Tweeki: Linking Named Entities on Twitter to a Knowledge Graph

Bahareh Harandizadeh
University of California
Irvine, CA 92617
bharandi@uci.edu

Sameer Singh
University of California
Irvine, CA 92617
sameer@uci.edu

Abstract

To identify what entities are being talked about in tweets, we need to automatically link named entities that appear in tweets to structured KBs like WikiData. Existing approaches often struggle with such short, noisy texts, or their complex design and reliance on supervision make them brittle, difficult to use and maintain, and lose significance over time. Further, there is a lack of a large, linked corpus of tweets to aid researchers, along with lack of gold dataset to evaluate the accuracy of entity linking. In this paper, we introduce (1) Tweeki, an unsupervised, modular entity linking system for Twitter, (2) TweekiData, a large, automatically-annotated corpus of Tweets linked to entities in WikiData, and (3) TweekiGold, a gold dataset for entity linking evaluation. Through comprehensive analysis, we show that Tweeki is comparable to the performance of recent state-of-the-art entity linkers models, the dataset is of high quality, and a use case of how the dataset can be used to improve downstream tasks in social media analysis (geolocation prediction).

1 Introduction

Popularity and steady increase in adoption of social media makes it a ripe domain for understanding and analyzing world events, with Twitter as one of the largest social media platforms. As a result, tweets now have become a rich source of information, and Twitter analysis has been widely applied for many applications such as trend detection (Lau et al., 2012), opinion mining (Pak and Paroubek, 2010), election politics (Conover et al., 2011), and many others. However, short length of the text, casual and error-prone writing style, and evolving topics over time is extremely challenging for existing text analysis tools (Derczynski et al., 2015).

One of the common approaches is to bridge the gap between unstructured text (e.g. a tweet) and structured, machine-readable knowledge bases (e.g. Wikidata) using *entity linking* (EL) that grounds named mentions to a unique, real-world entity, i.e., an entry in the knowledge base (see Figure 1 for an example). Entity linking has been widely applied in natural language processing (Ling et al., 2015; Gupta et al., 2017; Raiman and Raiman, 2018; Radhakrishnan et al., 2018) on domains including news, biographical text, movie/show plots, amongst others. Although entity linking has been used in social media applications as well (Miyazaki et al., 2018; Dai et al., 2018), it is much less common.

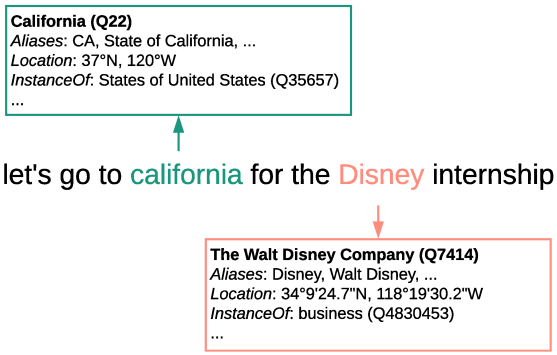

Figure 1: **Example** of an *entity-linked* tweet, containing two mentions linked to WikiData entities.

There are a number of reasons by existing entity linking systems are not commonly used for social media analysis. One of the primary concerns is that many of the existing entity linking systems are supervised (Yosef et al., 2011; Ganea and Hofmann, 2017), which not only makes them suitable for the domains they are trained on (Meij et al., 2012), but also makes them excessively reliant on context around the mention. For these reasons, supervised EL systems tend to be inaccurate on noisy and short text (Cornolti et al., 2013). Even unsupervised systems are heavily-engineered and complex in nature (Kulkarni et al., 2009), containing rules, obso-

Proceedings of the 2020 EMNLP Workshop W-NUT: The Sixth Workshop on Noisy User-generated Text, pages 222–231
Online, Nov 19, 2020. ©2020 Association for Computational Linguistics

lete lexicons, low coverage KBs, heuristic scoring, etc., making them difficult to maintain, extend, and adapt. This is especially a problem for social media where relevant (and new) entities, writing styles, and vocabulary change completely and frequently. Finally, we currently lack linked datasets to use for social media analysis, such as manually annotated data to evaluate, compare, and benchmark these entity linking systems. These shortcomings need to be addressed before entity linking can be used widely for social media analysis.

In this paper, we propose *Tweeki*, a system and resource for entity linking Twitter to WikiData, that addresses the above challenges. The Tweeki entity linking pipeline is unsupervised, simple, and modular, thus mitigating the problems arising from the supervised setup and complexity design. The pipeline components are chosen based on their performance on short, noisy texts, and can be easily extended with more/new entities and improved taggers. Further, we use WikiData[1] as the KB, benefiting from regular updates and higher coverage compared to Wikipedia and other KBs. We run the Tweeki system on a large collection of tweets to provide the first large, automatically-linked corpus of tweets, that we call *TweekiData*. Finally, we also manually annotate a small set of tweets to provide *gold* annotations for entity links to create the *TweekiGold* dataset, which can be used to evaluate and compare entity linking systems on social media text. The implementation of Tweeki, and the accompanying datasets (TweekiData and TweekiGold) are available publicly[2].

We provide a comprehensive evaluation of the pipeline, using the manually annotated TweekiGold data and other tweet-based datasets. We show that Tweeki performs comparably to leading sophisticated entity linking systems and proposed models for NEEL2016 challenge (Radovanovic et al., 2016). Further, we carry out analysis on mention extraction and candidate generation for different types of mentions, and show the variety and diversity of mentions and entities that get linked. Finally, as a potential downstream application of Tweeki, we develop a preliminary model for geolocation prediction from tweet text, and show that leveraging Tweeki leads to increased the accuracy of such systems for social media analysis.

[1] https://www.wikidata.org/wiki/
[2] Code and data at https://ucinlp.github.io/tweeki.

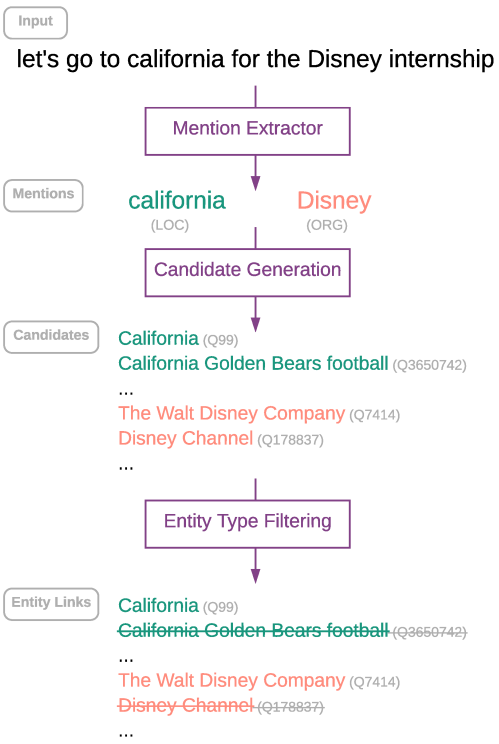

Figure 2: **Tweeki Pipeline (and example execution):** containing detection of mention spans, generation of candidates (and their probabilities, omitted for clarity), and a final entity type-based filtering.

2 Tweeki Entity Linking Pipeline

In this section we propose a simple, unsupervised entity linker called Tweeki that, as we will show later, performs similar to the state of the art. First, we introduce the task definition and some assumptions that we made to build Tweeki, then describe the pipeline components in more detail.

2.1 Task and Pipeline Overview

Given a tweet T, the *end-to-end* linking task is to provide a set of segmented mentions with their associated entities: $T \rightarrow \{(m_i, l_i)\}$. A mention (m_i) refers to a span of tokens in the tweet that refers to an entity, and an entity (l_i) refers to an item of a knowledge base (here, Wikidata unique ID). We exclude the so-called *NIL* entities in our setup, i.e. mentions of entities that do not appear in the KB.

The Tweeki pipeline consists of three modules, shown in Figure 2. Starting from taking raw text of the tweet as input, Tweeki first extracts the mentions and assigns each an entity type. Then, for each identified mention, Tweeki generates a set of candidate entities (with corresponding scores based on prior probability). Finally, after type

compatibility-based filtering, a candidate entity (remaining one with the maximum score) is selected as the predicted link. Since we are primarily interested in WikiData, we make certain assumptions that are only relevant for that KB, such as existence of aliases for most of the entities, being able to infer coarse-grained entity types for them ($\mathcal{T}_{\text{NER}} = \{\text{PER}, \text{LOC}, \text{ORG}, \text{MISC}\}$), and others that we mention in text.

2.2 Mention Extraction

Extracting mentions (contiguous spans of tokens) to be linked from the tweet is the first step of any entity linking pipeline. We focus on named entity linking in this work, and thus our mentions are similar to the standard task of named entity recognition (NER) in natural language processing. To support easy extensibility and deployment, we use an off-the-shelf NER systems released as part of AllenNLP library (Gardner et al., 2017). We compare against other NER systems to make this decision, which we present in the results. Further, we were concerned about the accuracy of these newswire trained taggers on the short and noisy tweets, however, as we will see later, they were fairly accurate on the standard NER metrics. The mention extraction module thus returns a list of mentions with associated types, i.e. $\{(m_i, t_i)\}$ where $t_i \in \mathcal{T}_{\text{NER}}$. Although we do not use contextual embeddings in the rest of the pipeline (which links each mention independently), the NER model uses contextual representation to identify the mentions and types, thus provides much of their benefit.

2.3 Candidate Generation

Given a mention m_i, we use the KB itself to produce a set of candidate entities, with associated scores, that will allow us to estimate the conditional probability $p(c|m_i)$. In current literature, there are primarily two ways to generate such candidates: (1) Crosswiki links, i.e. a web crawl that aggregates anchor links to Wikipedia entities (Ling et al., 2015), and (2) Intrawiki links: i.e. doing the same within Wikipedia (Ratinov et al., 2011). We use the latter approach since it is much easier to maintain and update over time. To adapt Intrawiki links to the WikiData KB, we use the existing links between Wikipedia and WikiData entities to gather all the entity aliases and number of time each alias is used in Wikipedia for the entity. The candidate generation module thus returns a set of scored candidates for each mention, i.e. $\{(m_i, C_i)\}$ where

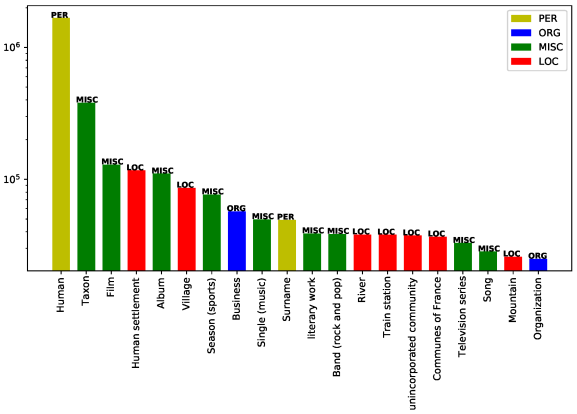

Figure 3: **Entity Type Distribution**: Distribution of the 20 most frequent objects of the *P31: Instance Of* relation from WikiData, which are manually classified into four types from $\mathcal{T}_{\text{NER}}$.

$C_i = \{(c_j, p(c_j|m_i)\}$, where c_j are candidate entities to link the mentions. See Figure 2 for an example (we omit the probabilities for clarity). Note that candidate generation ignores the context (rest of the tweet), and thus is far from optimal. However, when combined with entity type filtering (next), the "prior" probabilities often produce the correct links, as we will show in the experiments.

2.4 Incorporating Entity Types

As a motivating example, consider the following tweet: "*Syracuse and Pitt in the #ACC*". In general, "*Syracuse*" may refer to a name of a city, or a name of an American basketball team, with a lower prior probability for the latter. However, by applying contextual information(e.g "*#ACC*"), the basketball team seems more appropriate. As a result, an entity linking system should have a mechanism to promote more contextually consistent candidates, while lowering the score of the others.

We incorporate this intuition by filtering the prior probability computed during candidate generation ($p(c_j|m_i)$), the notion of entity types of the candidates ($t_j \in \mathcal{T}_{\text{NER}}$, for every c_j), and the type of each mention ($t_i \in \mathcal{T}_{\text{NER}}$ for each m_i). For extracting the entity type (t_j), we use the *P31: Instance Of* relation from the WikiData ontology to find unique entity types, and manually categorize all types that occure more than 100 times (2098 types) into the four types in $\mathcal{T}_{\text{NER}}$. We show the occurence frequency of the top-20 types in Figure 3, along with the coarse-grained types they are assigned (e.g. *Human*'s assigned type is PER, *Film* is MISC, *River* is LOC, and *Business* is ORG). For example Obama (Q76) is Instance Of (P31) *Human* and *Human*'s

assigned type is PER, so Q67 entity type is PER.

Mathematically, we assume $p(c_j|m_i, t_i, T) = p(c_j|m_i, t_i)$, where T denotes the tweet containing this mention, as we assume the candidate c_j and the tweet context are conditionally independent if both the mention and the mention type are given. Then we assign zero probabilities to all candidate where $t_j \neq t_i$[3], and renormalize the probabilities of the remaining candidates to obtain the final probability $\hat{p}(c_j|m_i)$. Although we pick the entity for each mention by picking the maximum score, $l_i = \arg\max_{c_j \in C_i} \hat{p}(c_j|m_i)$, we still renormalize to produce a valid confidence in the link. Returning to the above example, if NER assigns LOC to *Syracuse*, then we filter candidates to only be LOC type (e.g Q128069-city name), and similarly restrict to MISC types if that is assigned by NER (e.g Q15718182-basketball team).

3 TweekiData and TweekiGold

In this section, we will describe TweekiData, a massive automatically-linked corpus created by running Tweeki on a number of *source* datasets (3.1). We also introduce TweekiGold, a small, manually-annotated dataset for measuring the quality of the linker and the automatically linked dataset.

3.1 TweekiData

TweekiData is a large automatically-annotated dataset, which is linked to Wikidata using Tweeki. The linking of text and KG is a valuable resource for learning representations that enable better reasoning about entities and how they are expressed in text (tweets in this case), such as pretrained language models that use such resources for better contextual modeling (Peters et al., 2019; Logan et al., 2019). Having access to the full knowledge graph can also help models that perform entity analysis using hops in the knowledge graphs, i.e. location modeling by using relations such as livesIn or headquarteredIn of the linked entities, even if the location is not mention directly.

As the source of the tweets in TweekiData, we identify two prominent datasets that are commonly used in the community, in order to ensure the resulting dataset will be useful. (1) **BTC**, or "Broad Twitter Corpus" (Derczynski et al., 2016), is seven sets of gold datasets of tweets collected over stratified times, places and social, and is widely used for Named Entities Recognition. We use section A and H of this corpus, with the size of 1000 and 2000 tweets respectively, and (2) **UTGEO2011**, a massive Twitter dataset mainly created for tweet geolocation prediction, but also used for other purposes (Roller et al., 2012). The dataset is limited to US region and has two versions: UTGEO2011-Large containing 38M tweets belongs to almost 450K users, and UTGEO2011-Small contains 1.6M tweets with 10K users. We took a random subset of 5M tweets from UTGEO2011, to build TweekiData.

[3]if there are no such candidates, we ignore this filtering.

	TweekiGold	TweekiData
# tweets	500	5M
# tokens/tweet	16.31	14.41
# mentions (toks)	8,155	8,010,253
# mentions (spans)	958	5,038,870
# links	852	1,954,229
# uniq entities	638	273,685

Table 1: **Statistics** of the Tweeki-linked datasets.

3.2 TweekiGold

As there is no gold data available linking Twitter to Wikidata, we collect a gold dataset manually. We use mention extraction on tweets from UTGEO2011-small, and select 700 random tweets to annotate. An expert manually provides the following for each tweet: correct NER tags from $\mathcal{T}_{\text{NER}}$ (in IOB2 format), a Wikidata entity ID for all applicable spans, and Wikipedia page-title for corresponding Wikidata entity. If the tweet is too ambiguous or erroneous, it is deleted by annotator. Finally 500 tweets remain as the final gold dataset that we call TweekiGold for future use.

3.3 Dataset Statistics

We present statistics of both datasets in Table 1. TweekiGold contains 500 tweets mostly about sport and social events limited to US region geographically. There are 8155 tokens in total, and the length of each tweet is 16.31 on average. The 958 mentions consist of 399 LOC, 171 MISC, 212 ORG and 176 PER entity types. We are able to link 852 of them to WikiData (the rest could not be matched to existing entities), resulting in 1.91 mentions and 1.7 links per tweet on average. The number of all tokens in the larger TweekiData dataset is 72,086,330, so on average the length of each tweet is 14.41. It also contains 5M mentions of which almost 2M are linked to Wikidata successfully (40%).

	Spacy		Stanford		AllenNLP	
	P	R	P	R	P	R
TweekiGold	43.8	57.9	71.8	65.2	81.1	80.9
BTC-A	10.8	42.4	41.1	56.5	48.1	66.6
BTC-H	7.3	14.3	40.6	19.9	74.2	54.2
Average	20.6	38.2	51.2	47.2	**67.8**	**67.2**

Table 2: **NER Performance:** Token-wise accuracy of popular frameworks on tweet-based NER datasets.

Type	#mentions	#entities	Coverage
PER	2.1m	550k	25%
LOC	1.8m	950k	52%
ORG	550k	200k	35%
MISC	490k	200k	40%

Table 3: **Linking Coverage of TweekiData, by Types**: Number of mentions extracted for each type, along with how many of these mentions are linked by Tweeki.

	Precision	Recall	F1
w/o Entity Types	66.6	59.6	63.4
w/ Entity Types	**69.1**	**61.2**	**65.1**

Table 4: **Entity Type Filtering**: Tweeki with entity type filtering obtains more than 2% improvement on all metrics compared to without filetring.

4 Experiments

In this section, we present experiments to address the following questions: (Section 4.1) what is the quality of each components in Tweeki pipeline? (Section 4.2) how does the performance of our simple Tweeki linker compare to other existing linkers? and finally, (Section 4.3) how can we use Tweeki for other use cases for NLP on Twitter?

4.1 Evaluating the Linker Components

Here we will investigate few of the individual modules and design choices of the Tweeki pipeline.

Mention Extraction To find the best NER-Tagger to use for mention extraction, we examined the accuracy of the three well-known, available NER taggers on TweekiGold and the both sections of BTC datasets. Table 2 shows precision and recall of StanfordNLP[4], AllenNLP (Gardner et al., 2017), and Spacy[5]. Based on the results, on TweekiGold, AllenNLP has 10% more precision compare to the second best (Stanford), while also superior on BTC-A (by 7%) and in BTC-B (by 30%). On average (last row), AllenNLP has around 16% more precision and 20% more recall compare to the second best option(Stanford). This consistent outstanding performance on different datasets shows AllenNLP can handle the casual and error-prone nature of tweets well, and thus is best to use for extracting mentions among popular alternatives.

Candidate Generation The primary way to evaluate the effectiveness of the candidate generation is its coverage, i.e. what fraction of the input mentions is it able to provide candidates for? We evaluate the coverage of Tweeki using the complete 5M tweets from the TweekiData described in Section 3.1. Table 3 shows the results for each $\mathcal{T}_{\text{NER}}$ types separately. LOC type has the best performance with 52% linked items. Missed locations mainly happen due to overly specific information (e.g 2nd Street) or noisy writing style (using "la" instead of "LA" or "Los Angeles"). Also the low coverage of PER type is mostly related to the mentions started with '@', making them hard to match with any aliases in the KB. Although we use Twitter API[6] to substitute these mentions to their real name (e.g change "@MittRomney" to "Mitt Romney"), the source dataset was gathered in 2011, and many of these are not valid anymore.

Need for Entity Types Finally, to show how incorporating entity types can be helpful in linking process, Tweeki was tested on TweekiGold dataset, with and without considering the third module of the pipeline. As shown in Table 4, using entity types can improves all metrics more than 2%, justifying its inclusion in the entity linking pipeline.

4.2 Comparison to Existing Linkers

In this section, we analyze the overall performance of Tweeki by comparing it with other linkers on TweekiGold, NEEL2016 (Rizzo et al., 2015), and Derczynski datasets (Derczynski et al., 2015). For the linkers to compare against, we include TagMe (Ferragina and Scaiella, 2010) as it is designed for short and noisy text. AIDA (Yosef et al., 2011) and Babelfly (Moro et al., 2014) both use graph building and dense subgraph algorithms to tackle entity linking and usually provide a good baseline for many previous studies. We also compare more recent EL models, End-to-End Neural

[4] https://nlp.stanford.edu/software/CRF-NER.html
[5] https://spacy.io/usage/linguistic-features
[6] https://developer.twitter.com/en

	NEEL2016			Derczynski			TweekiGold		
	P	R	F1	P	R	F1	P	R	F1
TagMe	19.1	**30.0**	24.1	18.2	**50.1**	26.3	38.1	56.1	45.0
Babelfy	8.08	10.6	9.06	9.0	41.1	15.2	17.1	47.2	25.1
AIDA	-	-	-	-	-	-	53.2	32.1	38.5
End-to-End	**87.9**	13.1	22.8	**57.05**	29.2	**39.0**	**79.1**	35.2	49.4
OpenTapioca	11.0	19.1	14.8	9.1	36.0	14.0	20.2	50.4	29.1
Tweeki	58.0	15.2	**24.8**	41.1	34.2	37.1	69.0	**61.0**	**65.0**

Table 5: **Entity Linking Performance** of existing linkers, using *strong matching* metric on three datasets.

	Derczynski			TweekiGold		
	P	R	F1	P	R	F1
Tw-Stanford	36.4	29.4	32.5	56.7	44.6	49.9
Tw-AllenNLP	41.1	34.2	37.1	69.0	61.0	65.0

Table 6: **Choice of Mention Extraction:** using Stanford for mention extraction and NER, compared to using AllenNLP, in the first module of the pipeline.

	ORG		PER		LOC	
	P	R	P	R	P	R
TweekiGold	68.1	66.8	53.4	84.4	82.3	77.1
BTC-A	34.5	27.2	40.2	62.1	56.8	49.0
BTC-H	31.3	10.0	60.9	21.0	63.4	50.8
Average	44.6	34.6	51.5	55.8	67.5	58.9

Table 7: **Span-based accuracy** for each mention type on different datasets annotated with gold NER.

Entity Linking (Kolitsas et al., 2018) and OpenTapioca (Delpeuch, 2019).

Table 5 compares these models on different datasets using precision/recall/F1 based on *EL strong matching*. TagMe has acceptable performance on all datasets and the best Recall for NEEL2016, while Babelfy performs the worst, specifically on NEEL2016. While End-to-End is not specifically designed for short, noisy text, it is the winner of all three datasets in terms of precision. OpenTapioca has average performance on all datasets, with low accuracy on NEEL2016. Our proposed system, Tweeki, has the best F1 on NEEL2016 and TweekiGold, while being quite close to the best on Derczynski. Tweeki also has provides a relative high precision on all datasets.

We also compare Tweeki to the best submissions for the NEEL2016 challenge. Even though Tweeki is unsupervised and not specifically designed for this challenge (i.e. the prominent entities in that dataset), it would place third in the challenge, obtaining 24.8 F1 behind 39.6 F1 from Greenfield et al. (2016) and 50.1 F1 from KEA (Waitelonis and Sack, 2016)), both of which are supervised.

Finally, to emphasize how selecting NER tagger effects the pipeline, we substitute AllenNLP with Stanford for *Mention Extraction* in Tweeki and tested it on Derczynski and TweekiGold. As shown in Table 6, this choice has a significant effect. By using AllenNLP in mention extraction, not only desired spans are selected and passed to the next module properly, but also more accurate entity types improve the filtering of candidates.

4.3 Use Case: Geolocating Tweets

We can use the links of named mentions to a knowledge base for a number of interesting applications in social media analysis. Prediction the location of tweets, for instances, has been a popular task for understanding the geographic trends and behaviors (Cheng et al., 2010; Chang et al., 2012; Miura et al., 2017), since users do not provide this information accurately (Chang et al., 2012). Some recent models have even used KBs for geolocation (Miyazaki et al., 2018). We will study a use case of Tweeki for geolocation prediction.

Using UTGEO2011-small dataset (see Section 3.1 for more details), we want to predict the location of each tweet. Each tweet in the dataset has a *true* label (longitude, latitude), which is framed as a supervised classification problem of which US city (of 378 most popular ones) and state the tweet originated from. Our main intuition is to incorporate the locations mentioned in the tweet explicitly as part of the input to the classifier, since people likely talk about nearby locations.

As our focus is on LOC mentions, we analyze AllenNLP mention extraction capability on different mention types, and show, in Table 7, that it achieves a 67% accuracy on average for LOC mentions, the best accuracy among other types. We also show how often these mentions are linked in Table 8, indicating most of the LOC mentions actually get linked to entities in the KB. For all the linked mentions, we can extract the relevant data from the KB,

	Size	#LOC	#Links	Coverage
Train	544,667	385,295	219,167	**56%**
Test	527,783	322,852	201,563	**62%**

Table 8: **Statistics of UTGEO2011-small**, with number of all LOC mentions, number of mentions that are linked to KB, and fraction of tweets with at least 1 link.

	State prediction		City prediction	
	Base	+Tweeki	Base	+Tweeki
Tweet-level	18.0	**19.3**	9.1	**11.0**
User-level	24.3	**26.1**	13.0	**15.2**

Table 9: **City and State Prediction** using base model (without any extra information) and when combined with locations from Tweeki preprocessing.

in this case the actual geographical *coordinates* of each mentioned entity (using WikiData as the KG makes this much easier as it is structured, compared to linking to Wikipedia as is common). We convert these coordinates to their nearest US state and city, and append these locations to the tweet (a simple form of feature engineering). For example the tweet: "*Duran Duran Concert (@Nokia Theatre w/ others)*" will change to "*Duran Duran Concert (@Nokia Theater w/ others) Los Angeles, California*". We apply above to the whole dataset, and train a simple deep learning model (BiLSTM).

Based on the results in Table 9, using Tweeki and appending Wikidata information to tweets increases accuracy for the both tweet and user-level (for user-level prediction we aggregate output probabilities of all tweets from the user, then choose the most probable label). This demonstrates that even such a simple approach to incorporating KB information can provide improvements to existing problems, suggesting many applications of entity linking to tweets and other social media text.

5 Related Work

Entity Linking Systems Entity linking (EL) of tweets has attracted a lot of attention recently (Liu et al., 2013; Huang et al., 2014; Sikdar and Gambäck, 2016; Nie et al., 2018). Similar to entity linking systems for general text, EL for tweets is primarily composed of two major steps: 1) the identification of the mentions, similar to tasks such as term expansion (Zou et al., 2014), and 2) identifying the candidate entities to the identified mentions. For the latter step, roughly two types of features are used. Local features identify one mention at the time and disambiguate it separately such as using prior probability in Liu et al. (2013) or temporal relevance mention in Tran et al. (2015). Global features take a more comprehensive view and consider the relations between the entity candidates for the different mentions of the tweet (Huang et al., 2014; Feng et al., 2018). However, global approaches are more challenging in noisy domains like tweets, and unlikely to provide significant benefits for short texts. Some approaches use a graph-based representation to combine of local and global features (Huang et al., 2014). Moreover, recently, neural network methods have been applied to entity linking to model the local contextual information, such as (Nie et al., 2018) that captures semantic information between the local context and the candidate entity via representation-based and interaction-based neural semantic matching models.

Among all the proposed models, in this paper, we chose TagMe (Ferragina and Scaiella, 2010) that uses global features in an unsupervised manner, and Babelfy (Moro et al., 2014) that uses random walks and a densest subgraph algorithm for jointly disambiguating word senses and entity linking. Although Tag-Me is specifically designed for short text, Babelfy is a general purpose entity linker which also suitable for short and highly ambiguous text disambiguation (Moro et al., 2014). We also consider other popular linking approaches from outside of social media such as AIDA (Hoffart et al., 2011b), which uses Stanford NER Tagger and adopts the YAGO2 knowledge base (Hoffart et al., 2011a). From recent state-of-the-art models, we select End-to-End Neural Entity Linking (Kolitsas et al., 2018) that uses context-aware compatibility score based on word and entity embeddings, coupled with a neural attention and a global voting mechanisms, and OpenTapioca (Delpeuch, 2019) as an end to end EL approach to Wikidata that relies on topic similarities and local entity context. Although these models are not specifically designed for noisy and short text, but they are sophisticated general purpose EL proposed recently and tested on different data types including Twitter.

Twitter-related datasets Many Twitter-based datasets have been introduced for different research goals. Related to EL task, we can divide the datasets into two categories: named entity recognition (NER) datasets and named entity linking (NEL) datasets. NER datasets focus on identifying mentions and their types, such as dataset by Ritter et al. (2011) or BTC (Derczynski et al., 2016), with

gold dataset published for concept extraction challenge in #MSM2013 (Cano Basave et al., 2013).

Based on GERBIL benchmark report (Röder et al., 2018), for the A2KB (NER and NEL) task in tweets, the most commonly used datasets have been introduced in "Making Sense of Microposts" challenge (#Microposts) from 2014 till 2016. Among them, Named Entity Extraction and Linking Challenge2016 (NEEL2016) is the most popular one to use, consisting of 296 tweets in testset with 3.4 mentions in each tweet on average. It is also valuable to mention that 384 out of 1022 mentions in this dataset refer to three topics: *"Donald Trump"*, *"StarWars"* and *"StarWars (The Force Awakens)"*(Nie et al., 2018). Another dataset designed for A2KB task is Derczynski et al. (2015), consist of 183 tweets with 1.57 entities per tweet on avg. As these datasets links are not provided in Wikidata ID, we designed a converter to map each link to its corresponding Wikidata ID.

6 Conclusions

Although entity linking for social media text has many potential applications, it has not been widely adopted by the community due to presence of supervised and complex entity linking systems that are hard to maintain, extend, and apply to new entities and different writing styles. Further, there is no large-scale linked corpus of tweets available for researchers to use for social media analysis, and very few *gold* annotated tweet datasets to evaluate and compare different entity linking systems. Our proposed work, collectively called Tweeki, consists of an unsupervised, extensible entity linking pipeline, a massive automatically-linked dataset of tweets (TweekiData), and a small, manually annotated dataset of gold links (TweekiGold). Our experiments show that the linker is accurate, and the dataset can be used to obtain improvements in downstream applications. We have released the source code and datasets from this paper at https://ucinlp.github.io/tweeki.

Acknowledgements

We would like to thank the anonymous reviewers for their feedback. We are also grateful for authors of End-to-End Neural Entity Linking, OpenTapioca, Spacy, and AllenNLP for making their code available. This work was funded in part by National Science Foundation award #IIS-1817183.

References

Amparo Elizabeth Cano Basave, Andrea Varga, Matthew Rowe, Milan Stankovic, and Aba-Sah Dadzie. 2013. Making sense of microposts (# msm2013) concept extraction challenge.

H. Chang, D. Lee, M. Eltaher, and J. Lee. 2012. @phillies tweeting from philly? predicting twitter user locations with spatial word usage. In *2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 111–118.

Zhiyuan Cheng, James Caverlee, and Kyumin Lee. 2010. You are where you tweet: A content-based approach to geo-locating twitter users. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, CIKM 10, page 759768, New York, NY, USA. Association for Computing Machinery.

M. D. Conover, B. Goncalves, J. Ratkiewicz, A. Flammini, and F. Menczer. 2011. Predicting the political alignment of twitter users. In *2011 IEEE Third International Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third International Conference on Social Computing*, pages 192–199.

Marco Cornolti, Paolo Ferragina, and Massimiliano Ciaramita. 2013. A framework for benchmarking entity-annotation systems. In *Proceedings of the International World Wide Web Conference (WWW) (Practice & Experience Track)*.

Hongliang Dai, Yangqiu Song, Liwei Qiu, and Rijia Liu. 2018. Entity linking within a social media platform: A case study on yelp. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2023–2032, Brussels, Belgium. Association for Computational Linguistics.

Antonin Delpeuch. 2019. Opentapioca: Lightweight entity linking for wikidata. *ArXiv*, abs/1904.09131.

Leon Derczynski, Kalina Bontcheva, and Ian Roberts. 2016. Broad twitter corpus: A diverse named entity recognition resource. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1169–1179, Osaka, Japan. The COLING 2016 Organizing Committee.

Leon Derczynski, Diana Maynard, Giuseppe Rizzo, Marieke van Erp, Genevieve Gorrell, Raphal Troncy, Johann Petrak, and Kalina Bontcheva. 2015. Analysis of named entity recognition and linking for tweets. *Information Processing & Management*, 51(2):32 – 49.

Yue Feng, Fattane Zarrinkalam, Ebrahim Bagheri, Hossein Fani, and Feras Al-Obeidat. 2018. Entity linking of tweets based on dominant entity candidates. *Social Network Analysis and Mining*, 8:1–16.

Paolo Ferragina and Ugo Scaiella. 2010. Tagme: on-the-fly annotation of short text fragments (by wikipedia entities). In *CIKM '10*.

Octavian-Eugen Ganea and Thomas Hofmann. 2017. 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. Association for Computational Linguistics.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke S. Zettlemoyer. 2017. Allennlp: A deep semantic natural language processing platform.

K. Greenfield, Rajmonda S. Caceres, M. Coury, K. Geyer, Youngjune Gwon, J. Matterer, A. Mensch, C. Sahin, and Olga Simek. 2016. A reverse approach to named entity extraction and linking in microposts. In *#Microposts*.

Nitish Gupta, Sameer Singh, and Dan Roth. 2017. Entity linking via joint encoding of types, descriptions, and context. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Johannes Hoffart, Fabian M. Suchanek, Klaus Berberich, Edwin Lewis-Kelham, Gerard de Melo, and Gerhard Weikum. 2011a. Yago2: Exploring and querying world knowledge in time, space, context, and many languages. In *Proceedings of the 20th International Conference Companion on World Wide Web*, WWW 11, page 229232, New York, NY, USA. Association for Computing Machinery.

Johannes Hoffart, Mohamed Amir Yosef, Ilaria Bordino, Hagen Fürstenau, Manfred Pinkal, Marc Spaniol, Bilyana Taneva, Stefan Thater, and Gerhard Weikum. 2011b. Robust disambiguation of named entities in text. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 782–792, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Hongzhao Huang, Yunbo Cao, Xiaojiang Huang, Heng Ji, and Chin-Yew Lin. 2014. Collective tweet wikification based on semi-supervised graph regularization. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 380–390, Baltimore, Maryland. Association for Computational Linguistics.

Nikolaos Kolitsas, Octavian-Eugen Ganea, and Thomas Hofmann. 2018. End-to-end neural entity linking. In *Proceedings of the 22nd Conference on Computational Natural Language Learning*, pages 519–529, Brussels, Belgium. Association for Computational Linguistics.

Sayali Kulkarni, Amit Singh, Ganesh Ramakrishnan, and Soumen Chakrabarti. 2009. Collective annotation of wikipedia entities in web text. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD 09, page 457466, New York, NY, USA. Association for Computing Machinery.

Jey Han Lau, Nigel Collier, and Timothy Baldwin. 2012. On-line trend analysis with topic models: #twitter trends detection topic model online. In *Proceedings of COLING 2012*, pages 1519–1534, Mumbai, India. The COLING 2012 Organizing Committee.

Xiao Ling, Sameer Singh, and Daniel S. Weld. 2015. Design challenges for entity linking. *Transactions of the Association for Computational Linguistics*, 3:315–328.

Xiaohua Liu, Yitong Li, Haocheng Wu, Ming Zhou, Furu Wei, and Yi Lu. 2013. Entity linking for tweets. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1304–1311.

Robert L. Logan, Nelson F. Liu, Matthew E. Peters, Matt Gardner, and Sameer Singh. 2019. Barack's Wife Hillary: Using Knowledge Graphs for Fact-Aware Language Modeling. In *Association for Computational Linguistics (ACL)*, pages 5962–5971.

Edgar Meij, Wouter Weerkamp, and Maarten de Rijke. 2012. Adding semantics to microblog posts. In *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining*, WSDM 12, page 563572, New York, NY, USA. Association for Computing Machinery.

Yasuhide Miura, Motoki Taniguchi, Tomoki Taniguchi, and Tomoko Ohkuma. 2017. Unifying text, metadata, and user network representations with a neural network for geolocation prediction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1260–1272, Vancouver, Canada. Association for Computational Linguistics.

Taro Miyazaki, Afshin Rahimi, Trevor Cohn, and Timothy Baldwin. 2018. Twitter geolocation using knowledge-based methods. In *Proceedings of the 2018 EMNLP Workshop W-NUT: The 4th Workshop on Noisy User-generated Text*, pages 7–16, Brussels, Belgium. Association for Computational Linguistics.

Andrea Moro, Alessandro Raganato, and Roberto Navigli. 2014. Entity linking meets word sense disambiguation: a unified approach. *Transactions of the Association for Computational Linguistics*, 2:231–244.

F. Nie, Shuyan Zhou, Jing Liu, Jinpeng Wang, Chin-Yew Lin, and R. Pan. 2018. Aggregated semantic matching for short text entity linking. In *CoNLL*.

Alexander Pak and Patrick Paroubek. 2010. Twitter as a corpus for sentiment analysis and opinion mining. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

Matthew E. Peters, Mark Neumann, Robert L. Logan, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A. Smith. 2019. Knowledge enhanced contextual word representations. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Priya Radhakrishnan, Partha Talukdar, and Vasudeva Varma. 2018. 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. Association for Computational Linguistics.

Danica Radovanovic, Katrin Weller, and Aba-Sah Dadzie. 2016. Making sense of microposts (#microposts2016) social sciences track. In *#Microposts*.

Jonathan Raiman and Olivier Raiman. 2018. Deeptype: Multilingual entity linking by neural type system evolution. In *AAAI*.

Lev Ratinov, Dan Roth, Doug Downey, and Mike Anderson. 2011. Local and global algorithms for disambiguation to Wikipedia. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1375–1384, Portland, Oregon, USA. Association for Computational Linguistics.

Alan Ritter, Sam Clark, Mausam, and Oren Etzioni. 2011. Named entity recognition in tweets: An experimental study. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1524–1534, Edinburgh, Scotland, UK. Association for Computational Linguistics.

G. Rizzo, M. V. Erp, J. Plu, and Raphaël Troncy. 2015. Making sense of microposts (#microposts2016) named entity recognition and linking (neel) challenge. In *#Microposts*.

Michael Röder, Ricardo Usbeck, and Axel-Cyrille Ngonga Ngomo. 2018. GERBIL - benchmarking named entity recognition and linking consistently. *Semantic Web*, 9(5):605–625.

Stephen Roller, Michael Speriosu, Sarat Rallapalli, Benjamin Wing, and Jason Baldridge. 2012. Supervised text-based geolocation using language models on an adaptive grid. In *EMNLP-CoNLL*.

Utpal Kumar Sikdar and Björn Gambäck. 2016. Twitter named entity extraction and linking using differential evolution. In *ICON*.

Tuan Tran, Nam Khanh Tran, Asmelash Teka Hadgu, and Robert Jäschke. 2015. Semantic annotation for microblog topics using Wikipedia temporal information. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 97–106, Lisbon, Portugal. Association for Computational Linguistics.

Jörg Waitelonis and H. Sack. 2016. Named entity linking in #tweets with kea. In *#Microposts*.

Mohamed Amir Yosef, Johannes Hoffart, Ilaria Bordino, Marc Spaniol, and Gerhard Weikum. 2011. Aida: An online tool for accurate disambiguation of named entities in text and tables. *PVLDB*, 4:1450–1453.

Xianqi Zou, Chengjie Sun, Yaming Sun, Bingquan Liu, and Lei Lin. 2014. Linking entities in tweets to wikipedia knowledge base. In *Natural Language Processing and Chinese Computing*, pages 368–378, Berlin, Heidelberg. Springer Berlin Heidelberg.