Understanding the Impact of Entity Linking on the Topology of Entity Co-occurrence Networks for Social Media Analysis

Nevin, J.; Zhang, P.; Dimitrov, D.; Lees, M.; Groth, P.; Dietze, S.

DOI
10.1007/978-3-031-77792-9_5

Publication date
2025

Document Version
Final published version

Published in
Knowledge Engineering and Knowledge Management

Understanding the Impact of Entity Linking on the Topology of Entity Co-occurrence Networks for Social Media Analysis

James Nevin[1(✉)], Pengyu Zhang[1], Dimitar Dimitrov[2], Michael Lees[1], Paul Groth[1], and Stefan Dietze[2]

[1] University of Amsterdam, Amsterdam, Netherlands
{j.g.nevin,p.zhang,m.h.lees,p.t.groth}@uva.nl
[2] GESIS - Leibniz Institute for the Social Sciences, Cologne, Germany
{dimitar.dimitrov,stefan.dietze}@gesis.org

Abstract. A common form of analysis of textual data is entity co-occurrence, where networks of entities and their connections within the text are constructed and their topology analysed. As the analysis is focused on the entities and their relations, the tools used to extract them can have a potentially large effect on the results. A frequently used method as part of these analyses is entity linking, where extracted entities are mapped to a knowledge graph. Many established entity linking tools have been created for long text following standard spelling and grammar rules. As a result, the tools struggle on short, unstructured text such as tweets. On such text, it can be difficult to choose between tools and parameter settings, especially since ground truth is often unavailable. Given these challenges in entity linking on text and the direct influence of extracted entities on subsequent network analysis, we propose the need to apply multiple tools to create a more holistic set of results. We verify this assertion through a set of experiments. Using a dataset of approximately 21 million English-language tweets, we construct multiple entity co-occurrence networks using two tools (Fast Entity Linker and DBpedia Spotlight) and numerous confidence thresholds for each. We find that standard network analysis metrics, such as size, connectivity, and centrality are all heavily influenced by the choice of entity linking tool.

Keywords: Entity linking · Co-occurrence networks · Network analysis · Social media

1 Introduction

Large amounts of text are generated on the internet everyday. This text offers many opportunities for identifying trends and understanding online discourse [6]. However, many of these text data are unstructured. Given this lack of structure and the large size of these data, approaches that abstract the data

M. Alam et al. (Eds.): EKAW 2024, LNAI 15370, pp. 69–85, 2025.
https://doi.org/10.1007/978-3-031-77792-9_5

into interpretable forms can be powerful ways to analyse them [5,25,35]. One such abstract form is entity co-occurrence networks [3]. These co-occurrence networks are graphs[1] with entities being represented by vertices and weighted edges between them showing how frequently the entities occur together in pieces of text. The resultant graph can be analysed using standard network analysis metrics, including number of vertices or degree distributions [7]. Analysing these can give valuable insight into the text, such as, for example, topics of discussion through the entities identified [1,6]. The structure of the network can further show which entities play prominent roles (based on their centrality), or how well connected they are to each other [29,30]. Large scale corpuses of annotated text, such as TweetsKB [11], provide opportunities for creating these entity co-occurrence graphs and discovering novel links/relations between entities. The topology of such co-occurrence networks determines the results of the analysis. Hence, results and subsequent conclusions are solely dependent on the approach taken to identify the entities within the text.

In order to gain additional insights, entities can be not only identified, but also linked to a knowledge base (KB). These knowledge bases are carefully constructed to contain accurate, unambiguous information. By linking the entities to such a knowledge base, ambiguities in the identified entities can be removed, which can be important for the construction of accurate co-occurrence networks. However, this process of entity linking is a difficult task and an active area of research [24,27,31]. Because of this, the choice of entity linking algorithm and setting of parameters is not always straightforward. As the downstream co-occurrence network is heavily influenced by the algorithm, care needs to be taken in its selection.

One popular source of large scale text data is Twitter/X. Twitter is one of the most popular online social media websites, and is regularly used for information dissemination and discourse. However, the aforementioned challenges of lack of structure apply especially to tweets for a number of reasons: there is frequent use of acronyms; highly irregular grammar; and brevity plus little context. The usefulness of entities in analysing such short text has already been shown [2], but performing entity linking on text with these properties poses extra challenges [14]. Even state-of-the-art entity linkers achieve F1 scores of around 0.3 on a recent benchmark dataset [23]. Furthermore, it can be unfeasible to manually label additional datasets to determine which algorithms are best.

Given these challenges and the low accuracy scores, an alternative approach is needed for the selection and parameter setting of entity linking algorithms in the construction of co-occurrence networks. In the literature, it is common to test on a benchmark dataset and select only a single entity linker. However, since accuracy scores are so low and benchmark datasets relatively small, this chosen linker likely includes some bias and thus potentially overlooks different possibilities for the created network. Instead, we argue for application of multiple different entity linkers and settings on the full dataset. By analysing the

[1] Throughout this paper, we generally use the terms 'graph' and 'network' interchangeably.

multiple different constructed networks, one can identify the variance that can arise through changing linking algorithms and their parameters. In the case that downstream network results show large variance for different algorithms, it is prudent to report results based on the multiple algorithms, hence offering a more holistic picture. On the other hand, if results are robust, simply choosing the 'best' algorithm/parameters based on some standard or benchmark is reasonable.

We illustrate this approach through a set of experiments in constructing entity co-occurrence networks from tweets. We test two entity linking algorithms (Fast Entity Linker [4] and DBpedia Spotlight [22]) on a set of 21 million English-language tweets over a three year period, for which there is no entity linking ground truth. For each algorithm, we apply 5 different confidence thresholds. Networks are compared based on standard network analysis metrics, showing high sensitivity to the different possible algorithms and parameter settings. In some cases, these differences could influence the interpretation of the text, such as which topics are most important. This emphasises the need to report results across multiple algorithms.

The contributions of this paper are:

1. entity linking of a dataset of 21 million tweets with two different entity linking algorithms and multiple confidence thresholds;
2. comparison of entity co-occurrence networks created using these linked entities, showing the high sensitivity of standard network analysis metrics to algorithms and parameters.

The rest of the paper is structured as follows: Sect. 2 covers the related works; Sect. 3 introduces the data and their collection process; Sect. 4 describes the approach of the experiments; Sect. 5 analyses the results; and Sect. 6 offers conclusions.

2 Related Work

Network analysis is a powerful tool for identifying structure and importance of entities through their position in the network. For example, Named Entity Recognition (NER) has been applied to books to create an entity co-occurrence network, which was used to identify interesting topological features, such as the small world property, of the corpus [3]. In a similar fashion, Manaskasemsak et al. also used NER on a collection of tweets to identify clusters in entity co-occurrence networks [21]. The authors related these clusters to real-world events, and found this to be an effective approach for connecting tweets to the events. However, the use of NER poses potential risks. Fegley and Torvik investigated the effects of non-disambiguated or incorrectly combined entities in entity networks. They found that certain network measures, such as clustering and assortivity, can be strongly affected by ambiguity [13]. Because of this, going a step beyond entity recognition can improve the accuracy of network analysis results.

Entity linking aims to accomplish this disambiguation by linking keywords in text to entities in a knowledge graph, with various tools such as Fast Entity Linker (FEL) [4], DBpedia Spotlight [22], REL [34], and GENRE [8] available. However, there are a number of challenges to this linking task. There are frequent problems in benchmark datasets used for evaluating entity linking models [33]. Many of these datasets are long-tailed, making it sometimes difficult to evaluate linker performance [18]. Additionally, entities within the knowledge base can be a simplification of those appearing in the text [32], or be missing entirely [8,12]. Various approaches such as entity spaces [32], autoregressive language models [8], dense representations from Transformers [16], and entity co-occurrence [19] have been suggested for creating accurate models that address these problems.

Several studies have tackled the further challenges of entity linking in short, unstructured text like tweets. Numerous Twitter entity disambiguation datasets have been introduced for analysing method robustness on noisy texts and identifying key error sources [9,14]. The largest of these, TweetNERD [23], addresses previous limitations by providing a broad time window, consistent annotations, and splits for assessing out-of-domain and temporal generalization performance. Various approaches to disambiguation on such data have also been tested, including unsupervised systems [14] and hybrid approaches that combine dense retrieval with long contextual representations from Wikipedia [15]. Collective inference methods that integrate mention-entry, entry-entry, and mention-mention similarities have been shown to improve performance [20].

However, the studies above mainly focus on technical performance and evaluation metrics. It overlooks how the choice of entity linking tools affects downstream results – in this case, network analysis metrics like size, connectivity, and centrality.

3 Data and Data Collection

We employed the long term Twitter archive that is the foundation of TweetsKB [11]. The archive is based on continuously capturing a data stream of 1% randomly sampled tweets from Twitter/X [17]. The archive contains more than 14 billion tweets collected over the past 10 years. To reduce the computational time for the purposes of our analysis, we use a subset from the archive for a socially relevant topic, namely, nationalism during the COVID-19 pandemic. The dataset has been extracted using a list consisting of 73 keywords[2] about COVID-19 and nationalism and contains English-language tweets from 08/2019 to 08/2022. To extract tweets, we performed an exact string match in case sensitive fashion for the keywords "ppe" and "PCR". For the remaining keywords, a tweet is extracted from the archive if a keyword matches a sub-string of the tweet in a case insensitive manner. The dataset[3] contains approximately 21 million tweets.

[2] Keywords: https://github.com/jim-g-n/Tweet-Linked-Entity-Co-occurrence/tree/main/dataset.

[3] The dataset extraction is part of a bigger initiative to create a comprehensive dataset about nationalism during COVID-19 in English, German, and Italian by Mark Dang-Anh, Giorgia Riboni and Dimitar Dimitrov.

This dataset is large and carefully curated, exactly the kind to which entity co-occurrence analysis might be applied.

4 Approach

We perform a set of experiments to highlight the effect that choice of entity linking algorithm can have on co-occurrence network analysis.[4] As the dataset lacks a ground truth for entity linking, no conclusions can be drawn as to the quality of identified links. Hence, algorithms can only be compared based on differences in the downstream application. For each of the algorithm and parameter settings, we create a network by linking entities from the tweet dataset. Nodes and edges in the network are defined by the linked entities and how frequently they occur together in the same tweet.

4.1 Entity Linking Algorithms

We test two different entity linking algorithms, Fast Entity Linker (FEL) [4] and DBpedia Spotlight [22]. We choose these two as FEL is intended for short text such as tweets while DBpedia Spotlight is multilingual and highly configurable. Both tools are well-established.

FEL works in a unsupervised fashion and does not require any parameterisation while allowing changing the knowledge base and the entity embedding used for linking. The algorithm was initially developed to work on short text (i.e. search queries) but found adoption in the creation of TweetsKB [11] and TweetsCOV19 [10], two large Twitter corpuses containing annotated tweets. The version of FEL used for this paper links to a Wikipedia dump from November 2023. Each linked entity from a piece of text is given a confidence score from -3 to 0, where a higher score is more strict. We apply different confidence score cutoffs to exclude entities with low confidence.

DBpedia Spotlight is an entity linker that links to DBpedia [22]. As stated before, the main advantage of DBPedia Spotlight is that it is highly configurable, as it allows users to compute scores such as prominence (reflected by the number of times a resource is mentioned in Wikipedia), topical relevance (a similarity score between the paragraph containing the candidate resource and DBpedia resource's context) and contextual ambiguity (representing the relative difference in topic score between the first and the second ranked candidate resource, if more than one candidate is available), and confidence scores. We used the March 2022 version of the Spotlight model from the DBpedia Databus repository[5] and focus only on the confidence scores. Linking can be performed with confidence scores between 0 and 1, with higher being more strict. DBpedia Spotlight also returns a support score per entity, but this is not relevant for our study since we base entity prominence on how frequently they occur in the tweets.

[4] Created networks and analysis code are available here: https://github.com/jim-g-n/Tweet-Linked-Entity-Co-occurrence.

[5] https://databus.dbpedia.org/dbpedia/spotlight/spotlight-model/2022.03.01.

For the FEL algorithm, we test confidence thresholds of {-3.00, -2.75, -2.5, -2.25, -2.00}. For DBpedia Spotlight, we test confidence scores of {0.5, 0.6, 0.7, 0.8, 0.9}. These scores are presented as they cover a wide range of confidences. Analysis of other thresholds showed that they were too strict (very few entities), too noisy (an excessive number of entities), or did not change the main conclusions of this paper, so we exclude them in the interest of space. Networks based on these additional thresholds are included in the associated repository.

4.2 Network Construction and Analysis

For a given algorithm and confidence threshold, we create a set of linked entities for each tweet. The nodes in the network are the total collection of unique entities from all tweets. The edges between pairs of nodes are weighted based on the count of how many tweets the respective linked entities occur together in. There is a total of 10 different graphs – 5 created using FEL and 5 using DBpedia Spotlight.

We compare the different created networks using standard network analysis metrics [13]. These include: number of nodes (equivalent to the number of unique linked entities), number of edges, diameter (the shortest distance between the most distant nodes), average degrees (average number of edges per node, which can be weighted or scaled), and component sizes (number of nodes in connected components in the network). These metrics generally show how large and well-connected the network is, which has many downstream implications and is thus of frequent concern to those analysing such networks. For all of the constructed networks, we present these metrics and highlight how they compare across different algorithms and parameter settings. For much of the analysis, we restrict ourselves to the largest connected component within each network, as per standard analysis practices. There are two differences we wish to draw attention to: changes in network properties for different confidence thresholds for a given algorithm, and differences between the two algorithms for comparable confidence thresholds. In this case, by comparable confidence thresholds, we mean thresholds that identify a similar number of unique entities in the dataset.

We also consider the differences in top nodes in the connected networks. We use the degree centrality (scaled number of edges per node) to rank the nodes and compare the top 10 entities in all networks. Top nodes in networks give an indication of the topics being discussed and the roles entities play in connecting these. To emphasise the differences, we also show the top weighted edges in two of the graphs for the different algorithms.

Networks were created and analysed using the graph-tool python library [26].

5 Results

For the analysis, we first cover the overlap in linked entities of the two algorithms. Following this, we analyse the topology of the full graphs and their largest connected component. Lastly, we highlight prominent nodes in the connected components. Our analysis is broad but not in-depth, as our emphasis is on the differences that can arise rather than the precise results themselves.

5.1 Linked Entities Overlap

Figure 1 shows the similarity of the unique entities linked to with the different algorithms and confidence thresholds. The x-axis shows the DBpedia Spotlight confidence threshold, while different coloured and shaped dots represent different FEL confidence thresholds; the y-axis shows the Jaccard similarity of the set of unique entities created using the given DBpedia Spotlight and FEL confidence thresholds.

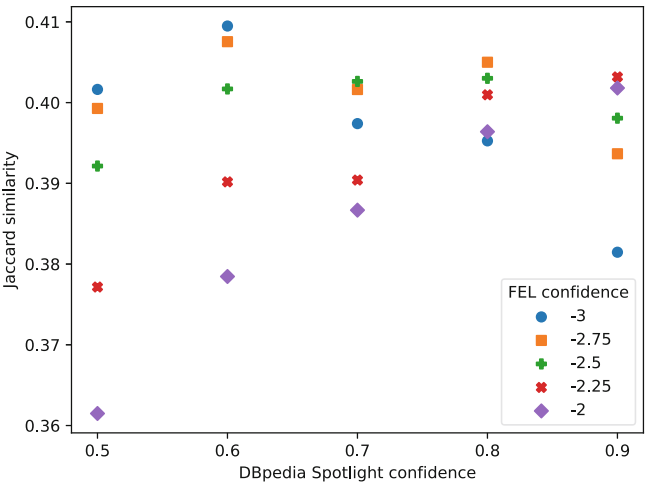

Fig. 1. Jaccard similarity (y-axis) of sets of unique entities linked using different DBpedia Spotlight (x-axis) and FEL (colours and shapes) confidence thresholds. (Color figure online)

The Jaccard similarity generally falls between 0.38 and 0.41. This is a relatively narrow range, given that the sizes of the sets can vary significantly for different confidence thresholds. That being said, the maximum value of 0.41 is fairly low, even when the size of the sets are similar, meaning the entities found by the two algorithms can differ substantially.

5.2 Full Graphs Topology

Tables 1 and 2 show, respectively, properties of the various co-occurrence networks created using FEL and DBpedia Spotlight. For both algorithms, the number of nodes (unique entities discovered) varies between approximately 535,000 and 355,000. The connectivity in the DBpedia Spotlight networks are relatively higher than the FEL networks for comparable confidence thresholds, with a larger number of edges and lower pseudodiameter (an estimate of the diameter).

Table 1. Graph properties of full co-occurrence networks created using FEL algorithm

Confidence	Num. nodes	Num. edges	Pseudodiameter	Average degree	Average weighted degree	Largest component
−3.00	535572	10019969	9	18.71	93.09	488506
−2.75	496670	7539889	10	15.18	75.77	442142
−2.50	456268	5496269	10	12.05	60.69	392744
−2.25	409826	3846504	11	9.39	46.83	336089
−2.00	357385	2594176	12	7.26	36.58	277843

Table 2. Graph properties of full co-occurrence networks created using DBpedia Spotlight algorithm

Confidence	Num. nodes	Num. edges	Pseudodiameter	Average degree	Average weighted degree	Largest component
0.5	515022	14113950	6	27.40	240.85	505326
0.6	473225	8975858	6	18.97	175.57	459781
0.7	435409	5855384	9	13.45	76.20	385827
0.8	401286	4176344	9	10.41	57.62	342799
0.9	358632	2920492	10	8.14	44.25	292811

Figure 2 shows the number of nodes (left) and percentage of nodes in the largest connected component (right) for different confidence thresholds of entity co-occurrence networks created using FEL (blue circles) and DBpedia Spotlight (orange diamonds). As seen in the tables, the number of nodes decreases for both algorithms as the confidence threshold is increased, and this decrease is roughly linear.

For both algorithms, the percentage of nodes in the largest connected component also decreases as the confidence threshold is increased. For the FEL algorithm, this decrease is slightly non-linear, reducing from 0.91 to around 0.78. For the DBpedia Spotlight algorithm, the decrease is not regular, with most nodes (¿0.95) remaining in the largest component with a confidence threshold of 0.5 or 0.6. There is a sharp drop from a confidence threshold of 0.6 to 0.7, followed by a roughly linear decrease. Overall, the DBpedia Spotlight graphs have generally higher connectivity than the FEL graphs, also relating back to the larger number of edges mentioned previously.

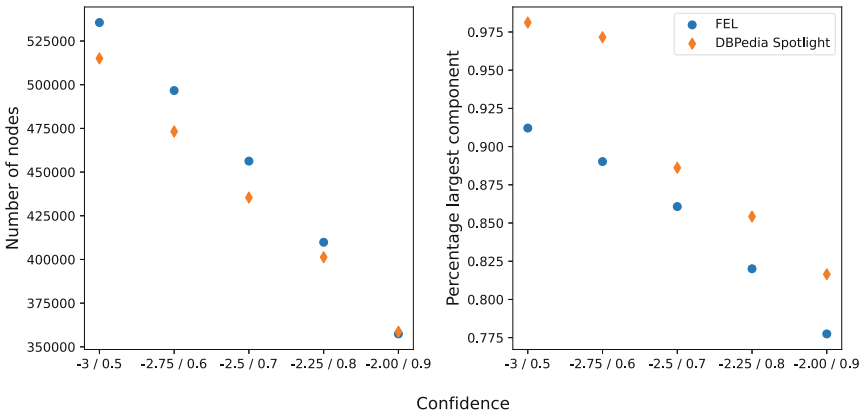

Fig. 2. Number of nodes (left) and percentage of nodes in largest connected component (right) for different confidence thresholds using FEL (blue circles) and DBpedia Spotlight (orange diamonds) (Color figure online)

5.3 Connected Graphs Topology

We restrict the rest of the analysis to the largest connected component of each graph, as this generally includes the majority of the nodes. Tables 3 and 4 show, respectively, the properties of the connected graphs created using the FEL and DBpedia Spotlight algorithms.

Table 3. Graph properties of connected co-occurrence networks created using FEL algorithm

Confidence	Num. nodes	Num. edges	Pseudodiameter	Average degree	Average weighted degree
−3.00	488506	10017512	9	20.51	102.05
−2.75	442142	7536960	10	17.05	85.11
−2.50	392744	5492703	10	13.99	70.48
−2.25	336089	3842187	11	11.43	57.07
−2.00	277843	2589306	12	9.32	47.02

Table 4. Graph properties of connected co-occurrence networks created using DBpedia Spotlight algorithm

Confidence	Num. nodes	Num. edges	Pseudodiameter	Average degree	Average weighted degree
0.5	505326	14113714	6	27.93	245.47
0.6	459781	8975501	6	19.52	180.70
0.7	385827	5853024	9	15.17	85.98
0.8	342799	4173397	9	12.17	67.43
0.9	292811	2917002	10	9.96	54.17

Figure 3 shows a number of different network metrics for the connected graphs created using both algorithms. Values for FEL are shown in blue circles, while DBpedia Spotlight are shown in orange diamonds. The average number of nodes (top left) are once again comparable between the two algorithms, with both showing a linear decrease as the confidence threshold is increased. However, as in the full graphs, other metrics show some large differences in behaviour.

The pseudodiameter, an approximation of the diameter of the graphs, is shown in the top right of the figure. In both cases, despite the number of nodes being smaller with a higher confidence threshold, increasing the confidence threshold increases the pseudodiameter, in a semi-stepwise fashion. Between the two algorithms, the pseudodiameter is much higher in the FEL graphs than the DBpedia Spotlight graphs, showing lower connectivity, even when the number of nodes is similar.

The middle row of the figure shows the (unscaled) average and weighted average degrees in the networks. We first note that both the average degree and average weighted degree are higher in the DBpedia Spotlight networks, but this difference becomes less pronounced at higher confidence levels. As the confidence threshold is increased, both the weighted and unweighted average degrees decrease for both algorithms, and this change is less linear in the DBpedia Spotlight networks than in the FEL networks. In particular, the weighted average degree has an inflection point at a confidence score of 0.7, but this behaviour is not observed in the FEL networks.

Similar behaviour can be observed in the scaled average weighted and unweighted degrees (bottom row of the figure). In this row, the average (weighted) degrees have been scaled based on the number of nodes in the networks. Differences between the two algorithms are slightly more pronounced, and the inflection point is still present in the weighted version for DBpedia Spotlight.

Figure 4 shows some calculated properties of the different connected networks. The top row shows the average local (left) and global (right) clustering, while the bottom row shows the unweighted (left) and weighted (right) assortivity, for different confidence thresholds. As before, FEL values are in blue circles and DBpedia Spotlight values in orange diamonds.

We again see differences between the two algorithms and different confidence thresholds for each. The average local clustering decreases as the confidence threshold is increased for both algorithms; however, this decrease is non-linear for DBpedia Spotlight, with a large drop going from 0.6 to 0.7. On the other hand, the global clustering values do not change monotonically, sometimes increasing and sometimes decreasing as the confidence threshold is increased. For the DBpedia Spotlight graphs, there is once again a large change when increasing the confidence threshold from 0.6 to 0.7. For all graphs, the average local clustering is fairly high, always above 0.4, indicating small-world properties regardless of choice of algorithm/confidence threshold.

The assortivity values are generally low for all networks. However, we still see trends, such as increasing unweighted assortivity for increasing confidence threshold and more sporadic metric behaviour in the DBPedia Spotlight graphs.

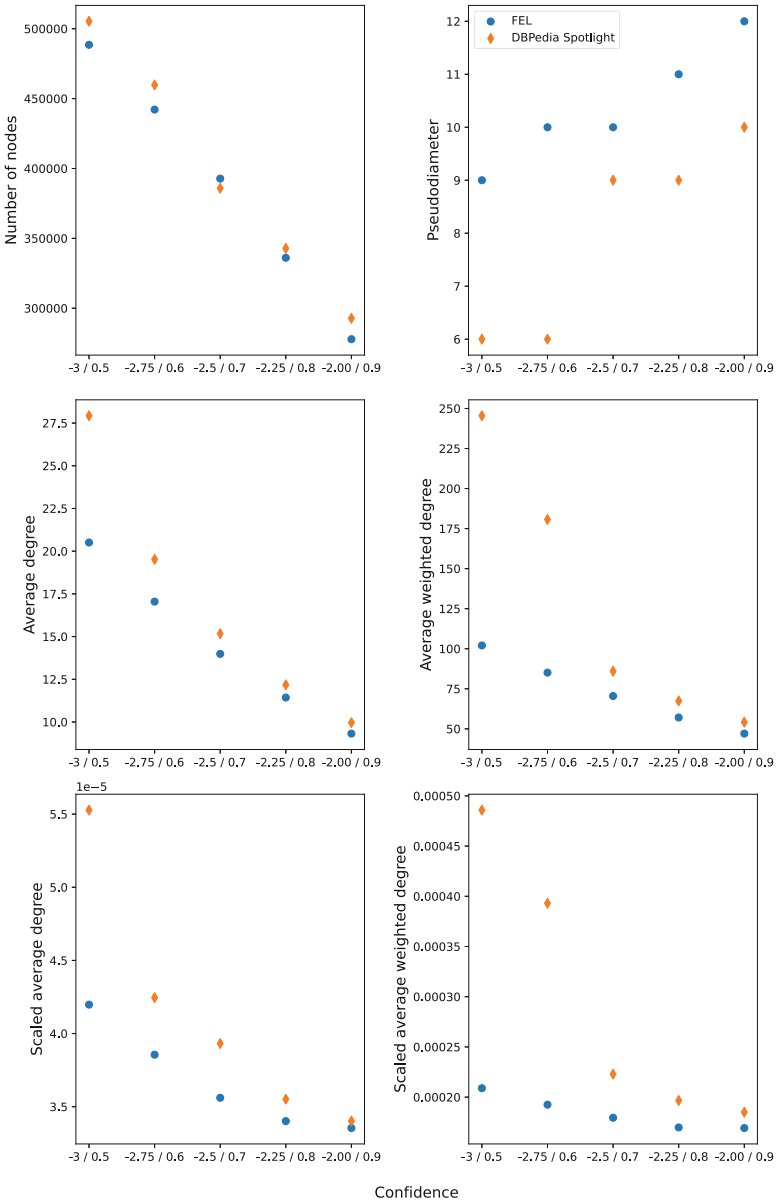

Fig. 3. Network metrics of largest connected components for different confidence thresholds (x-axes). Values for FEL and DBpedia Spotlight are in blue circles and orange diamonds, respectively. The first row shows the number of nodes (left) and pseudodiameters (right); the middle row shows the average degree (left) and average weighted degree (right); the bottom row shows the scaled average degree (left) and scaled average weighted degree (right). (Color figure online)

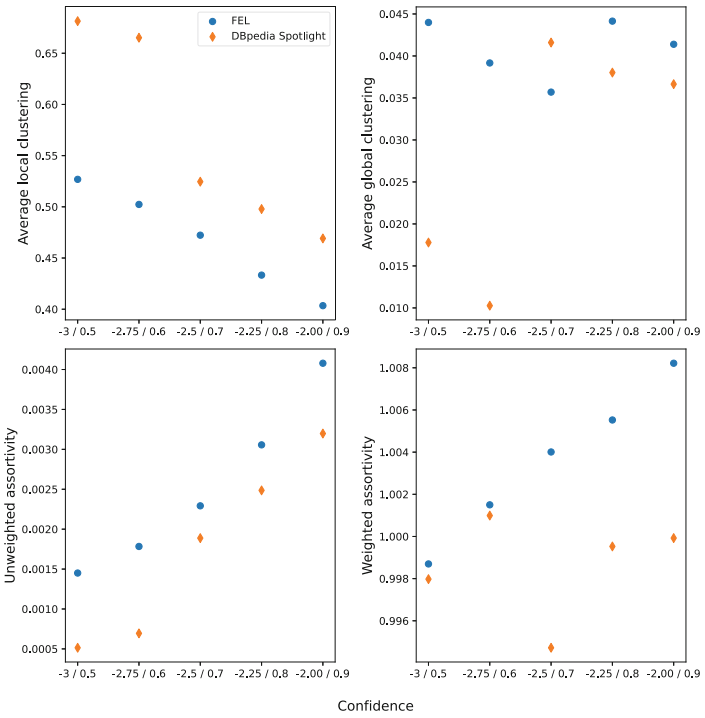

Fig. 4. Average network properties of largest connected components for different confidence thresholds (x-axes). Values for FEL and DBpedia Spotlight are in blue circles and orange diamonds, respectively. The top row shows the average local clustering (left) and average global clustering (right); the bottom row shows the unweighted assortivity (left) and weighted assortivity (right). (Color figure online)

5.4 Prominent Nodes

Tables 5 and 6 show the top 10 nodes by degree centrality in the FEL and DBpedia Spotlight networks, respectively. Top ranked nodes are generally more robust to changes in confidence thresholds than the other network properties, but there are some changes worth noting.

In all of the DBpedia Spotlight graphs, COVID-19 and related terms play a key role, regardless of confidence threshold. In the FEL graphs, however, making the threshold too strict (>-2.5) means that COVID-19 is no longer identified. We also see that the DBpedia Spotlight knowledge base has multiple entries relating to different aspects of COVID-19, but this is not the case with FEL. Besides for COVID-19, the top nodes in the two algorithms can be quite different, with the only other real overlap being 'China' and 'India'. Overall, the terms found by DBpedia Spotlight appear more reasonable.

We also note that applying stricter confidence thresholds can result in certain 'noisy' terms becoming more prominent. For example, 'Test_cricket' in DBpedia

Table 5. Top 10 degree nodes of connected co-occurrence networks created using FEL algorithm

−3	−2.75	−2.5	−2.25	−2
COVID-19	COVID-19	COVID-19	YouTube	YouTube
Twitter	Quarantine	India	India	India
Quarantine	India	YouTube	BTS	BTS
India	Midfielder	Twitter	China	Midfielder
Midfielder	YouTube	BTS	Midfielder	Twitter
Vaccination	Twitter	China	Twitter	Spotify
YouTube	Non-fungible_token	Midfielder	Non-fungible_token	Non-fungible_token
Non-fungible_token	China	Non-fungible_token	Spotify	Netflix
China	BTS	Spotify	Reblogging	Pfizer
BTS	Bachelor_of_Science	Bachelor_of_Science	Pfizer	MTV

Table 6. Top 10 degree nodes of connected co-occurrence networks created using DBpedia Spotlight Algorithm

0.5	0.6	0.7	0.8	0.9
Twitter	Twitter	COVID-19_vaccine	COVID-19_vaccine	COVID-19_vaccine
COVID-19_pandemic	COVID-19_pandemic	COVID-19_pandemic	COVID-19_pandemic	Test_cricket
COVID-19_vaccine	COVID-19_vaccine	COVID-19	Test_cricket	COVID-19
COVID-19	Coronavirus	Virus	COVID-19	Pfizer
Coronavirus	COVID-19	Test_cricket	Vaccine	COVID-19_pandemic
COVID-19_lockdowns	Virus	Vaccine	Pfizer	Vaccine
Virus	President_of_the_United_States	India	India	BTS
United_Kingdom	Test_cricket	Pfizer	China	Twitter
Quarantine	Vaccine	China	Twitter	YouTube
India	India	Radiotelephone	YouTube	India

Spotlight or 'Spotify' in FEL. This perhaps highlights potential issues with the algorithms, as it goes against the assumption that stricter confidence thresholds reduce noise.

Finally, Table 7 shows the top 10 weighted edges in the FEL -3 confidence threshold graph and the DBpedia Spotlight 0.5 confidence threshold graph. We see that the top edges with the two algorithms are completely different. The top edges in the DBpedia Spotlight graph are all related to the top nodes. This is not the case in the FEL graph, where many of the top edges connect to nodes not in the top 10. This once again highlights the differences in behaviour of the two algorithms.

5.5 Discussion and Possible Extensions

The results of these experiments highlight the sensitivity of network metrics to the choice of entity linker. Even when using confidence scores that identify a similar number of unique entities, DBpedia Spotlight finds these entities more regularly. The consequences of this are seen especially in the levels of network connectivity, with higher average degrees and largest components.

Table 7. Top 10 weighted edges in FEL graph with -3 confidence threshold and DBpedia Spotlight graph with 0.5 confidence threshold

FEL (-3 conf.)	DBpedia Spotlight (0.5 conf.)
(MTV, BTS)	(COVID-19_pandemic, Twitter)
(Pfizer, Moderna)	(Twitter, COVID-19_vaccine)
(Ministry_of_Health_and_Family_Welfare, India)	(Twitter, Coronavirus)
(Bachelor_of_Science, D%C3%A9FI)	(Twitter, COVID-19_lockdowns)
(Bachelor_of_Science, Whitelist)	(Twitter, COVID-19)
(Reblogging, Twitter)	(Quarantine, Twitter)
(D%C3%A9FI, Launchpad_%28website%29)	(Twitter, Test_cricket)
(Bachelor_of_Science, Launchpad_%28website%29)	(United_Kingdom, Twitter)
(D%C3%A9FI, Whitelist)	(Twitter, Virus)
(Umar, Riaz_%28actor%29)	(Twitter, India)

There are two clear avenues for further experiments. First, the already-created networks could be analysed in more sophisticated ways. There are a number of other metrics of interest, such as clustering and community analysis. These are common within the domain. However, they tend to be computationally expensive, and would need to be performed for all created networks. Another option would be running and analysing models on these networks. There are models for information spread or navigability, or sophisticated models for identifying important nodes in the network. These are also fairly expensive to run.

A second extension possibility is in the application of additional entity linking algorithms or network construction rules. Algorithms such as REL [34], GENRE [8], or Falcon2 [28] offer comparable performance to DBpedia Spotlight on tweets [23]. However, published versions of these link to an old KB, and thus miss crucial entities for this dataset such as COVID-19. Applying them would require refitting to a more recent KB. It would also be possible to add extra rules for including or excluding entities. For example, one could exclude entities that occur infrequently, allowing for less strict confidence thresholds. This would correspond to something like a global versus local significance criterion.

Finally, results of the analysis could be used to identify biases in the algorithms. Certain algorithms may perform better at finding long-tailed entities, while others might be better at identifying frequent entities. Comparing the networks constructed with different algorithms could be useful for seeing how such biases play out.

6 Conclusions

Analysing large scale text presents many challenges. The creation and interpretation of co-occurrence networks can offer useful insight into entities and their relationships in textual data. Extraction of these entities, and thus the

construction of the networks, is not an easy task. This is especially the case for loosely structured text, like in tweets. In the literature, usually only a single entity linking algorithm and parameter set are used. Hence, we highlighted some of the changes that can be seen in the co-occurrence networks when changing algorithms and their settings.

We found that all regularly reported network metrics can be affected. These effects can differ between the algorithms tested, sometimes showing non-linear behaviour with respect to parameter settings. Things like identified entities, relative importance of entities, and level of connectivity can all be affected. Given the sensitivities observed, it would be prudent to report results for this kind of network analysis using multiple entity linking approaches. This offers a wider range of possibilities in interpretation, and can thus increase confidence in the robustness of results.

Acknowledgments. This publication has benefited from an invited research stay at GESIS - Leibniz Institute for the Social Sciences.

Disclosure of Interest. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Al-Moslmi, T., Ocaña, M.G., Opdahl, A.L., Veres, C.: Named entity extraction for knowledge graphs: a literature overview. IEEE Access **8**, 32862–32881 (2020)
2. Alam, M., Bie, Q., Türker, R., Sack, H.: Entity-based short text classification using convolutional neural networks. In: Keet, C.M., Dumontier, M. (eds.) EKAW 2020. LNCS (LNAI), vol. 12387, pp. 136–146. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-61244-3_9
3. Amancio, D.R.: Network analysis of named entity co-occurrences in written texts. Europhys. Lett. **114**(5), 58005 (2016)
4. Blanco, R., Ottaviano, G., Meij, E.: Fast and space-efficient entity linking for queries. In: Proceedings of the Eighth ACM International Conference on Web Search and Data Mining, pp. 179–188 (2015)
5. Bono, C.A., Cappiello, C., Pernici, B., Ramalli, E., Vitali, M.: Pipeline design for data preparation for social media analysis. ACM J. Data Inf. Qual. **15**(4), 1–25 (2023)
6. Botzer, N., Weninger, T.: Entity graphs for exploring online discourse. Knowl. Inf. Syst. **65**(9), 3591–3609 (2023)
7. Cohen, R., Havlin, S.: Complex Networks: Structure, Robustness and Function. Cambridge University Press (2010)
8. De Cao, N., Izacard, G., Riedel, S., Petroni, F.: Autoregressive entity retrieval. arXiv preprint arXiv:2010.00904 (2020)
9. Derczynski, L., et al.: Analysis of named entity recognition and linking for tweets. Inf. Process. Manage. **51**(2), 32–49 (2015)
10. Dimitrov, D., et al.: TweetsCOV19-a knowledge base of semantically annotated tweets about the COVID-19 pandemic. In: Proceedings of the 29th ACM International Conference on Information & Knowledge Management, pp. 2991–2998 (2020)

11. Fafalios, P., Iosifidis, V., Ntoutsi, E., Dietze, S.: TweetsKB: a public and large-scale RDF corpus of annotated tweets. In: Gangemi, A., et al. (eds.) ESWC 2018. LNCS, vol. 10843, pp. 177–190. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-93417-4_12
12. Färber, M., Rettinger, A., El Asmar, B.: On emerging entity detection. In: Blomqvist, E., Ciancarini, P., Poggi, F., Vitali, F. (eds.) Knowledge Engineering and Knowledge Management: 20th International Conference, EKAW 2016, Bologna, Italy, November 19-23, 2016, Proceedings, pp. 223–238. Springer International Publishing, Cham (2016). https://doi.org/10.1007/978-3-319-49004-5_15
13. Fegley, B.D., Torvik, V.I.: Has large-scale named-entity network analysis been resting on a flawed assumption? PLoS ONE **8**(7), e70299 (2013)
14. Harandizadeh, B., Singh, S.: Tweeki: linking named entities on twitter to a knowledge graph. In: Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020), pp. 222–231 (2020)
15. Hebert, L., Makki, R., Mishra, S., Saghir, H., Kamath, A., Merhav, Y.: Robust candidate generation for entity linking on short social media texts. arXiv preprint arXiv:2210.07472 (2022)
16. Heist, N., Paulheim, H.: Nastylinker: nil-aware scalable transformer-based entity linker. In: The Semantic Web: 20th International Conference, ESWC 2023, Hersonissos, Crete, Greece, May 28-June 1, 2023, Proceedings, pp. 174–191. Springer-Verlag, Berlin, Heidelberg (2023). https://doi.org/10.1007/978-3-031-33455-9_11
17. Twitter 1 stream. Accessed 20 Aug 2023
18. Ilievski, F., Vossen, P., Schlobach, S.: Systematic study of long tail phenomena in entity linking. In: Proceedings of the 27th International Conference on Computational Linguistics, pp. 664–674 (2018)
19. Ju, Y., Adams, B., Janowicz, K., Hu, Y., Yan, B., McKenzie, G.: Things and strings: improving place name disambiguation from short texts by combining entity co-occurrence with topic modeling. In: Blomqvist, E., Ciancarini, P., Poggi, F., Vitali, F. (eds.) EKAW 2016. LNCS (LNAI), vol. 10024, pp. 353–367. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-49004-5_23
20. Liu, X., et al.: Entity linking for tweets. In: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 1304–1311 (2013)
21. Manaskasemsak, B., Netsiwawichian, N., Rungsawang, A.: Entity co-occurrence graph-based clustering for twitter event detection. In: Barolli, L. (ed.) Advanced Information Networking and Applications: Proceedings of the 38th International Conference on Advanced Information Networking and Applications (AINA-2024), Volume 2, pp. 344–355. Springer Nature Switzerland, Cham (2024). https://doi.org/10.1007/978-3-031-57853-3_29
22. Mendes, P.N., Jakob, M., García-Silva, A., Bizer, C.: Dbpedia spotlight: shedding light on the web of documents. In: Proceedings of the 7th International Conference on Semantic Systems, pp. 1–8 (2011)
23. Mishra, S., Saini, A., Makki, R., Mehta, S., Haghighi, A., Mollahosseini, A.: TweetNERD-end to end entity linking benchmark for tweets. Adv. Neural. Inf. Process. Syst. **35**, 1419–1433 (2022)
24. Noullet, K., Ourgani, A., Färber, M.: A full-fledged framework for combining entity linking systems and components. In: Proceedings of the 12th Knowledge Capture Conference 2023, pp. 148–156 (2023)
25. Pastrav, C., Dignum, F.: Norms in social simulation: balancing between realism and scalability. In: Verhagen, H., Borit, M., Bravo, G., Wijermans, N. (eds.) Advances

in Social Simulation: Looking in the Mirror, pp. 329–342. Springer International Publishing, Cham (2020). https://doi.org/10.1007/978-3-030-34127-5_32
26. Peixoto, T.P.: The graph-tool python library. figshare (2014). https://doi.org/10.6084/m9.figshare.1164194, http://figshare.com/articles/graph_tool/1164194
27. Ristoski, P., Lin, Z., Zhou, Q.: KG-ZESHEL: knowledge graph-enhanced zero-shot entity linking. In: Proceedings of the 11th Knowledge Capture Conference, pp. 49–56 (2021)
28. Sakor, A., Singh, K., Patel, A., Vidal, M.E.: FALCON 2.0: an entity and relation linking tool over Wikidata. In: Proceedings of the 29th ACM International Conference on Information & Knowledge Management, pp. 3141–3148 (2020)
29. Salavati, C., Abdollahpouri, A., Manbari, Z.: Ranking nodes in complex networks based on local structure and improving closeness centrality. Neurocomputing **336**, 36–45 (2019)
30. Sciarra, C., Chiarotti, G., Laio, F., Ridolfi, L.: A change of perspective in network centrality. Sci. Rep. **8**(1), 15269 (2018)
31. Sevgili, Ö., Shelmanov, A., Arkhipov, M., Panchenko, A., Biemann, C.: Neural entity linking: a survey of models based on deep learning. Semant. Web **13**(3), 527–570 (2022)
32. Van Erp, M., Groth, P.: Towards entity spaces. In: Proceedings of the Twelfth Language Resources and Evaluation Conference, pp. 2129–2137 (2020)
33. Van Erp, M., et al.: Evaluating entity linking: an analysis of current benchmark datasets and a roadmap for doing a better job. In: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), pp. 4373–4379 (2016)
34. Van Hulst, J.M., Hasibi, F., Dercksen, K., Balog, K., de Vries, A.P.: REL: an entity linker standing on the shoulders of giants. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 2197–2200 (2020)
35. Zadgaonkar, A., Agrawal, A.J.: An approach for analyzing unstructured text data using topic modeling techniques for efficient information extraction. New Gener. Comput., 1–26 (2023)