Unsupervised Named Entity Disambiguation for Low Resource Domains

Debarghya Datta and **Soumajit Pramanik**
Department of Computer Science
Indian Institute of Technology, Bhilai
{debarghyad,soumajit}@iitbhilai.ac.in

Abstract

In the ever-evolving landscape of natural language processing and information retrieval, the need for robust and domain-specific entity linking algorithms has become increasingly apparent. It is crucial in a considerable number of fields such as humanities, technical writing and biomedical sciences to enrich texts with semantics and discover more knowledge. The use of Named Entity Disambiguation (NED) in such domains requires handling noisy texts, low resource settings and domain-specific KBs. Existing approaches are mostly inappropriate for such scenarios, as they either depend on training data or are not flexible enough to work with domain-specific KBs. Thus in this work, we present an unsupervised approach leveraging the concept of Group Steiner Trees (GST), which can identify the most relevant candidates for entity disambiguation using the contextual similarities across candidate entities for all the mentions present in a document. We outperform the state-of-the-art unsupervised methods by more than 40% (in avg.) in terms of Precision@1 across various domain-specific datasets.

1 Introduction

Named Entity Disambiguation (NED) is the task of resolving the ambiguity associated with entity mentions in a document by linking them to the appropriate entries in a Knowledge Base (KB). Recently, NED has been applied in various fields, including digital humanities, art, architecture, literature, and biomedical science, for tasks such as searching (Meij et al., 2014), question answering (Yih et al., 2015) and information extraction (Nooralahzadeh and Øvrelid, 2018).

The key challenges in such domain specific NED tasks are twofold - (a) they provide little or no training data with ground truth annotations and (b) the associated knowledge graphs (KG) are typically small and with no or very limited entity descriptions (Shi et al., 2023). In order to deal with such challenges, in this work we consider the setting where entity disambiguation is needed to be performed with *absolute absence of annotated data*. In such constrained scenarios, leveraging the state-of-the-art neural entity linkers become infeasible as they are primarily dependent on a large corpus of annotated data and long enough entity descriptions from KG (Cadavid-Sánchez et al., 2023; Arora et al., 2021). Similarly this setting also disqualifies unsupervised NED approaches such as (Pan et al., 2015) which rely on labeled data to generate candidate entities such as domain-adaptive transformer-based models (Aydin et al., 2022), BLINK (Ledell Wu, 2020), Zeshel (Logeswaran et al., 2019), and auto-regressive models like GENRE (De Cao et al., 2021).

In the literature, only a few approaches fit our constrained setting such as graph-based using mention distances (Hoffart et al., 2011), PageRank/random walk based (Guo and Barbosa, 2018), and graph ranking based (Alhelbawy and Gaizauskas, 2014). A recent approach by (Arora et al., 2021) also explores singular value decomposition, showing gold entities in a low-rank subspace. However, these methods often struggle in achieving the required efficacy while disambiguating entities.

In this work, we present a novel unsupervised NED approach for domain specific low-resource scenarios, which leverages the concept of Group Steiner Trees (GSTs) (Garg et al., 2000). In this approach, we map the candidate entities for each mention in the document, to nodes in the associated knowledge graph, obtain the subgraph connecting these nodes and then extract minimum cost GSTs from this sub-graph. Such GSTs facilitate collective entity disambiguation exploiting the fact that the entities that are truly mentioned in a document (the 'gold entities') tend to form a dense subgraph among the set of all candidate entities in the document.

Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 14922–14928
November 12-16, 2024 ©2024 Association for Computational Linguistics

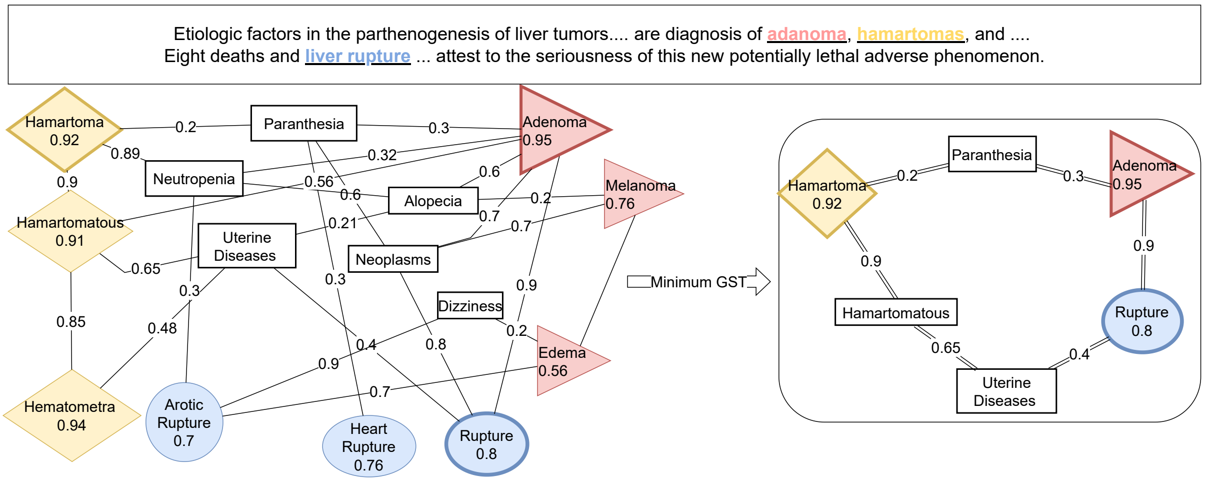

Figure 1: Proposed GST-NED approach: The sample document at the top contains three mentions; the subgraph extracted from the KB is shown at the left and the minimum cost GST is shown using the box at the right. In the induced subgraph, the candidates for 'adenoma' are marked in red, 'hamartomas' are marked in yellow and 'liver rupture' are marked in blue.

In summary, our main contributions are the following - (a) We propose an unsupervised **G**roup **S**teiner **T**ree based **N**amed **E**ntity **D**isambiguation (*GST-NED*) method which is capable to perform NED for low resource domains at the absence of any annotated data; (b) We compare our proposed approach with several state-of-the-art baselines across multiple domain specific datasets and demonstrate its superior performance with significant improvements in the metrics (more than 40% in avg. in Precision@1 scores) [1].

2 Problem Statement

Similar to most previous works in the NED literature (with a few exceptions (Kolitsas et al., 2018; Sil and Yates, 2013)), we assume that document-wise mention spans (usually obtained by a named entity recognizer) are already provided. Let d be a single document from a collection D of documents. Also, let $M_d = \{m_1, m_2, \ldots, m_M\}$ be the set of M mentions contained in d, and let $\mathcal{E}$ be the collection of all the entities contained in the reference domain specific Knowledge Graph KG. The task here is to find, for each mention m_i the correct entity $e \in \mathcal{E}$ it refers to.

Typically, given the set of mentions, an NED approach performs the disambiguation in two steps - (a) Candidate generation, where candidate entities from the KG are retrieved for each of the mentions, and (b) Candidate Ranking, where the candidate entities are ranked based on their propensities to be mapped with the corresponding mentions. Our primary focus in this study is the candidate ranking/disambiguation step. In the following, we describe our proposed candidate ranking method and mention the approaches adhered for the other step.

3 Methodology

Candidate Generation

We index the domain specific KG and use fuzzy text search (Bachmann, 2021) to retrieve candidates based on the surface form of the annotated mention. This is found to be the standard practice in most of the recent unsupervised NED approaches (Yang et al., 2023; Simos and Makris, 2022) Fuzzy text search returns a confidence value with each potential match; we keep only the candidates which are returned with more than 0.75 confidence value (chosen empirically) [2].

Candidate Ranking

We use the knowledge graph (KG) to create a subgraph connecting all pairs of candidate entities obtained from the candidate generation step for a particular document d. To keep the graph size manageable, we limit path lengths to be a maximum of three hops between entity candidates. We further

[1] Code is available at https://github.com/deba-iitbh/GST-NED

[2] In case of exact match with a KG node, we consider it to be the correct match for the mention and skip the candidate ranking step.

enhance the graph by adding node weights based on the Jaro-Winkler distance (Wang et al., 2017) (reflecting similarities of candidates with mentions), and edge weights based on cosine similarities of Node2Vec (Grover and Leskovec, 2016) structural embeddings of the endpoints. In Figure 1, we depict a document with three mentions and the corresponding induced subgraph of candidate entities (left side).

Finding GST: Our approach to identify the correct candidates relies on the intuition that a gold entity candidate from a document d should be more tightly connected with other gold candidates in the induced subgraph compared to other non-gold candidates. In other words, we expect the gold entities within the induced subgraph to form cohesive and closely linked subgraphs due to their contextual proximities (as they are used in the same document). In order to exploit this intuition, we first define the notion of terminals - for every mention m_i, we denote the corresponding candidate entity nodes as the terminal nodes for that mention and group them together as T_i. Further the task remains is to select the correct candidate node from each terminal group for which we leverage the concept of Group Steiner Trees (GST) (Ding et al., 2006; Pramanik et al., 2024) as defined below,

- Given an undirected and weighted graph (V, E) and given groups of terminal nodes $\{T_1, ..., T_l\}$ with each $T_\nu \subseteq V$, compute the minimum-cost tree (V^*, E^*) that connects at least one node from each of $\{T_1, ..., T_l\}$: $\min \sum_{ij \in E^*} c_{ij}$ s.t. $T_\nu \cap V^* \neq \emptyset$, $\forall T_\nu$.

In our case, we consider $c_{ij} = (1 - w_{ij})$ where w_{ij} represents the edge weight between nodes i and j. As per definition, each GST would have to necessarily choose at least one candidate entity from each of the terminal groups. Hence, each detected GST would provide at least one potential solution to the entity disambiguation problem. As we further posit that the gold candidate entities are more tightly connected compared to non-gold candidates, the probability of the gold candidates to be chosen in the minimum cost GST increases (as the minimum cost GST ensures shorter distances between the chosen candidates and higher weighted edges i.e. lower edge-costs). For instance, in the right side of the Fig. 1, we depict that the minimum cost GST extracted from the induced subgraph contains all the gold candidate entities corresponding to the mentions in the document.

Relaxation to GST-k and Ranking Criteria: In our setting, we actually look for the entity candidates extracted from k least cost GSTs (used k=10 for our work empirically) rather than relying upon only the minimum cost GST. This is for enhancing the robustness of the approach as it allows us to rank the different candidate entities efficiently. We utilize the following three intutive ranking schemes to rank the candidate entities for each mention and choose the higher ranked one - **(a) GST count**: Number of GSTs where the candidate is present; the higher the better, **(b) GST Cost**: Total cost of the GSTs where the candidate is present; the lower the better, and **(c) Node Weight:** The sum of node weights in the GSTs where the candidate is present; the higher the better. Subsequently, we compare the performance of all three schemes to choose the best one.

Complexity: Steiner trees are among the classical NP-complete problems (Ding et al., 2006), and this holds for the GST problem too. However, the problem has tractable fixed-parameter complexity when the number of terminals is treated as a constant (Downey et al., 2013), and there are also good polynomial-time approximation algorithms extensively applied in the area of keyword search over databases (Ding et al., 2006; Kacholia et al., 2005; Li et al., 2016b). In *GST-NED*, we build on the exact solution method by (Ding et al., 2006), which uses a dynamic programming approach and has exponential runtime in the number of mentions (which is typically limited) but has $O(n \log n)$ complexity in the graph size.

4 Experimental Setup

Datasets

In order to show the efficacy of our model, we choose the following four datasets from diverse domains of literature, law, museum artifacts and chemicals (see Table. 1 for more details).

WWO[3] is a collection of textual documents (poems, plays and novels) by pre-Victorian women writers, partially annotated (Flanders and Melson, 2010) with person, works and places entities.

1641[4] consists of legal texts in the form of court witness statements recorded after the Irish Rebellion of 1641, partially annotated with person names against a subset of DBpedia KB (Klie et al., 2020).

[3] https://www.wwp.northeastern.edu/wwo

[4] http://1641.tcd.ie/

Dataset	#D	#M	#N	#E	#C	#R
WWO	76	14651	9065	4936	10	0.83
1641	16	480	3503	338	10	0.26
Artifact	168	6311	41180	42634	11	0.66
Chemical	135	15769	176415	249275	10	0.73

Table 1: Data statistics of the four used datasets: Total number of Documents ($\#D$), Total number of mentions ($\#M$), Number of Nodes ($\#N$) and Edges ($\#E$) in KG, Average number of candidates per mention ($\#C$) and Recall of the candidate entities i.e fraction of mentions with gold entities present among the candidates ($\#R$) .

Chemical dataset is sourced from the BC5CDR corpus (Li et al., 2016a). It features a comprehensive human annotations of chemicals, each tagged with unique MeSH identifiers. For the categorization of chemicals, the Chemicals vocabulary is sourced from the Comparative Toxicogenomics Database (CTD) [5].

Artifact (Cadavid-Sánchez et al., 2023) is a collection of digital descriptions of Museum objects annotated with four different text fields: title, detailed description, free-form metadata against the Getty Arts, and Architecture Thesaurus [6](AAT).

Baselines

To compare the performance of our proposed approach, we leverage the following baselines [7].

NameMatch(Klie et al., 2020). We employ a string-matching approach to select candidates that exactly match the surface form of the mention.

BLINK* (Ledell Wu, 2020). We adapt a fine-tuned BLINK model in our domain specific setup for predicting named entities for each mention. As it matches entities to Wikipedia[8] by default, we subsequently perform a fuzzy matching process to align the predicted entities with our domain specific knowledge base.

WalkingNED (Guo and Barbosa, 2018) is a graph-based approach to disambiguate the mention candidates, based on local similarity (surface form similarity) and global similarity (similarity between the semantic signatures of the candidate and the document computed using PageRank).

Eigenthemes (Arora et al., 2021) is an approach which leverages the inherent property of 'gold entities' to cluster together within the embedding space by representing entities as vectors and utilizing Singular Value Decomposition (SVD).

Dataset	Model	P@1	HIT@5
WWO	NameMatch	0.35	0.35
	BLINK*	0.07	0.09
	WalkingNED	0.18	0.49
	EigenThemes	0.14	0.45
	GST-NED	**0.57**	**0.72**
1641	NameMatch	0.06	0.06
	BLINK*	0.05	0.11
	WalkingNED	0.11	0.17
	EigenThemes	0.17	**0.25**
	GST-NED	**0.20**	0.22
Artifact	NameMatch	0.23	0.23
	BLINK*	0.02	0.03
	WalkingNED	0.26	0.56
	EigenThemes	0.15	0.44
	GST-NED	**0.54**	**0.61**
Chemical	NameMatch	0.08	0.08
	BLINK*	0.13	0.22
	WalkingNED	0.50	**0.66**
	EigenThemes	0.36	0.59
	GST-NED	**0.52**	**0.66**

Table 2: NED performance comparison for WWO, 1641, Artifact and Chemical datasets.

Metrics

Similar to the state-of-the-art literature in NED, we use Precision@1 (correctness of top ranked candidate) and Hit@5 (presence of gold entity in top five ranked candidate) as our evaluation metrics.

5 Results and Discussion

We compared our proposed *GST-NED* approach with other baselines algorithms and the corresponding results are depicted in Table. 2. We can observe that our method outperforms the state-of-the-art in all the datasets (especially in terms of P@1). In 1641, the relatively poor performance of all the algorithms stems from the poor recall of the candidate entities (see Table. 1). BLINK* in general works poorly as it struggles to find a suitable match in the domain specific knowledge bases.

Analysing Ranking Schemes In Table. 3, we analyse the impact of choosing different ranking schemes for candidate ranking in *GST-NED*. It is observed that the GST-count scheme performs the best in our scenario.

[5] https://www.ctdbase.org/

[6] https://www.getty.edu/research/tools/vocabularies/aat/about.html

[7] All the Datasets and Baseline codes are available under MIT & Apache License.

[8] https://www.wikipedia.org/

	WWO	Artifact
GST count	**0.57**	**0.54**
GST cost	0.55	0.51
Node weight	0.54	0.53

Table 3: Comparison over ranking schemes (P@1) on two datasets

Parameter Fine-tuning In order to optimize the metric values, we conduct extensive empirical experiments with varying fuzzy threshold values for candidate generation and different numbers of top-ranked GSTs (k) for candidate ranking. These experiments are performed on a small held-out subset (10%) of the 'WWO' and 'Artifact' datasets, with results presented in Table. 4, 5. Based on our analysis, considering the fuzzy threshold value of 0.75 and top-10 GSTs yield the highest Precision@1 score for our setup. Consequently, these parameters are used for all the experiments reported in this work.

Threshold	WWO	Artifact
0.70	0.632	0.574
0.75	0.634	0.580
0.80	0.632	0.562
0.85	0.631	0.554
0.90	0.633	0.554

Table 4: Precision@1 for held-out WWO and Artifact datasets with various Fuzzy Matching thresholds

k	WWO	Artifact
1	0.63	0.55
5	0.63	0.57
10	0.64	0.58
20	0.62	0.56
50	0.63	0.56

Table 5: Precision@1 for held-out WWO and Artifact datasets at various k (number of top ranked GST) values

Error Analysis We conduct a detailed error analysis to identify the distribution of errors in our proposed pipeline. Specifically, we compute the proportion of instances where error occurs due to: (a) the gold (correct) entity not being present in the candidate list, (b) the gold entity being present in the candidate list but not in the top-k GSTs, and (c) the gold entity being included in the top-k GSTs but does not rank in the top-1 position. On the 'WWO' dataset, 14% of errors corresponded to (a), 11% to (b), and 18% to (c), while the remaining 57% of cases were correctly resolved, resulting in a precision@1 score of 0.57. These findings suggest that enhancing both the ranking mechanism and candidate generation process are critical for achieving improved performance.

6 Conclusion

In this paper, we have addressed the problem of NED of domain-specific corpora in the absence of annotated data. It works based on the intuition that a gold entity candidate from a document should be more cohesively connected with other gold candidates in the knowledge graph compared to other non-gold candidates. We have leveraged the concept of Group Steiner Trees (GSTs), that relies solely on the availability of candidate entity names and a domain specific knowledge graph. Extraction of minimum cost GSTs in our proposed approach *GST-NED*, ensures that the chosen entities are closely connected in the domain specific knowledge graphs. Experiments on benchmark datasets from varied domains have portrayed the effectiveness of our proposed approach against the state-of-the art unsupervised and zero-shot approaches.

Limitations

Our entity disambiguation method, *GST-NED*, depends on the presence of sufficient number of entities per document to function accurately as we rely upon joint disambiguation of entities. As a result, when the entity count is very low, it fails to provide the correct response. On the other hand, considering relatively longer document chunks with too many entities increases the graph size, affecting our computational efficacy. Hence, it is essential to analyze this trade-of with a detailed and thorough study. Interestingly, considering longer documents also enhances the possibility of same mention being used multiple times with different meanings which is beyond the capability of our model for the time being. Additionally, further works need to be done to improve the scalability of the Steiner tree algorithm we use to compute the optimal trees. Presently it takes around 2 seconds per document for small KGs like WWO, 1641 or Artifact and around 40 seconds per document on the relatively larger KG of Chemical dataset (on a system with 3.9GHz CPU with 16 GB RAM).

Ethics

The data and models in this work are publicly available. They could contain bias, and should be used with discretion.

References

Ayman Alhelbawy and Robert Gaizauskas. 2014. Graph ranking for collective named entity disambiguation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 75–80.

Akhil Arora, Alberto García-Durán, and Robert West. 2021. Low-rank subspaces for unsupervised entity linking. *arXiv preprint arXiv:2104.08737*.

Gizem Aydin, Seyed Amin Tabatabaei, George Tsatsaronis, and Faegheh Hasibi. 2022. Find the funding: Entity linking with incomplete funding knowledge bases. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 1937–1942, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Max Bachmann. 2021. maxbachmann/rapidfuzz: Release 1.8.0.

Sebastián Cadavid-Sánchez, Khalil Kacem, Rafael Aparecido Martins Frade, Johannes Boehm, Thomas Chaney, Danial Lashkari, and Daniel Simig. 2023. Evaluating end-to-end entity linking on domain-specific knowledge bases: Learning about ancient technologies from museum collections. *ArXiv*, abs/2305.14588.

Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. 2021. Autoregressive entity retrieval. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Bolin Ding, Jeffrey Xu Yu, Shan Wang, Lu Qin, Xiao Zhang, and Xuemin Lin. 2006. Finding top-k min-cost connected trees in databases. In *2007 IEEE 23rd international conference on data engineering*, pages 836–845. IEEE.

Rodney G Downey, Michael R Fellows, et al. 2013. *Fundamentals of parameterized complexity*, volume 4. Springer.

Julia Hammond Flanders and John Melson. 2010. Encoding names for contextual exploration in digital thematic research collections.

Naveen Garg, Goran Konjevod, and Ramamoorthi Ravi. 2000. A polylogarithmic approximation algorithm for the group steiner tree problem. *Journal of Algorithms*, 37(1):66–84.

Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 855–864.

Zhaochen Guo and Denilson Barbosa. 2018. Robust named entity disambiguation with random walks. *Semantic Web*, 9(4):459–479.

Johannes Hoffart, Mohamed Amir Yosef, Ilaria Bordino, Hagen Fürstenau, Manfred Pinkal, Marc Spaniol, Bilyana Taneva, Stefan Thater, and Gerhard Weikum. 2011. Robust disambiguation of named entities in text. In *Proceedings of the 2011 conference on empirical methods in natural language processing*, pages 782–792.

Varun Kacholia, Shashank Pandit, S Sudarshan, Rushi Desai, and Hrishikesh Karambelkar. 2005. Bidirectional expansion for keyword search on graph databases.

Jan-Christoph Klie, Richard Eckart de Castilho, and Iryna Gurevych. 2020. From zero to hero: Human-in-the-loop entity linking in low resource domains. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 6982–6993.

Nikolaos Kolitsas, Octavian-Eugen Ganea, and Thomas Hofmann. 2018. End-to-end neural entity linking. *arXiv preprint arXiv:1808.07699*.

Martin Josifoski Sebastian Riedel Luke Zettlemoyer Ledell Wu, Fabio Petroni. 2020. Zero-shot entity linking with dense entity retrieval. In *EMNLP*.

Jiao Li, Yueping Sun, Robin J Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J Mattingly, Thomas C Wiegers, and Zhiyong Lu. 2016a. Biocreative v cdr task corpus: a resource for chemical disease relation extraction. *Database*, 2016.

Rong-Hua Li, Lu Qin, Jeffrey Xu Yu, and Rui Mao. 2016b. Efficient and progressive group steiner tree search. In *Proceedings of the 2016 International Conference on Management of Data*, pages 91–106.

Lajanugen Logeswaran, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, Jacob Devlin, and Honglak Lee. 2019. Zero-shot entity linking by reading entity descriptions. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3449–3460, Florence, Italy. Association for Computational Linguistics.

Edgar Meij, Krisztian Balog, and Daan Odijk. 2014. Entity linking and retrieval for semantic search. *WSDM*, 10:2556195–2556201.

Farhad Nooralahzadeh and Lilja Øvrelid. 2018. SIRIUS-LTG: An entity linking approach to fact extraction and verification. In *Proceedings of the First Workshop on Fact Extraction and VERification*

(FEVER), pages 119–123, Brussels, Belgium. Association for Computational Linguistics.

Xiaoman Pan, Taylor Cassidy, Ulf Hermjakob, Heng Ji, and Kevin Knight. 2015. Unsupervised entity linking with abstract meaning representation. In *Proceedings of the 2015 conference of the north american chapter of the association for computational linguistics: Human language technologies*, pages 1130–1139.

Soumajit Pramanik, Jesujoba Alabi, Rishiraj Saha Roy, and Gerhard Weikum. 2024. Uniqorn: unified question answering over rdf knowledge graphs and natural language text. *Journal of Web Semantics*, page 100833.

Jiyun Shi, Zhimeng Yuan, Wenxuan Guo, Chen Ma, Jiehao Chen, and Meihui Zhang. 2023. Knowledge-graph-enabled biomedical entity linking: a survey. *World Wide Web*, pages 1–30.

Avirup Sil and Alexander Yates. 2013. Re-ranking for joint named-entity recognition and linking. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 2369–2374.

Michael Angelos Simos and Christos Makris. 2022. Computationally efficient context-free named entity disambiguation with wikipedia. *Information*, 13(8):367.

Yaoshu Wang, Jianbin Qin, and Wei Wang. 2017. Efficient approximate entity matching using jaro-winkler distance. In *International conference on web information systems engineering*, pages 231–239. Springer.

Siyu Yang, Peiliang Zhang, Chao Che, and Zhaoqian Zhong. 2023. B-lbcona: a medical entity disambiguation model based on bio-linkbert and context-aware mechanism. *BMC bioinformatics*, 24(1):97.

Wen-tau Yih, Ming-Wei Chang, Xiaodong He, and Jianfeng Gao. 2015. Semantic parsing via staged query graph generation: Question answering with knowledge base. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1321–1331, Beijing, China. Association for Computational Linguistics.