disambiguation dataset for Twitter. The authors perform a comparative study on several NER systems over full-text documents and a NER system for Twitter content. They found that all studied systems perform poorly on tweets compared to their performances on full-text documents. The authors also show that NEL for tweets is a difficult task, especially because NER for tweets, which is the first phase for NEL, does not perform well enough to feed reliable recognized entities into the disambiguation phase.

Some researchers have already leveraged social, temporal and spatial clues present in tweets for the sake of the entity linking process. Liu et al. (2013) propose a model to link a set of mentions, rather than one single mention, to their corresponding entries in a knowledge base at the same time. To this end, the authors introduce a mention-mention similarity measure and integrate it with entity-entity and entity-mention similarities in order to comprise a collective inference method. One major advantage of this method is that it boosts performance on out of vocabulary mentions. Shen et al. (2013) exploit inter-tweet information and intra-tweet local information in their entity linking framework. The local information is extracted using a set of features, which include *context similarity* between the tweet and the knowledge base entry and *topical coherence* of the other mentions within the same tweet. They use inter-tweet information to compensate for lack of sufficient clues within a tweet and extract it using user interest modeling. Each user's interest is modelled by assigning each user an interest distribution over named entities mentioned in that person's tweets.

Hua et al. (2015) offer an approach based on modeling tweets' social and temporal contexts. The contextual information is formalized in three features, namely *entity popularity, entity recency,* and *user interest information*. The authors report that social interactions, especially the followee-follower relationship, are indicators of user interest to a large extent; therefore, a user's interest in a given entity is modelled as their interest in the community that is most related to that entity. The entity recency feature describes the freshness of an entity and is measured by assessing the extent to which a burst in the number of tweets related to that entity in a given short time happens. Fang and Chang (2014) propose a weakly supervised model for entity linking where they integrate and employ spatial and temporal information, so called *spatio-temporal* signals, for the first time and thus make considerable improvements. In their framework given a tweet and its timestamp and location of generation as input, the output is a set of entities present in the tweet. Ibrahim et al. (2014) also leverage temporal information in the form of entity temporal importance to reflect the time-varying topics of interest for people and trends. They extract the temporal importance of entities using the page view statistics of Wikipedia articles. This feature along with contextual enrichment and mention normalization enables the proposed framework to gain considerable performance improvements. Feng et al. (2018) propose a method in order to optimize the task of entity linking in tweets by narrowing down the candidate entities that need to be disambiguated. Based on their hypothesis, only a subset of candidate entities need to be considered for disambiguation in a tweet since there are a certain set of entities that are likely to be discussed by the users on Twitter. They achieve a better performance on accuracy as well as a reduced execution time for performing entity linking.

There are studies which adopt combinatorial approaches in order to integrate the two stages of named entity recognition and linking (Guo et al., 2013; Chang et al., 2014; ter Horst et al., 2017). Guo et al. (2013) propose an end-to-end entity linking system based on a structural SVM, which jointly optimizes both phases of entity linking, i.e., mention detection and disambiguation. The authors argue that the bottleneck in tweet entity linking is mention detection. Chang et al. (2014) also introduce an end-to-end entity linking system designed for microblog texts and text messages. Similar to Guo et al. (2013), the authors also perform entity linking by jointly optimizing entity recognition and disambiguation phases. ter Horst et al. (2017) also perform the task of entity linking in technical domains and test their performance on recognizing and linking disease names. This paper is relevant to our work specifically as it employs *undirected probabilistic graphical models* for modelling the probability distribution for the span of entity mentions and their corresponding knowledge base entries. Rizzo et al. (2017) introduced the NEEL dataset, which is very similar to our dataset in terms of the taxonomy they developed. Their taxonomy is comprised of 7 coarse-grained categories which is derived from the NERD Ontology. Although our taxonomy is based on DBpedia, their dataset taxonomy shares many similarities with our dataset including Person, Organization, Location, Event, and Product. Their taxonomy contains two additional categories, namely Character and Thing, while our dataset contains the Work category with two main fine-grained classes, i.e., WrittenWork and Film. These two important classes are included as fine-grained categories under the coarse-grained class of Product in their taxonomy. In 2016, the NEEL challenge had 5 entries among which (Waitelonis and Sack, 2016), (Torres-Tramón et al., 2016), and (Greenfield et al., 2016) were the best performing systems. The work by Waitelonis and Sack (2016) adopted an already developed natural language processing tool for Named Entity Disambiguation called KEA to account for microblog post special characteristics. Their method involves mapping tokens to DBpedia-based gazetteer and disambiguation of the matching tokens through scorers that are based on features to be extracted from the text. The system described by Torres-Tramón et al. (2016), known as Kanopy4Tweets, ranked second in the challenge. The work adopted Kanopy which is a topic disambiguation system relying on graph-based methods for microblog post processing. The work by Greenfield et al. (2016) ranked third in NEEL 2016. This work concentrated on the candidate selection phase by preparing a manual ontology mapping as well as attempting to capture the abbreviation trends used by tweet authors in order to achieve a high recall. For entity linking, they extract a total of seven features, for instance 'commonness', with the help of which they train a random forest classifier.

2.3. Implicit Named Entity Linking in Tweets

More Recently, Perera et al. (2016) introduced the notion of *implicit entities* in tweets, which will serve as the *baseline* for our experiments in this paper. The authors define the implicit entity linking problem as finding the most relevant descriptive knowledge base entity for a given tweet that encapsulates the essence and core topic of the tweet, which has not been explicitly mentioned in the tweet. They prepare and publicly share a dataset of tweets containing implicit entities in two domains, namely *Movies* and *Books*. The authors leverage contextual and factual knowledge from the knowledge graph in order to solve the problem of implicit entity linking and base their graph-based model heavily on contextual knowledge derived from pooled tweets with temporal affinity.

More specifically and in their work, for each input tweet containing an implicit entity of a known domain, a graph denoted as *entity model network* is built based on explicitly observed entities and the relationships between those entities derived from DBpedia's triple relations. Furthermore, the graph is complemented by knowledge acquired from one thousand tweets posted closest to the time of the tweet of interest referred to as *tweet clues*. The tweet clues are exploited in order to generate unigrams and weighted phrases to be used in the entity model in case they exist as Wikipedia anchor texts or page titles. Finally, implicit entity linking is done in two steps: 1) candidate selection, and 2) candidate ranking. The initial candidate set in the candidate selection phase includes those entities which have at least one edge with matching clue nodes and tweet clues in the graph. The top-k entities with the highest relevance are selected to be passed onto the ranking phase. The candidate ranking phase is done as a learning to rank task with an SVM$^{\text{rank}}$ model using a pairwise approach (the input includes pairs of candidate entities). To the best of our knowledge, these authors are the first to address the problem of *implicit entity linking*. Our work builds on and compares with this work based on the gold standard dataset.

2.4. Entity Representation

Recently, neural networks have been used extensively in order to embed knowledge base entities. In this section, we review a few studies, among many, in this realm since we have proposed an entity representation model in order to capture entity similarity (Yamada et al., 2018; Li et al., 2016; Yamada et al., 2017; Bianchi et al., 2018). The work by Yamada et al. (2018) trains a neural network to learn entity embeddings given a document. For each entity, they use its document words as well as the entities it contains to output the target entity. In another attempt, Yamada et al. (2017) propose a method to jointly learn embeddings of knowledge base entities. The proposed model predicts an entity given a document. They use their model in order to perform the task of entity linking and achieve state-of-the-art performance. The work by Li et al. (2016) uses the knowledge base and its hierarchical structure in order to learn entity representations. Their model also learns the category representations along with entity representations exploiting the knowledge base category information. We are specifically interested in the work by Yamada et al. (2018) and Li et al. (2016) because we have used their entity embeddings as baselines in our experiments for comparative analysis.

3. Gold Standard Datasets

As discussed in the related work section, datasets for the task of NERC have traditionally been collected based on a taxonomy, often based on a two-level taxonomy. Additionally, a typical dataset for named entity recognition and classification would be comprised of two major categories of tweets, namely tweets with and without explicit mention/s of entities, which could optionally include or not include explicit entities. While complying with the aforementioned characteristics, a dataset for the task of implicit entity linking should additionally comprise a third category containing tweets with implicit mention/s of entities. One gold standard dataset for this task has been collected and made public by Perera et al. (2016), which will be described in the following. To the best of our knowledge, the collected gold standard dataset in our work is both larger and more comprehensive than the existing datasets in the literature and incorporates all the tweets from the work by Perera et al. (2016). The statistics regarding our dataset are presented in Tables 2 and 3.

Table 2
Statistics for our gold standard dataset.

Type	**Implicit**	**Explicit**	**No Entity (NE)**
Count	1,345	2,483	3,842
Average explicit entity per tweet	2.53	2.68	0
Average token per tweet	26.16	21.96	16.60

Table 3
Proposed taxonomy for our gold standard dataset.

Coarse-grained Class		**Fine-grained Class**	**Frequency Implicit/Explicit**	
Person	a	Artist (Actor, Comedian, Rapper, MusicalArtist, MusicalArtist, Director, ...)	91	163
	b	Athlete	18	78
	c	Businessperson (Leader, CEO, Founder, Co-founder, Entrepreneur, Executive, ...)	74	144
	d	Celebrity (Model)	4	53
	e	Politician (PrimeMinister, VicePresident, ...)	84	200
	f	Scientist	10	21
	g	Writer (Author)	70	136
Organization	a	Company	82	58
	b	EducationalInstitution	1	13
	c	Group (MusicGroup, Band, PoliticalParty, Charity)	85	156
	d	SportsClub (SoccerClub, BaseballTeam, HockeyTeam, BasketballTeam, CricketTeam)	105	210
Location	a	Monument/HistoricalPlace	14	29
	b	PopulatedPlace (Country, city, ...)	174	338
	c	Building/Tower	10	20
	d	ArchitecturalStructure (Skyscraper,ReligiousBuilding, ...)	50	95
Event	a	NaturalEvent (Earthquake, Cyclone, ...)	10	19
	b	SocietalEvent (Awards, Festival, FilmFestival, SocietalEvent, SoccerTournament, SportsEvent, ReligiousEvent, ...)	87	159
Product/Device	a	MobilePhone/CellularTelephone	36	68
	b	Instrument/MusicalInstrument	8	2
	c	Software	5	5
Work	a	WrittenWork (Book)	158	261
	b	Film	169	255

3.1. Perera et al.'s Dataset

The first gold standard dataset for the task of implicit entity linking was provided by Perera et al. (2016)[1]. This dataset, which is limited to two domains, namely, Books and Movies, consists of 207 tweets with an average length of 18 words for 54 distinct entities in the domain of Movies and 190 tweets with an average length of 18.5 words for 53 distinct entities in the domain of Books. The dataset was created by first retrieving tweets in the month of August 2014 containing the keyword 'book' or 'novel' for the domain of Books and 'film' or 'movie' for the domain of Movies. Then, the tweets were manually annotated by two annotators as one of the following: 'explicit' for tweets which contain an explicit mention, 'implicit' for tweets which contain an implicit mention, or 'NE' for tweets which contain neither an explicit mention nor an implicit mention. Finally, only tweets which received similar annotations and contained an implicit entity by both human judges were entered into the evaluation dataset. In total, they annotated 209 and 190 tweets containing 'implicit entities' for the domain of Movies and Books, respectively.

3.2. Proposed Gold Standard Dataset

Inspired by traditional datasets developed for the task of NERC, we develop a two-level, i.e., fine-grained and coarse-grained, hierarchy for structuring and collecting our gold standard dataset. Our taxonomy contains 6 coarse-grained entity types, namely Person[2], Organization[3], Location[4], Product/Device[5], Event[6], and Work[7]. These elements and the fine-grained classes pertinent to each element are based on the DBpedia taxonomy. The proposed coarse grained taxonomy is meant to retain the elements (classes) of traditional named entity recognition and classification taxonomies. However, the fine-grained taxonomy was determined once the tweets for each coarse-grained class were collected and manually labeled with implicit entities. The identified implicit entities in the collected tweets led to the finer-grained classes in the taxonomy.

Our dataset contains three major categories of tweets: tweets containing implicit mentions of entities, which we denote as *Implicit* tweets; tweets containing explicit mentions of entities, denoted as *Explicit* tweets; and tweets containing neither of the two mention types, denoted as *No Entity (NE)*. For the tweets in the first category that do in fact refer to an implicit entity, we have additionally run each tweet through TagMe and extracted explicit entities for these tweets where available. Therefore, each implicit tweet is labeled with its corresponding implicit entity as well as the list of explicit entities that were observed in that tweet. The purpose for this is to provide the list of available explicit entities to other researchers for replication purposes. The core of our experiments in this paper is based on the implicit tweets and we do not use the explicit and NE tweets in our experiments. However, we do provide explicit and NE tweets in our datasets, so that researchers can investigate the possibility of discerning between the three tweet types, namely implicit, explicit and NE tweets, as future work.

---

[1] https://github.com/sujanucsc/IEL-Twitter/wiki
[2] http://mappings.dbpedia.org/server/ontology/classes/Person
[3] http://mappings.dbpedia.org/server/ontology/classes/Organisation
[4] http://mappings.dbpedia.org/server/ontology/classes/Place
[5] http://mappings.dbpedia.org/server/ontology/classes/Device
[6] http://mappings.dbpedia.org/server/ontology/classes/Event
[7] http://mappings.dbpedia.org/server/ontology/classes/Work

Table 4
Implicit tweet examples.

Domain	**Implicit tweet example**	**Target entity**
Person	*"Imo Angelina owes Jennifer an apology for stealing her husband. Then again he wasn't that great a prize so these women just need to let it go and be friendly-ish."*	Brad Pitt
Organization	*"You're a real gentleman. You tagged Zayn too even he left the band. We want you to come back with your 4 mates. That's what we want. Plz do this favor for your fans. We're still waiting. We love our 5 boys."*	One Direction
Location	*"Hockey Museum. CN tower. Eaton Centre. Skating at the City Hall. Its a remarkable city."*	Toronto
Event	"So that night as I watched over the baby, I saw Jenny whose house was lit up across the street at her window. Kids all over were excited because this was the night that Santa Claus comes to visit us bringing gifts and toys. We want toys."	Christmas
Product/ Device	*"Ok, so Siri rarely works correctly, Apple eliminated the ability to use a headset and charge your phone..."*	iPhone
WrittenWork	*"I can relate more to Jay Gatsby than any other book character."*	The Great Gatsby
Film	*"I wish Richard Linklater spent 12 years making a movie about my cat."*	Boyhood

Table 5
Explicit tweet examples.

Domain	**Explicit tweet example**	**Target entity**
Person	*"If a guy ever sang Perfect by Ed Sheeran to me I'd probs propose to him"*	Ed Sheeran
Organization	*"So uh, I'm considering applying for a summer school course, at Yonsei University, in Seoul. Uhhhhhhh"*	Yonsei University
Location	*"The International Space Station is passing over the United Kingdom at January 17, 2018 at 07:03PM, for 642 seconds."*	United Kingdom
Event	"I must admit, we went once on Boxing Day...never again, too cold and standing room only!"	Boxing Day
Product/ Device	*"Hi, do you still have OnePlus X in stock?"*	OnePlus X
WrittenWork	*"Still got the same feeling like 7/8 years ago after reading harry potter and philosopher's stone... uhhhh im so happy"*	Harry Potter and the Philosopher's Stone
Film	*"Blade Runner 2049 was so good. I need to see it again ASAP while it's still in some theaters."*	Blade Runner 2049

A sample set of tweets pertaining to the three categories is provided in Tables 4-6. Also, the statistics related to our dataset and its taxonomy are provided in Table 2 and Table 3, respectively. In order for the dataset to reflect the actual ratios between those three types, we performed repeated random sampling using the Twitter API and calculated the ratios between the three types of categories. In total, we evaluated four hundred tweets by three human judges and found the following ratios to hold: 35% Explicit, 15% Implicit, and 50% NE. Therefore, these ratios are respected in our gold standard as well.

In order to tag the dataset, we collected a large pool of tweets in a four-month time frame spanning from October 2017 to January 2018. We opted for a four-month time period to avoid entity drift (Masud et al., 2010). Afterwards, three human annotators (judges) tagged the tweets manually for all the three category types of our dataset, namely Implicit, Explicit, and NE. The tweets included in the gold standard received the same label by all three judges. For each tweet in the dataset, we provide its tweet ID and user ID, as well as category and the target entity labels.

Table 6
Sample tweets without either implicit or explicit entities (NE).

Domain	Sample tweet	Target entity
N/A	"We can't be friends if you don't know the difference between home friends and college friends. This isn't hate it's just that you can't go off for a semester and create a whole new friend group. Stop with the nonsense "	N/A
N/A	"haha yeah nice cool dude im proud of u wanna make a family"	N/A
N/A	"i would like to convey the basic idea that carp flies don't necessarily need to imitate something specific, but should move in a way to imitate life"	N/A

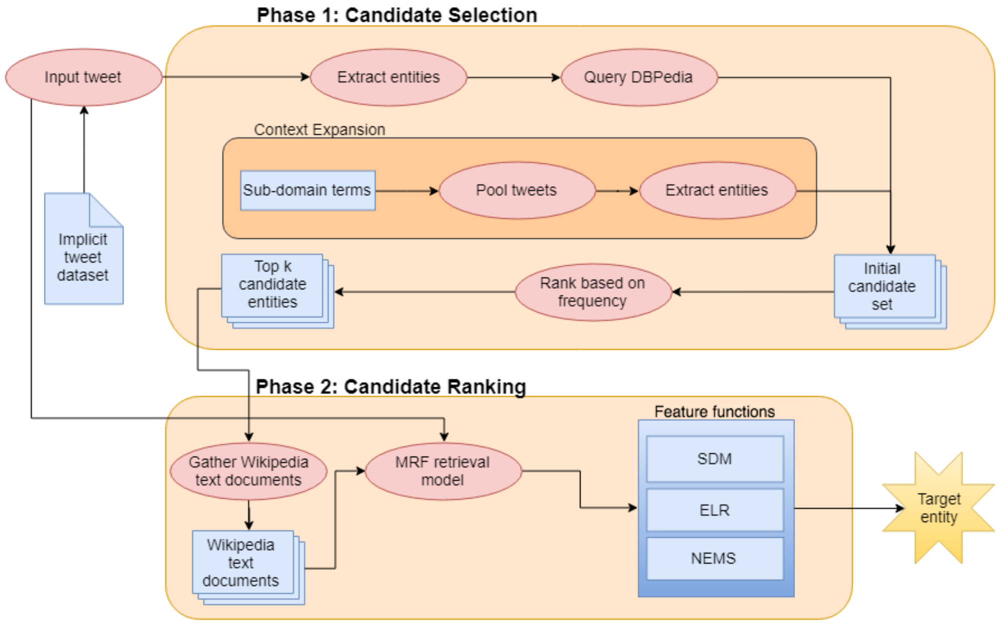

Fig. 1. An overview of the proposed approach for implicit entity linking.

We incorporate the 327[8] Implicit tweets collected by Perera et al. into our gold standard dataset under the Work class with two fine-grained classes of WrittenWork (encompassing Book domain tweets) and Film (containing Movie domain tweets). The statistics for our gold standard dataset, including 327 tweets from Perera et al. (2016) and our taxonomy are presented in Tables 2 and 3. We make our dataset publicly available to facilitate the reproducibility of our work[9].

4. Proposed Approach for Implicit Named Entity Linking

Our approach for implicit entity linking in tweets, as shown in Figure 1, is comprised of two main steps, namely (1) *candidate selection* and (2) *candidate ranking*. The knowledge graph-based method

[8]There is a slight discrepancy between what Perera et al. (2016) report in their paper and what is available in terms of number of tweets. The unavailability of some of the tweets is, among other things, due to change in level of access to some users' profiles on Twitter over the years.
[9]https://github.com/HawreH/Implicit-Entity-Recognition-and-Linking-in-Tweets-Resources-and-Dataset

for Candidate selection phase identifies and selects a set of entities that are (potentially) relevant to the tweet of interest. In the candidate ranking step, we then further process the selected candidate entities based on an ad-hoc retrieval model in order to rank them based on their relevance to the tweet. We have adopted the names and the two step pipeline from the work by Perera et al. However, the main distinguishing aspect of our work is that while Perera et al. are focused on the traversal of the knowledge graph for identifying implicit entities, we are interested in identifying implicit entities through an ad hoc retrieval strategy based on three different feature functions. In the following section of the paper, we start by formalizing the details of the candidate selection step.

4.1. Candidate Selection

The first step in our work is to narrow down the search space by selecting a set of candidate entities, which have a higher probability of being related to the tweet of interest. We make use of the DBpedia knowledge graph as the entity search space. Given a Tweet *twt*, and the type of the implicit entity that is being sought ϑ, e.g., `wikidata:Film`, the objective of the candidate selection method is to retrieve a set of entities from DBpedia relevant to *twt* and of type ϑ. To this end, we first extract the explicit entities that are present in the input tweet using a standard entity linking tool, e.g., TagME. Once the explicit entities are extracted, we query the DBpedia knowledge graph for all those triples whose subject (or object) match one of the identified explicit entities and the object (or subject) has `rdf:type` ϑ. All such retrieved entities of type ϑ are included in a candidate entity set. This method will identify a set of entities that are related to the explicit entities within the tweet and are of the specific type that we are looking for.

Due to the short length as well as the informal language of tweets, it is possible that we face the problem of *entity sparsity*. In other words, it is possible that only very few and in many cases only one entity is retrieved based on performing entity linking on a tweet and therefore, the search space for identifying candidates would be too narrow. To this end, we perform *context expansion* for candidate selection where a set of relevant tweets to the tweet of interest are pooled and added as context. Empirically speaking, for the tweets in our dataset, there were on average only 1.77 explicit entities related to each tweet, which was then increased to 19.02 after performing context expansion. We use Twitter API to find tweets with specific keywords during a closely related time interval and expand the initial tweet. In order to perform the search, we look for those tweets that include the surface form of the explicit entities observed in *twt* and also have a mention of the `dbp:label` for ϑ, e.g., `wikidata:Film` `dbp:label` Film. From among the retrieved tweets that match the search criteria, we only retain those that are within a two week time frame from *twt* and have at least one explicit entity when ran through an entity linking system. As a result of context expansion, *twt* now includes both the initial tweet as well as the content of the tweets that were retrieved in this process. From among all the retrieved entity, we choose the top K entities based on their frequency of appearance. There are cases where the target implicit entity is not linked due to the explicit entity linking tool performance or lack of RDF data. Major sources of error, including the aforementioned one, are elaborated in Section 5.3.

4.2. Candidate Ranking through Ad-hoc Retrieval

Having found the set of candidate entities for the input queries, the goal of the second step of our model is to rank them based on their relevance to the input query. This phase of our proposed model is comprised of two major steps. First, we formulate the problem of implicit entity linking as an ad-hoc retrieval model, where the input tweet is considered the query and the candidate entities are regarded as the documents. We make this possible through representing the entities in the candidate set by their textual representation. This is done by retrieving textual content found on each entity's Wikipedia page. On this basis, we perform document retrieval for each input tweet in a Markov Random Fields (MRF) framework and extract the feature values of MRF feature functions for each document in order to be used in the consequent step, i.e., Step 2. In the second step, we used the extracted features to train a learning to rank model. We adopted the ad hoc information retrieval strategy for ranking candidates, as it provides a systematic framework to define effective features for measuring the importance of candidates given a query. It also provides the means through *learning to rank* to effectively integrate the various feature functions and develop a final ranking mechanism. We define three feature functions within an markov random fields framework, namely *term dependency* features, *entity* features, and *neural embedding* features.

More specifically, in the first step of this phase, for each entity, we extract the textual content of the related Wikipedia page that can be used to perform the document retrieval. This has the advantage of making the tweet and the entity spaces comparable because they are now both composed of textual content. For instance and as we will show in the experiments section, for an implicit entity linking process where ϑ is `wikidata:Film`, we can benefit from the textual content available for the entity to form the entity representation space. Given the comparability of the two spaces, the problem of ranking the entities based on their relevance to tweet can be formulated as an ad-hoc retrieval process whereby the tweet is an input query and the Wikipedia textual content available for the entities form the document space.

In the following sections, we first draw upon the MRF framework for the task of ad hoc retrieval and we show how the set of feature functions we use, i.e., term dependency, explicit entity and neural embedding features, can be embedded in this framework. Afterwards, we explain how we train a learning to rank model with the feature function values in order to perform the task of ranking candidate entities.

4.2.1. Markov Random Fields

As an instance of undirected graphical models, Markov Random Fields (MRFs) are generally exploited for modelling joint distributions. Metzler and Croft (Metzler and Croft, 2005) were among the first to show MRFs are effective in modelling term dependencies within the ad-hoc retrieval task. In the context of this study, given a tweet *twt* and a document D, the objective is to compute the joint distribution $P(twt, D)$. In MRF, random variables are used as nodes to build a graph G, i.e., tweet terms twt_i and a document D, and the edges determine the dependence between them. The joint probability between pairs of random variables *twt* and D is computed as:

$$P_\Lambda(twt, D) = \frac{1}{Z_\Lambda} \prod_{c \in C(G)} \phi(c; \Lambda) \tag{1}$$

where D is a document, $C(G)$ is the set of cliques in G, and $\phi(c;\Lambda)$ is a non-negative function. A clique (c) is a subset of vertices in an undirected graph where every two vertices are adjacent. As seen in Figure 2, each of the tweet terms "Ethan", "Hawke", "Patricia", and "Arquette" form cliques with the document as they are adjacent to the document. The bigrams "Ethan hawke" and "Patricia Arquette" also form cliques with the document where any two nodes are adjacent. Examples of unordered bigrams are "Hawke Arquette" and "Hawke Patricia", just to name two. As seen in the graph, "Hawke Arquette"

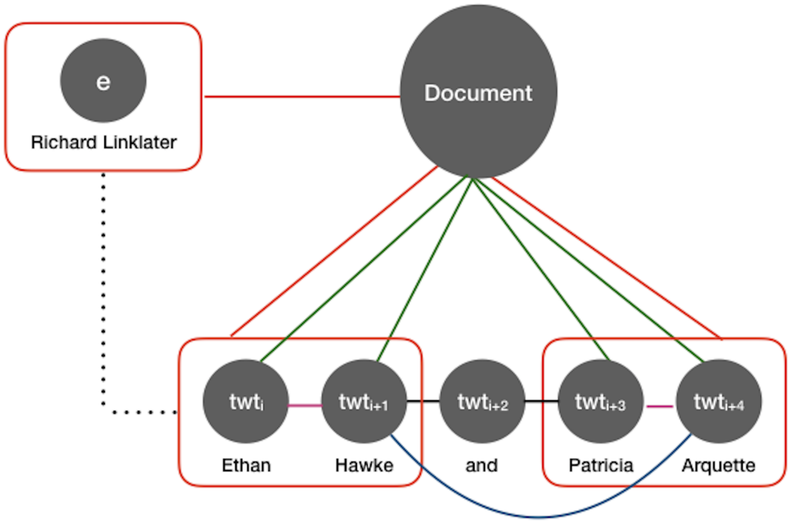

Fig. 2. A graph-based demonstration of the cliques formed between a tweet and a document based on different feature functions.

forms a clique with the document where each two nodes are adjacent; this is not true in the case of the first example as the nodes for "Hawke" and "Patricia" are not adjacent. The feature function, denoted by $f(c)$, and Λ are used to parameterize the function. After ruling out the normalization factor Z_Λ due to its computational cost, the ranking equation is simplified as:

$$P(D|twt) \stackrel{rank}{=} \sum_{c \in C(G)} \lambda f(c) \tag{2}$$

Potential feature functions can take different information into account. A variety of retrieval tasks have exploited various feature functions in their models (Metzler and Croft, 2007, 2005; Ensan and Bagheri, 2017). We are specifically interested in the *Sequential Dependence Model (SDM)* and *Entity Linking incorporated Retrieval (ELR)* (Hasibi et al., 2016) since they encompass information on term dependency and explicit entity mentions in their feature functions, respectively. We further propose a third type of feature function based on neural embeddings, called *Neural Embedding-based Measure of Similarity (NEMS)* in our framework.

For the sake of explanation, we introduce a running example in Figure 3. The tweet in this example will be used to show how the information in the feature function will be derived.

4.2.2. SDM Features

As an effective and efficient MRF-based retrieval model, the Sequential Dependence Model (SDM) assumes that tweet terms are sequentially dependent on each other. Following that assumption, tweet terms which are adjacent or in close proximity are connected to each other in the MRF-based graph G. The cliques formed between tweet terms and a document give way to three feature functions based on: (1) cliques formed by a tweet unigram (term) and a document node, and (2) cliques involving a node containing two or more ordered terms of a tweet and a document, and (3) cliques involving a node containing two or more unordered terms of a tweet and a document.

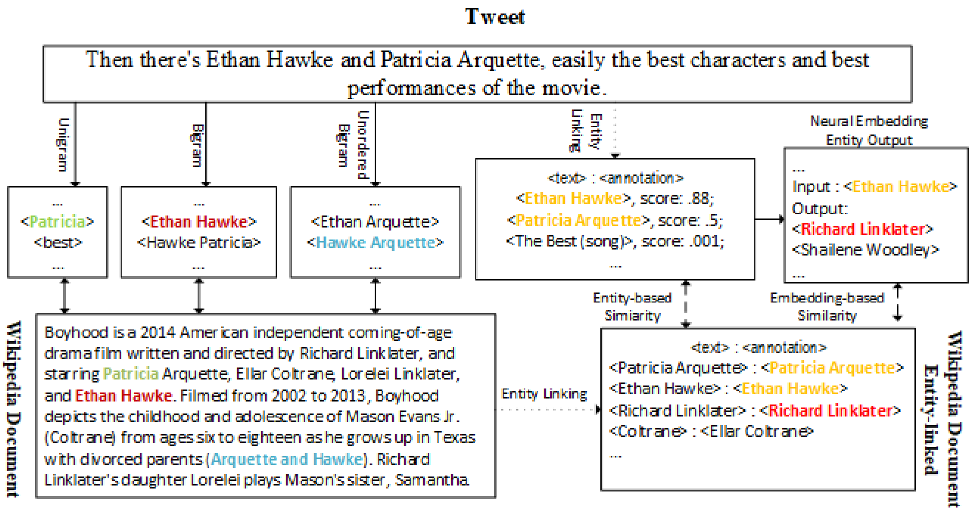

Fig. 3. A demonstration of the information extracted from a tweet and a Wikipedia page for the calculation of the feature functions.

The feature function for the first term-based clique type is formulated as follows:

$$f_t(twt_i, D) = log\left[(1-\alpha_D)\frac{tf_{twt_i,D}}{|D|} + \alpha_D\frac{cf_{twt_i}}{|C|}\right] \tag{3}$$

where twt_i is a tweet term, $tf_{twt_i,D}$ is the frequency of tweet term twt_i in document D, $|D|$ is the length of document D, cf_{twt_i} is the frequency of tweet term twt_i in the entire collection of documents, $|C|$ is the size of the collection of documents, and α_D is a smoothing parameter. It is worth mentioning that the collection of documents is comprised of the set of documents retrieved based on the candidate selection step for each query. In the example in Figure 3, 'Patricia' is a unigram that is shared between the tweet and the document.

The other two feature functions are designed based on the second clique type involving contiguous terms called *Ordered Phrase* feature function and non-contiguous tweet terms, which are in a close proximity referred to as the *Unordered Phrase* feature function. The formulations for ordered and unordered feature functions are as follows:

$$f_O(twt_i, ..., twt_{i+k}, D) = log\left[(1-\alpha_D)\frac{tf_{\#1(twt_i,...,twt_{i+k}),D}}{|D|} + \alpha_D\frac{cf_{\#1(twt_i,...,twt_{i+k})}}{|C|}\right] \tag{4}$$

$$f_U(twt_i, ..., twt_j, D) = log\left[(1-\alpha_D)\frac{tf_{\#uwN(twt_i,...,twt_j),D}}{|D|} + \alpha_D\frac{cf_{\#uwN(twt_i,...,twt_j)}}{|C|}\right] \tag{5}$$

where $tf_{\#1(twt_i,...,twt_{i+k}),D}$ denotes the frequency of an ordered tweet phrase in document D and $tf_{\#uwN(twt_i,...,twt_j),D}$ denotes the frequency of an unordered tweet phrase formed by terms in a window

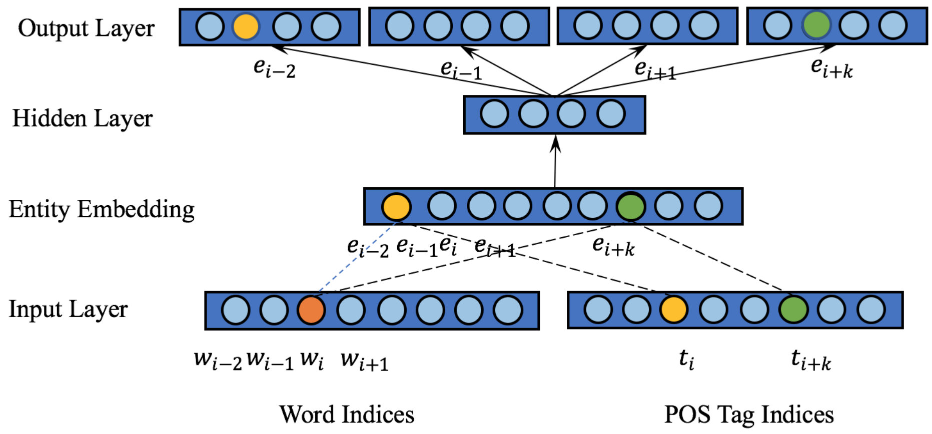

Fig. 4. The neural architecture employed for learning entity embeddings.

of size N in the document. In the running example of Figure 3, 'Ethan Hawke' is a shared ordered bigram that is observed both in the tweet and the document while 'Hawke Arquette' is an unordered bigram that is shared between the tweet and document spaces.

4.2.3. ELR Feature

Explicit entities, as mentioned previously, are key to identifying candidate entities from the input tweet in our work. Therefore, we additionally exploit a feature function based on explicitly observed entities in the form of cliques involving the explicit entity mentions of the tweet and the document. The feature function is inspired by the Entity Linking incorporated Retrieval (ELR) framework (Hasibi et al., 2016). This feature function, denoted as $f_E(e, D)$, is formulated as:

$$f_E(e, D) = log\left[(1 - \alpha_D)tf_{\{0,1\}(e,D)} + \alpha_D\frac{tf_{e,D}}{|C_e|}\right] \tag{6}$$

where $tf_{\{0,1\}(e,D)}$ refers to existence of the entity in the document, $tf_{e,D}$ represents the entity frequency in the document, and $|C_e|$ denotes the total occurrence of the entity in the collection of documents. We integrate the confidence score of the entity tagged by the linker, denoted as $s(e)$ into the probability equation for this clique type to obtain the feature function in the following form:

$$P(D|t) = \sum_{c \in C(G)} \lambda_E s(c) f_E(c, D) \tag{7}$$

In the running example of Figure 3, `dbr:Ethan_Hawke` and `dbr:Patricia_Arquette` are extracted by the entity linking system for both the tweet and the document and hence are used in ELR for calculating the feature function.

4.2.4. Neural Embedding-based Measure of Similarity Feature

A limitation of the ELR feature function is that in case an entity in *twt*, or other reachable entities from *twt*, are not present in D, the value of the feature function would be zero. However, a practically effective feature function has to be able to account for situations where while an entity c_1 in *twt* does not occur in

D, some closely related entity c_2 is mentioned in *twt*. In the running example, the tweet mentions ‘Ethan Hawke’ and ‘Patricia Arquette’; therefore, the ELR feature function would only retrieve documents that contain these two entities or proximal reachable entities in the knowledge graph. For other documents where ‘Richard Linklater’ is mentioned, this feature function considers them unrelated although ‘Ethan Hawke’ and ‘Richard Linklater’ are an actor-director duo who have worked together in eight films.

To relieve the effect of this limitation, we propose a neural embedding-based feature function in order to take the relationship between entities into consideration even when they are not observed in the tweet or documents. The *neural embedding learning* model’s objective is to learn vector representations that are useful for predicting entity relevance. Different from the vanilla word2vec models where word neighborhood is used for learning word similarity, we formulate our neural embeddings to learn the similarity at phrase and entity levels. As referenced in the literature review section of this paper, a variety of approaches have been proposed for learning entity representations. For instance, document context, entity hierarchical relations, and joint embedding of words and entities are among the common approaches for learning entity embeddings. Distinct from existing work, in our work, we use three sources of information for learning entity representations as shown in Figure 4: 1) the context of entities observed in the document collection where the context is the nearby entities and words depending on whether or not the neighbors are named entities, 2) the role of an entity’s surface form derived through part of speech tagging using traditional NLP framework, and 3) the relationship between entities that are within similar contexts. This allows us to learn entity representations that are cognisant of word and document context, entity interaction and the grammatical role of entities in their context. Such a combination of information for learning entity embeddings is novel.

Before building our embedding model, we use Honnibal and Johnson (2015)’s Spacy NLP framework to extract named entities from sentences. Formally, a sentence with a list of words $w_1, w_2, ..., w_n$ and their corresponding part-of-speech tags $t_1, t_2, ..., t_n$, can be related to a list of entities $e_1, e_2, ..., e_m$ where $e_i = [(w_k, t_k), ..., (w_{k+l-1}, t_{k+l-1})]$ is an entity with a surface form of length $l \geqslant 1$. If $l = 1$, then w_k is not an entity or it is a single word entity. Therefore, e_i can be an entity or a word in the sentence. It means that if we find a named entity, then we replace the corresponding words by the entity. Otherwise, we will use the words themselves to train the embedding model. The relation between $w_1, \ldots, w_n$, $t_1, \ldots, t_n$, and entities $e_1, \ldots, e_m$ can be shown as follows: 1) $e_i = (w_k, t_k)$ if w_k is not an entity; 2) $e_i = [(w_k, t_k), ..., (w_{k+l-1}, t_{k+l-1})]$ if words w_k to w_{k+l-1} are in the span of an entity e_i. So we always have $m \leqslant n$. In the special case, $m = n$ when there are no entities in the current sentence.

In addition, each entity or word e_i is represented by a one-hot encoding vector linked to the input words and their corresponding POS tags. Hence, the entity vectors are one-hot encoded in the hidden layer. As entity vector space is high dimensional, we map the entity vectors to lower dimensional space in the hidden layer. Our objective is to maximize the log probability as follows:

$$\frac{1}{m}\sum_{i}^{m}\sum_{(j\in S)\cap(j\neq 0)} log\ p(e_{i+j}|e_j) \tag{8}$$

where m is the total number of explicit entities, $log\ p(e_{i+j}|e_j)$ is the skip-gram softmax function and S is the context window. As mentioned earlier, each entity is assigned a unique identifier and encoded in a sparse vector space using one-hot encoding technique. As shown in Figure 4, the distinguishing aspect of our neural embedding model is that it jointly considers word neighborhood, part-of-speech tag information and entity relations in the neural embedding, which has not been attempted to build entity embeddings in the past.

With a trained neural embedding model, we expand the tweet by searching for similar entities in the model and selecting top-k entities to retrieve relevant documents. Formally, we define a novel embedding-based feature function as follows. Let $v_1, v_2, ..., v_m$ be the vector representations of entities $e_1, e_2, ..., e_m$, we have the kernel matrix of these entities:

$$M(v_i, v_j) = v_i \cdot v_j \tag{9}$$

For each entity e_i, we find a list of most relevant entities:

$$E(v_i, k) = \{e_i\} \cup \{e_j | argmax_{j \neq i}\, (M(v_i, v_j), k)\} \tag{10}$$

Based on the list of most relevant entities, the Neural Embedding-based Measure of Similarity (NEMS) feature function is defined as:

$$f_N(e, D) = \sum_{c \in E(e,k)} f_E(c, D) \tag{11}$$

where f_N is the feature function based on the two-cliques involving most relevant entities to the explicit entities of a tweet and the document D and F_E is from ELR. In Figure 3, while there are two exact entity matches between the tweet and the document, another entity `dbr:Richard_Linklater` is also present in the document, which is semantically similar to the other entities in the tweet. This similarity is captured in our embedding of entities and employed in our feature function.

4.3. Learning to Rank

In order to address the ranking problem, we exploited the SVM$^{\text{rank}}$ model trained with the aforementioned features. The choice of SVM$^{\text{rank}}$ is motivated by two factors; first, it would replicate our baseline's experimental set-up, hence leading to a fairer comparison. Second, SVM$^{\text{rank}}$ has proved to perform well in ranking problems similar to ours. The specifications of our model, identical to the baseline, are as follows: linear kernel, 0.01 as the trade-off between training error and margin, and the loss function is the number of swapped pairs summed over all queries.

5. Experiments

In order to evaluate our work, we adopt the evaluation methodology proposed by Perera et al. (2016), the gold standard dataset provided in the present work augmented by the existing dataset, and evaluation metrics used in the aforementioned study as the main baseline. In the main baseline, authors have focused on only two domains, namely *books* and *movies*. The objective of the experiment was to identify the best implicit entity from DBpedia that relates to the tweet. As suggested in the baseline, the evaluations encompass the performance of the model's two steps, namely candidate selection and candidate ranking *individually* and *in tandem*. Since we needed to examine the baseline's performance on our proposed gold standard dataset for comparison purposes, we implemented Perera et al. (2016)'s proposed method and evaluated it on our gold standard. It should be mentioned that Perera et al. designed their method when the character limit of Twitter was 140 but the current character limit is 280. As such, while the dataset used for comparing two methods is the same, the authors of Perera et al. (2016) may have

Table 7

Recall of candidate selection phase.

	Context Expansion	Person	Organization	Location	Event	Product/Device	WrittenWork	Film
Our approach	No	71.2	56.7	66.9	87.6	87.7	72	75
	Yes	**82.3**	**71.4**	**78.6**	**92.7**	96	**80**	**81**
Perera et al.	No	18	15	11	10	8	23	40
	Yes	80	69	78	90	**98**	77	79

intended their method to only be used with tweets of length 140. For this reason, in our experiments, we have compared our proposed approach with the baseline separately on datasets that have tweets that are at most 140 characters long and also otherwise. We have posted our code, annotations of tweets and documents on Github[10] for the sake of reproducibility. It is important to mention that when training NEMS, the hyperparameters were context window size S of 5, negative sample of 5 and an embedding dimension size of 128. In the following, we draw upon the experiments we have done for candidate selection as well candidate ranking and elaborate on the details and experiment environment setup for each phase. Additionally, we perform error analysis of the candidate ranking phase for tweets of each domain within our dataset. It should be noted that given the initial two assumptions of our work mentioned in the introduction, namely (1) the assumption that the processed tweets do in fact contain an implicit entity, (2) the high-level domain of the implicit entity be given as input to this process, our experiments in this section are focused on those tweets in our dataset that are within the implicit tweets category.

The technical terms referring to evaluation metrics used in this section are defined in the context of our work as follows. *Candidate selection recall* for a given domain is calculated as the number of tweets for which the target implicit entity has been retrieved as a candidate divided by the total number of tweets in that domain. *Precision@1* of candidate ranking, in simple terms, is calculated as the number of tweets for which the target entity has been correctly ranked first divided by the total of tweets in that domain. For more elaboration in the context of SVM^{rank} pairwise ranking setup, please refer to Section 4.2. *Accuracy* of the whole system, i.e., the combination of the two modules, is calculated as the product of candidate selection recall and candidate ranking precision.

5.1. Candidate Selection Evaluation

In the candidate selection phase, whose results are rendered in Table 7, we performed the steps for input tweets of each domain separately. Therefore, we calculate the candidate selection recall per domain. The candidate selection recall was calculated as the portion of tweets which had the target entity ranked within the top-k, with k=25, results. The value of k is motivated by two factors: first, the baseline has used k=25 for candidate selection; second, as seen in Figure 5, our experiments indicated that the candidate selection recall does not improve noticeably after k=25. We performed the selection of relevant entities without as well as through *context expansion* in order to evaluate the effect of contextual information.

The quality of the selected candidates determines the *recall* of our proposed approach. As seen in Table 7, our method outperforms the baseline on all but one of the categories. It is also observed that candidate

[10]https://github.com/HawreH/Implicit-Entity-Recognition-and-Linking-in-Tweets-Resources-and-Dataset

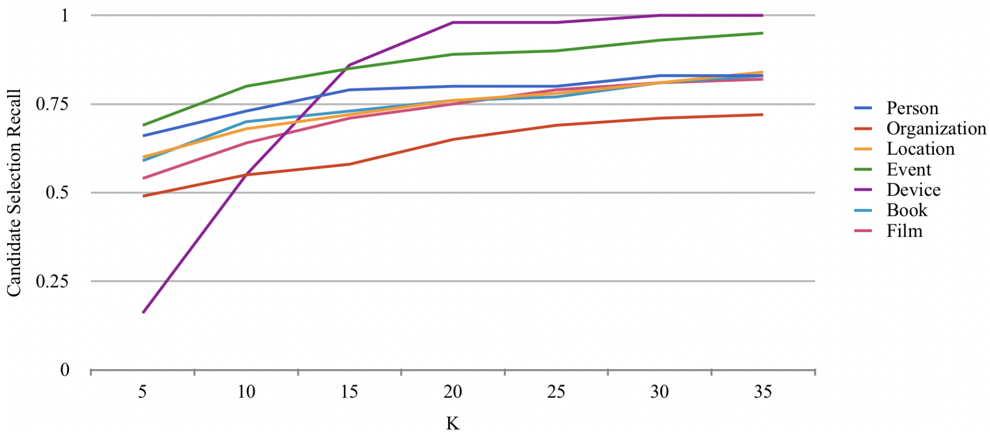

Fig. 5. An overview of the proposed approach for implicit entity linking.

selection has an improved performance when context expansion is applied. It is worth mentioning that for context expansion, we performed the pooling by gathering 1,000 tweets posted nearest to the time of the input tweet using terms associated with the sub-domain as required keywords. The keywords used for pooling relevant tweets from different sub-domains are as follows:

(1) Person: a) actor, actress, musical artist, artist, rapper, musical performer, comedian, cricketer, director, performer, musical performer; b) athlete, player, sportsman, sportswoman, Olympian; c) business person, businessman, billionaire, ceo, founder, co-founder, leader, executive, entrepreneur; d) celebrity, model; e) president, prime minister, vice president, king, queen, royalty; f) scientist, naturalist, physicist, mathematician, genius; and, g) writer, author;
(2) Organization: a) company, organization; b) university, college, educational institution; c) charity, band, music group, political party; and, d) basketball, hockey, soccer, cricket, sports;
(3) Location: a) historical place, historical building; b) city, country; c) building, tower, house, structure; and d) skyscraper, tall building, temple, mosque, church, religious building, architectural structure;
(4) Event: a) hurricane, earthquake, tsunami, natural event; and b) awards, ceremony, film festival, festival, holiday, religious event, societal event, tournament, sports tournament, sports event;
(5) Product/Device: a) smartphone, phone; b) instrument; and c) software, application, app, website;
(6) Work (WrittenWork): book, novel; and,
(7) Work (Film): film, movie.

These keywords were selected by the annotators who were performing the dataset curation task and consist of words that were agreed on by all the curators. It should be noted that the above enumerations (a,b,c,...) correspond to the fine-grained classes reported in Table 3.

For the extraction of explicit entities, we used an off-the-shelf standard tagging system, namely TagMe (Ferragina and Scaiella, 2012). We repeated the experiments using AIDA (Ibrahim et al., 2014), another standard entity tagger, in order to measure the impact of the tagger on the system's performance. Although the results using TagMe were found to outperform those using AIDA, the impact was not significantly different.

Table 8
The P@1 of the baseline ranking model using TF and TFIDF.

Domain	Person	Organization	Location	Event	Product/Device	WrittenWork	Film
TF	1.5	0.0	3.7	14.8	18.3	2.7	6.6
TFIDF	13.6	11.5	10.1	33.3	55.1	20.6	45.1

Table 9
The P@1 of the ranking models. In this table SDM, ELR and NEMS stand for Sequential Dependence Model, Entity Linking incorporated Retrieval and Neural Embedding-based Measure of Similarity, respectively.

Domain	Person	Organization	Location	Event	Product/Device	WrittenWork	Film
SDM	42.236	49.162	46.664	41.538	64.444	58.382	58.382
SDM+ELR	53.404	58.99	53.242	49.232	66.888	62.474	62.474
SDM+ELR+NEMS	**59.827**	**61.23**	**58.258**	**54.092**	**67.63**	**72.975**	**76.43**
Perera et al.	49.6	49	49.8	50.4	48.9	61.05	60.97

Table 10
The performance (P@1) of different embedding methods compared to NEMS.

Domain	Person	Organization	Location	Event	Product/Device	WrittenWork	Film
NEMS	**59.827**	**61.23**	**58.258**	**54.092**	67.63	**72.975**	**76.43**
Yamada et al.	52.688	55.516	56.866	49.232	64.666	70.38	75.296
Li et al.	57.362	56.184	56.808	46.156	**72.888**	61.142	75.296

Table 11
The accuracy of the implicit entity linking method compared to the baseline.

	Person	Organization	Location	Event	Product/Device	WrittenWork	Film
Our Approach (SDM+ELR+NEMS)	**49.237**	**43.71**	**45.79**	**50.14**	**64.92**	**58.36**	**61.9**
Perera et al.	39.6	33.8	38.8	45.3	47.9	38.5	40.0

Table 12
The average accuracy of the implicit entity linking method compared to the baseline on different subsets of the datasets.

	Perera et al.'s Dataset	Our Dataset	Combined
Our Approach (Macro Average)	60.13	50.759	53.43
Perera et al.'s Approach (Macro Average)	39.25	41.08	40.55
Our Approach (Micro Average)	60.18	47.75	50.786
Perera et al.'s Approach (Micro Average)	39.27	38.789	38.919

5.2. Candidate Ranking Evaluation

We further analyze the performance of the candidate ranking process. We evaluate the candidate ranking performance as the proportion of tweets with the target entity correctly ranked as the top entity in our SVM^{rank} learning to rank set-up, using 5-fold cross validation. In order to show the complexity of disambiguation for implicit entity linking, we have implemented two baselines with basic IR techniques, i.e., TF and TF-IDF. Results, as seen in Table 8, show P@1 of retrieval of candidate entities' textual documents with the input tweet as the query. It is clear that the performance of these baselines is quite weak and points to the challenging nature of the problem that is being addressed in this paper.

Table 13

Error analysis of the ranking model.

Error Type	Person	Organization	Location	Event	Product/ Device	WrittenWork	Film	Example tweet
Competing entities	26%	–	33%	–	4%	6%	25%	From Film: `@LordHighway: "There is no nobility in poverty."" Leonardo DiCaprio playing as Jordan belfort. Massive movie.`
Similar entities	22%	43%	17%	31%	–	47%	28%	From Person: `No he’s not the de caprio of the music industry. Check the reviews His albumhas a 68 on metacritic while every other album that was nominated in the album of the year had better reviews. DAMN-95 Melodrama-91 4:44-82 Awaken My love-78 24K Magic- 71`
Insufficient Wikipedia textual content	–	–	–	3%	–	14%	35%	From WrittenWork: `Best Sales Speech of All Time! I read this book 7 years ago, I am glad it was a hit! Jordan Belfort definitely...`
Cross links across Wikipedia pages	11%	24%	–	9%	24%	–	–	From Organization: `Dwyane Wade put his ego aside because he knew it was best for his team: "I knew for me to be successful and for me to come to this team and bring what I can to this team, the starting unit just wasn’t a unit for me." https://usat.ly/2l3Kesl`
Temporally evolving entities	–	–	–	19%	54%	–	–	From Device: `Cute, but hardly a major selling point. I love my Apple devices and their many, but animoji’s aren’t one of them. Try harder.`
Incorrect or misleading explicit entities	16%	11%	24%	33%	–	7%	–	From Event: `I really can’t believe that our president is so horrible that people are encouraging Oprah to run all because she gave a speech at an awards show...`
Miscellaneous	25%	22%	26%	5%	18%	26%	12%	From WrittenWork: `SRK with Brad Pitt,in Ted Talks,with Netflix CEO, the top actor of Indian cinema in IMDB SRK WINs WEF AWARD pic.twitter.com/ehLvGw6Jsk`

To evaluate our work, as seen in Table 9, we compare the performance of the three main variations of the retrieval models with the baseline, i.e., SDM, interpolation of SDM and ELR, as well as the interpolation of SDM, ELR and NEMS. We find that the SDM+ELR+NEMS model outperforms the baseline, i.e., Perera et al.'s model, as well as SDM and SDM+ELR. The improvement of this model over SDM+ELR shows that the neural embedding feature introduced in this paper significantly impacts the ranking performance. This is in line with earlier findings that neural embedding features can enhance retrieval performance in the context of learning to rank methods (Ensan et al., 2017) and ad-hoc retrieval (Zamani and Croft, 2017; Bagheri et al., 2018).

In order to evaluate the performance (P@1) of the NEMS model as compared to other entity embedding models, we use two entity embedding models as baselines, namely Yamada et al. (2018) and Li et al. (2016). To do so, we replace our NEMS model with the aforementioned embedding models in order to obtain entity similarities. In so doing, we have interpolated the three embedding models' output feature values with SDM+ELR and performed ranking. Based on the obtained results, as seen in Table 10, our model outperforms both models on all categories except for the Product/Device, which points to the suitability of our proposed NEMS embedding method for this task compared to the two baseline embedding models.

Table 11 reports the overall accuracy of the implicit entity linking process. It should be noted that the recall of the candidate selection step is not comparable to P@1 of the ranking method because these are the performances of two separate steps. The overall performance of our method and the baseline, when considering both steps, would be the product of the performance of each step, reported in Table 11. As seen in the table, our approach shows significant improvement compared to the baseline on all domains.

Furthermore, the average accuracy of our method was compared to the baseline per different subsets of the dataset in Table 12. Our dataset is comprised of 5 categories of Person, Organization, Location, Event and Product/Device; Perera et al.'s dataset contains WrittenWork (referred to as Book in their paper) and Film categories; and finally, the combined dataset contains all the tweets from our dataset as well as Perera et al.'s dataset. As seen in the table, our proposed approach consistently outperforms the baseline regardless of which subset of the dataset is used.

5.3. Qualitative Error Analysis

We manually checked each instance that was not correctly identified by our approach in order to understand why and how the errors were caused in our framework. This resulted in the identification of seven error types which have been reported in Table 13. Alongside each error is a percentage demonstrating how often it occurred in the total amount of errors for that domain and an example explaining the error. In the following, we elaborate on each error type:

(1) *Competing entities*: This type of error deals with instances which include the mention of a highly influential and important entity in their description. For instance, Leonardo DiCaprio has a lead role in both Shutter Island and the Wolf of Wall Street movies, and thus he is a significant element in both entities. Therefore, the films' Wikipedia pages contain information about the shared entity which, in turn, leads to an inaccurate disambiguation of the implicit entity;
(2) *Similar entities*: This type of error occurs between two candidates which share very similar characteristics and descriptions. For instance, Bruno Mars and Rihanna who have similar music genres, associations, nominations, and are both A-list celebrities, can be easily confused with one another. Such similarities will lead to many similar entity mentions between their Wikipedia pages, which will cause this error;
(3) *Insufficient textual content*: This type of error occurs when the target entity's Wikipedia page has a low amount of textual content. In such a case, there is not much data for the MRF feature functions to accurately extract feature function values to distinguish the target entity from the non-target entity. For instance, The Wolf of Wall Street (book) Wikipedia page only has approximately 140 words;
(4) *Cross links across Wikipedia pages*: This type of error occurs when entities contain information about each other on their Wikipedia pages. This is common among entities such as representation of organizations which compete against each other because they will often have information about each other on their Wikipedia pages to provide context. For instance, Miami Heat and Cleveland Cavaliers both play in the NBA, and thus they have information about each other regarding player transfers, competitions, and so forth.
(5) *Temporally evolving entities*: This error occurs often when an evolution of the same entity happens over time. For instance, when a new version of a phone is released and each have their own Wikipedia pages. Often the two Wikipedia pages contain very similar information which causes this error. For instance, iPhone 5 and iPhone 4 can be confused together;
(6) *Incorrect or misleading explicit entities*: This type of error occurs when there is no entity connecting the Wikipedia page and tweet which may occur because of one of the following reasons: (1) the entity tagger does not detect a mention in the tweet or incorrectly tags a mention or (2) the Wikipedia page does not mention an explicit entity which is present in the tweet; in both cases, the model cannot benefit from the shared entities between the tweet and Wikipedia textual content pertinent to the target entity. As an instance, Oprah Winfrey was tagged in a tweet about the Golden Globe Awards, but she is not found in Golden Globe Award's Wikipedia page contents; and,
(7) *Miscellaneous*: This category applies to any error type other than the aforementioned ones. Majority of errors of this type could not be related to a certain cause and the rest are too low in number to be assigned a category. In the example tweet as seen in the table, Peter Thiel is ranked higher than Reed Hastings (target entity); we assume that it is due to abundance of entities in the tweet which increases the chance of the model failing to rank the entity correctly. As the output of the entity tagger, the important entities tagged are Netflix and Twitter. Reed Hastings is the CEO of

Netflix and Peter Thiel mentions Twitter on his Wikipedia page. This is a unique case as it deals with tweets which have many entities which can be related to different candidates, while having the implicit entity clear to the reader.

6. Concluding Remarks

In this paper, we proposed an approach for performing implicit entity linking on tweets and introduced a gold standard dataset for that task. Our work is novel in that: (1) we do not directly link a tweet to the knowledge graph entity of interest and instead indirectly link the tweet based on its similarity to the textual content available for it on Wikipedia; (2) we formulate the problem of implicit entity linking as an ad hoc document retrieval task where the ranking of textual content, here Wikipedia textual content, for a tweet determines the relevant implicit entity; (3) we propose a new neural embedding-based feature function and incorporate it into the MRF-based retrieval framework; and finally, (4) we introduce and make publicly available a gold standard dataset for the task of implicit entity linking, which we believe will play a role in fostering research on this task. We have shown, based on the gold standard, that our method outperforms the baseline in precision and accuracy. There are several areas that warrant further work in the area of implicit entity linking and would strengthen our proposed work in this paper:

(1) The work presented in this paper is restricted to using Wikipedia textual content for shaping the document space. There are other sources of textual content, for instance online user-generated content, where the entities of some domains receive explicit user reviews. An example of this is professional sport players, e.g., Kobe Bryant. For such an entity, there are many Web pages or forums that discuss this player. Our future work will focus on deriving user-generated content from sources including user-written reviews, forums and/or Facebook/Twitter posts.

(2) The main assumption of our work and the state of the art work in implicit entity linking has been that the domain of the tweet is known. This could be useful for domain-specific implicit entity linking. However, a more practical process would need to be able to perform linking without the need for a known domain. In our future work, we are interested in how *open-domain* implicit entity linking can be performed.

(3) Finally, due to the unavailability of a complete dataset in this field, the first logical step for doing implicit entity linking, i.e., implicit entity recognition, has been overlooked. With the proposed gold standard dataset, we are interested in exploring the detection of implicit entities in text, specifically tweets, as well as classification of the implicit mention domain.

References

Bagheri, E., Ensan, F. & Al-Obeidat, F. (2018). Neural word and entity embeddings for ad hoc retrieval. *Inf. Process. Manage.*, *54*(4), 657–673. doi:10.1016/j.ipm.2018.04.007.

Basile, P., Basile, V., Nissim, M. & Novielli, N. (2015). Deep tweets: from entity linking to sentiment analysis. In *Proceedings of the Italian Computational Linguistics Conference (CLiC-it 2015).*

Bianchi, F., Palmonari, M. & Nozza, D. (2018). Towards Encoding Time in Text-Based Entity Embeddings. In *International Semantic Web Conference* (pp. 56–71). Springer.

Cano Basave, A.E., Varga, A., Rowe, M., Stankovic, M. & Dadzie, A.-S. (2013). Making sense of microposts (# msm2013) concept extraction challenge.

Chang, M., Hsu, B.P., Ma, H., Loynd, R. & Wang, K. (2014). E2E: An End-to-End Entity Linking System for Short and Noisy Text. In *Workshop on Making Sense of Microposts.* (pp. 62–63).

Cornolti, M., Ferragina, P. & Ciaramita, M. (2013). A Framework for Benchmarking Entity-annotation Systems. In *Proceedings of the 22Nd International Conference on World Wide Web*. WWW '13 (pp. 249–260). New York, NY, USA: ACM. doi:10.1145/2488388.2488411.

Derczynski, L., Maynard, D., Rizzo, G., van Erp, M., Gorrell, G., Troncy, R., Petrak, J. & Bontcheva, K. (2015a). Analysis of named entity recognition and linking for tweets. *Information Processing & Management*, *51*(2), 32–49.

Derczynski, L., Maynard, D., Rizzo, G., van Erp, M. & et al, G.G. (2015b). Analysis of named entity recognition and linking for tweets. *Inf. Process. Manage.*, *51*(2), 32–49. doi:10.1016/j.ipm.2014.10.006.

Edouard, A., Cabrio, E., Tonelli, S. & Le Thanh, N. (2017). Semantic linking for event-based classification of tweets. *International Journal of Computational Linguistics and Applications*, 12.

Ensan, F. & Bagheri, E. (2017). Document Retrieval Model Through Semantic Linking. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, WSDM 2017, Cambridge, United Kingdom, February 6-10, 2017* (pp. 181–190). http://dl.acm.org/citation.cfm?id=3018692.

Ensan, F., Bagheri, E., Zouaq, A. & Kouznetsov, A. (2017). An Empirical Study of Embedding Features in Learning to Rank. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM 2017, Singapore, November 06 - 10, 2017* (pp. 2059–2062). doi:10.1145/3132847.3133138.

Fang, Y. & Chang, M.-W. (2014). Entity linking on microblogs with spatial and temporal signals. *Transactions of the Association for Computational Linguistics*, *2*, 259–272.

Feng, Y., Zarrinkalam, F., Bagheri, E., Fani, H. & Al-Obeidat, F. (2018). Entity linking of tweets based on dominant entity candidates. *Social Network Analysis and Mining*, *8*(1), 46.

Ferragina, P. & Scaiella, U. (2012). Fast and Accurate Annotation of Short Texts with Wikipedia Pages. *IEEE Software*, *29*(1), 70–75. doi:10.1109/MS.2011.122.

Finin, T., Murnane, W., Karandikar, A., Keller, N., Martineau, J. & Dredze, M. (2010). Annotating named entities in Twitter data with crowdsourcing. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk* (pp. 80–88). Association for Computational Linguistics.

Greenfield, K., Caceres, R.S., Coury, M., Geyer, K., Gwon, Y., Matterer, J., Mensch, A., Sahin, C.S. & Simek, O. (2016). A Reverse Approach to Named Entity Extraction and Linking in Microposts. In *# Microposts* (pp. 67–69).

Grishman, R. & Sundheim, B. (1996). Message understanding conference-6: A brief history. In *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics* (Vol. 1).

Guo, S., Chang, M. & Kiciman, E. (2013). To Link or Not to Link? A Study on End-to-End Tweet Entity Linking. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 9-14, 2013, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA* (pp. 1020–1030). http://aclweb.org/anthology/N/N13/N13-1122.pdf.

Hasibi, F., Balog, K. & Bratsberg, S.E. (2016). Exploiting Entity Linking in Queries for Entity Retrieval. In *Proceedings of the 2016 ACM International Conference on the Theory of Information Retrieval*. ICTIR '16 (pp. 209–218). New York, NY, USA: ACM. doi:10.1145/2970398.2970406.

Honnibal, M. & Johnson, M. (2015). An Improved Non-monotonic Transition System for Dependency Parsing. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* (pp. 1373–1378). Lisbon, Portugal: Association for Computational Linguistics. https://aclweb.org/anthology/D/D15/D15-1162.

Hosseini, H., Nguyen, T.T. & Bagheri, E. (2018). Implicit Entity Linking Through Ad-Hoc Retrieval. In *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)* (pp. 326–329). IEEE.

Hua, W., Zheng, K. & Zhou, X. (2015). Microblog Entity Linking with Social Temporal Context. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*. SIGMOD '15 (pp. 1761–1775). New York, NY, USA: ACM. doi:10.1145/2723372.2751522.

Ibrahim, Y., Amir Yosef, M. & Weikum, G. (2014). AIDA-Social: Entity Linking on the Social Stream. In *Proceedings of the 7th International Workshop on Exploiting Semantic Annotations in Information Retrieval*. ESAIR '14 (pp. 17–19). New York, NY, USA: ACM. doi:10.1145/2663712.2666185.

Jovanović, J. & Bagheri, E. (2017). Semantic annotation in biomedicine: the current landscape. *Journal of biomedical semantics*, *8*(1), 44.

Kim, J.-D., Ohta, T., Tsuruoka, Y., Tateisi, Y. & Collier, N. (2004). Introduction to the bio-entity recognition task at JNLPBA. In *Proceedings of the international joint workshop on natural language processing in biomedicine and its applications* (pp. 70–75). Association for Computational Linguistics.

Li, Y., Zheng, R., Tian, T., Hu, Z., Iyer, R. & Sycara, K. (2016). Joint embedding of hierarchical categories and entities for Concept Categorization and Dataless Classification. *arXiv preprint arXiv:1607.07956*.

Liu, X., Li, Y., Wu, H., Zhou, M., Wei, F. & Lu, Y. (2013). Entity Linking for Tweets. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 1304–1311).

Masud, M.M., Chen, Q., Khan, L., Aggarwal, C., Gao, J., Han, J. & Thuraisingham, B. (2010). Addressing concept-evolution in concept-drifting data streams. In *Data Mining (ICDM), 2010 IEEE 10th International Conference on* (pp. 929–934). IEEE.

Meij, E., Weerkamp, W. & De Rijke, M. (2012). Adding semantics to microblog posts. In *Proceedings of the fifth ACM international conference on Web search and data mining* (pp. 563–572). ACM.

Metzler, D. & Croft, W.B. (2005). A Markov random field model for term dependencies. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 472–479).

Metzler, D. & Croft, W.B. (2007). Latent Concept Expansion Using Markov Random Fields. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '07 (pp. 311–318). New York, NY, USA: ACM. doi:10.1145/1277741.1277796.

Perera, S., Mendes, P.N., Sheth, A.P., Thirunarayan, K., Alex, A., Heid, C. & Mott, G. (2015). Implicit Entity Recognition in Clinical Documents. In *Proceedings of the Fourth Joint Conference on Lexical and Computational Semantics, *SEM 2015, June 4-5, 2015, Denver, Colorado, USA.* (pp. 228–238). http://aclweb.org/anthology/S/S15/S15-1028.pdf.

Perera, S., Mendes, P.N., Alex, A., Sheth, A.P. & Thirunarayan, K. (2016). Implicit Entity Linking in Tweets. In *European Semantic Web Conference 2016* (pp. 118–132). doi:10.1007/978-3-319-34129-3$_8$.

Ritter, A., Clark, S., Etzioni, O., et al. (2011). Named entity recognition in tweets: an experimental study. In *Proceedings of the conference on empirical methods in natural language processing* (pp. 1524–1534). Association for Computational Linguistics.

Rizzo, G., Pereira, B., Varga, A., van Erp, M. & Cano Basave, A.E. (2017). Lessons learnt from the Named Entity rEcognition and Linking (NEEL) challenge series. *Semantic Web*, 1–34.

Satoshi, S. & Hitoshi, I. (2000). IREX: IR and IE Evaluation project in Japanese. In *Proceedings of the 2nd International Conference on Language Resources & Evaluation.*

Sekine, S. & Nobata, C. (2004). Definition, Dictionaries and Tagger for Extended Named Entity Hierarchy. In *LREC* (pp. 1977–1980). Lisbon, Portugal.

Shen, W., Wang, J. & Han, J. (2015). Entity linking with a knowledge base: Issues, techniques, and solutions. *IEEE Transactions on Knowledge and Data Engineering*, *27*(2), 443–460.

Shen, W., Wang, J., Luo, P. & Wang, M. (2013). Linking Named Entities in Tweets with Knowledge Base via User Interest Modeling. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '13 (pp. 68–76). New York, NY, USA: ACM. doi:10.1145/2487575.2487686.

Song, Y., Kim, E., Lee, G.G. & Yi, B.-k. (2004). POSBIOTM-NER in the shared task of BioNLP/NLPBA 2004. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications* (pp. 100–103). Association for Computational Linguistics.

ter Horst, H., Hartung, M. & Cimiano, P. (2017). Joint Entity Recognition and Linking in Technical Domains Using Undirected Probabilistic Graphical Models. In *International Conference on Language, Data and Knowledge* (pp. 166–180). Springer.

Tjong Kim Sang, E.F. & De Meulder, F. (2003). 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-Volume 4* (pp. 142–147). Association for Computational Linguistics.

Torres-Tramón, P., Hromic, H., Walsh, B., Heravi, B.R. & Hayes, C. (2016). Kanopy4Tweets: Entity Extraction and Linking for Twitter. In *# Microposts* (pp. 64–66).

Waitelonis, J. & Sack, H. (2016). Named Entity Linking in# Tweets with KEA. In *# Microposts* (pp. 61–63).

Yamada, I., Shindo, H. & Takefuji, Y. (2018). Representation Learning of Entities and Documents from Knowledge Base Descriptions. *arXiv preprint arXiv:1806.02960.*

Yamada, I., Shindo, H., Takeda, H. & Takefuji, Y. (2017). Learning distributed representations of texts and entities from knowledge base. *arXiv preprint arXiv:1705.02494.*

Zamani, H. & Croft, W.B. (2017). Relevance-based Word Embedding. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval 2017* (pp. 505–514). New York, NY, USA: ACM. doi:10.1145/3077136.3080831.