Predicting Guiding Entities for Entity Aspect Linking

Shubham Chatterjee
shubham.chatterjee@unh.edu
University of New Hampshire
Durham, New Hampshire, USA

Laura Dietz
dietz@cs.unh.edu
University of New Hampshire
Durham, New Hampshire, USA

ABSTRACT

Entity linking can disambiguate mentions of an entity in text. However, there are many different aspects of an entity that could be discussed but are not differentiable by entity links, for example, the entity "oyster" in the context of "food" or "ecosystems". Entity *aspect* linking provides such fine-grained explicit semantics for entity links by identifying the most relevant aspect of an entity in the given context. We propose a novel entity aspect linking approach that outperforms several neural and non-neural baselines on a large-scale entity aspect linking test collection. Our approach uses a supervised neural entity ranking system to predict relevant entities for the context. These entities are then used to guide the system to the correct aspect.

ACM Reference Format:
Shubham Chatterjee and Laura Dietz. 2022. Predicting Guiding Entities for Entity Aspect Linking. In *Proceedings of the 31st ACM International Conference on Information and Knowledge Management (CIKM '22), October 17–21, 2022, Atlanta, GA, USA.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3511808.3557671

1 INTRODUCTION

Entity-oriented search systems leverage semantic information about entities for document ranking [9, 13, 24, 41, 50–52]. Often, an important component of such systems is entity linking. Entity linking [14, 28, 36] disambiguates the mentions of an entity from the context in text. For example, in Figure 1, the entity link can discern that "oyster" refers to the animal and not to the town. Our goal is to further refine this information using entity *aspect* links that identify the aspect of "oyster" (e.g., "food" or "ecosystem service") referenced in this context. In this work, we address this task:

Entity Aspect Linking Task. Given (1) the mention of an entity in a context (e.g., sentence), and (2) a set of predefined aspects with their contents (text and entity links), link the mention to the entity's aspect that best captures the contextual reference. Following previous work [32, 40], we consider the top-level sections from an entity's Wikipedia page as the entity's aspects.

Entity aspect links have been shown to be useful for text analysis tasks [32]. They have also been shown to be useful for entity ranking [4, 5]. In this work, we investigate the opposite perspective: *Can we leverage entity ranking for the entity aspect linking task?*

CIKM '22, October 17–21, 2022, Atlanta, GA, USA

ACM ISBN 978-1-4503-9236-5/22/10…$15.00
https://doi.org/10.1145/3511808.3557671

Previous approaches to entity aspect linking [32, 40] identify the correct aspect by matching the logical ids of entities (identified via entity linking) found in the context and aspect. The issue here is that often, few entities are shared between the context and the correct aspect (see RQ4 in Section 5). Alternatively, one may use an entity relatedness measure such as the cosine similarity of entity embeddings [2, 37, 55] to identify the correct aspect. However, this approach is oblivious to the link context critical for the task.

Contributions. In this work[1], we propose an alternative, yet complementary approach to entity aspect linking. At the heart of our approach is a supervised neural entity ranking model that, given the context, will predict entities contained in the correct aspect's content (henceforth *guiding entities*). Our approach alleviates the issue with entity matching found in previous work on entity aspect linking. We show that our approach to entity aspect linking can outperform several strong baselines, both neural and non-neural, on a large-scale entity aspect linking test collection. We further demonstrate that our approach to ranking guiding entities for the context plays a significant role in this performance improvement.

2 RELATED WORK

Sections as Aspects. Fetahu et al. [15] enrich Wikipedia sections with news-article references by considering the top-level sections as sub-topics. Reinanda et al. [42] present a method for document filtering for long-tail entities using the top-level sections from Wikipedia as entity aspects. Nanni et al. [32] consider top-level sections as aspects and present a learning-to-rank method for the entity aspect linking task using lexical and semantic features derived from the context. Ramsdell et al. [40] released a test collection for entity aspect linking along with strong baselines whereas Hayashi et al. [20] released a dataset for multi-domain aspect-based summarization. Both datasets consider top-level sections from Wikipedia as entity aspects. We too consider the top-level sections from Wikipedia as aspects, and use the dataset from Ramsdell et al. in this work.

Text Similarity. The entity aspect linking task may be addressed by learning the similarity between the texts of the aspect and the context. Often, text similarity is learnt by using BERT [10] to learn embeddings of the two texts such that the cosine distance of the embeddings is minimized. The similarity may also be learnt using a fully-connected layer trained jointly with the model [27]. Alternatively, pre-trained text embedding methods such as Word2Vec [30] or GloVe [35] may be used to create embeddings of the two texts for use with cosine similarity. We include a baseline that learns embeddings of the context and aspect using BERT, then learns the similarity between them using a fully-connected layer.

Document Retrieval. One may consider the context as a query and the aspect as a document to be retrieved. Hence, a related

[1]Code and data available at: https://github.com/shubham526/CIKM-2022-EAL

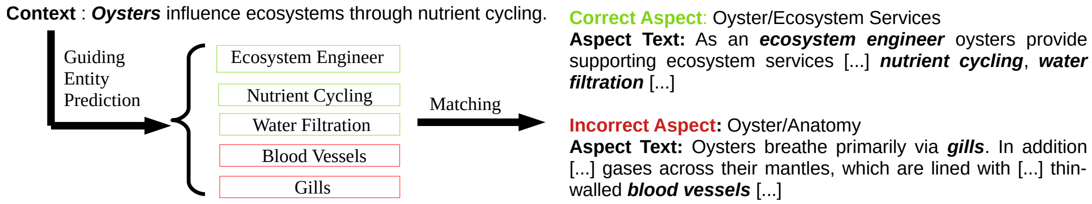

Figure 1: Our entity aspect linking approach. Correct entities/aspects in green, incorrect in red. First, we predict context-relevant entities to guide us to the correct aspect. Then, we rank aspects by the number of guiding entities they mention. Here, the aspect "Oyster/Ecosystem Services" (true aspect) is ranked highest as it mentions many top context-relevant guiding entities.

task is document retrieval. Some traditional, term-based document retrieval models are BM25 [21], Language Models [38], and TF-IDF [45, 48]. Recently, much work has been done in the area of Neural-IR [8, 16, 31, 34, 47, 53]. In this work, we include the following Neural-IR models as baselines: KNRM [53] and Conv-KNRM [7]. KNRM is a kernel-based neural ranking model that uses a translation matrix to model the word-level similarities between the query and document. To score documents, features extracted from this matrix are combined using a learning-to-rank layer. Instead of a translation matrix, Conv-KNRM uses Convolutional Neural Networks to learn n-gram features from queries and documents.

Entity Ranking. Often, entity ranking systems first construct term-based entity descriptions using the introductory paragraph from an entity's Wikipedia page (henceforth *lead text*) [25, 27, 54], then rank these descriptions using document retrieval models (e.g., BM25). The issue here is that often, the lead text contains only the most popular knowledge about the entity which may not even be relevant for the query. Hence, a model using the lead text would be unable to understand the relevant connections between the query and entity to determine the entity's relevance. In this work, we provide a model that leverages a better entity description to determine the entity's relevance for the query. For comparison, we include a BERT-based model that uses the lead text to rank entities.

Other approaches to entity ranking include using the Markov Random Field to model term dependencies [19, 29, 33, 57], leveraging types [1, 17] and relationships [3, 6] in Knowledge Bases to rank entities, and Learning-To-Rank by representing each query-entity pair as a feature vector [4, 11, 46]. Recently, the utility of entity embeddings for entity ranking has also been explored [5, 18].

Entity Relatedness. Entity relatedness measures the degree to which two entities are similar. Many entity relatedness measures have been developed [39, 44, 49, 56], based on proximity of entities in the knowledge graph, the number of in-links and out-links, etc. Entity relatedness may also be measured using the cosine similarity of the entity embeddings obtained using a graph embedding method [2, 37, 43, 55]. The entity aspect linking task can be addressed as a semantic similarity task based on the relatedness of the entities in the context and aspect. Hence, we include semantic baselines based on Wikipedia2Vec [55] and E-BERT [37].

3 APPROACH

We propose an easy-to-implement and novel method for the entity aspect linking task that given a context, first uses a neural entity ranking model to predict guiding entities from the correct aspect. This model is trained using a ground truth of entities linked to the correct aspect. We leverage this guiding entity ranking to rank aspects based on the number of context-relevant guiding entities mentioned in an aspect. Finally, using learning-to-rank, we combine this entity-guide-based aspect ranking with other aspect rankings obtained using lexical similarity measures such as BM25.

Entity ranking for entity aspect linking. As described in Section 2, entity ranking systems often use the lead text of an entity as the entity's description. The issue is that the lead text is query-agnostic (here, query = context). For example, the lead text[2] of the entity "water filtration" (Figure 1) does not even mention oysters or their connection to water filtration. Hence, a model using the lead text would be unable to recover this relevant connection. We provide a model that is capable of understanding the connection between an entity and the context to determine whether that entity is relevant. At its core, our method leverages a better entity description that elaborates on the connection of an entity to the context. Concretely, given the context, we identify a *context-specific* description of an entity e from an aspect via Pseudo-Relevance Feedback [23]: We retrieve a candidate set of paragraphs from a search index with BM25 using the context, then use the highest ranked paragraph that mentions the entity e as the entity's description $\mathcal{D}$.

We use BERT [10] to rank entities e using entity descriptions $\mathcal{D}$ and the context. The input to BERT is a sequence of context and description tokens separated by the special tokens *[CLS]* and *[SEP]*. We learn the score for an entity by passing the embedding of the *[CLS]* token obtained from the last hidden layer of BERT through a fully-connected layer trained jointly with the model. The model is trained using a ground truth of entities derived from the entity aspect linking dataset (Section 4).

Entity aspect ranking. We consider the top-k entities[3] from the entity ranking obtained above as guides to the correct aspect. We rank aspects by the number of times they mention a guiding entity from the top-k of the entity ranking. We treat this entity-guide-based aspect ranking as a feature; then using list-wise Learning-To-Rank, we combine this ranking with (lexical) aspect rankings obtained using lexical similarity measures found to be useful in previous work [32, 40]: (1) BM25, (2) TF-IDF, and (3) word overlap.

4 EXPERIMENTAL METHODOLOGY

Entity aspect linking data. We use the large-scale entity aspect linking (EAL) dataset from Ramsdell et al. [40]. The dataset is based

[2] https://en.wikipedia.org/wiki/Water_filter
[3] We empirically choose $k = 100$ in this work.

on an English Wikipedia dump (January 2020) and considers the top-level sections from Wikipedia as entity aspects. It contains an aspect catalog with the name (section heading) and content (text and entity links) of each aspect. Different train, validation, and test sets are also provided. In this work, we use the *Train-Small* partition (5498 EAL instances covering 1000 entities) for training our model. For testing, we use the following two partitions: (1) *Nanni-Test* containing 18,289 EAL instances covering 201 entities in the dataset from Nanni et al. [32], and (2) *Test* containing 4967 EAL instances covering 1000 entities (not including those in *Nanni-Test*). The dataset contains both sentence and paragraph as contexts. In this work, we use the sentence context as previous work [32] has shown it to be superior to the paragraph context.

Entity ranking data. We derive an entity ranking dataset from the EAL dataset: The train and test data are derived from the corresponding train and test partitions in the EAL dataset. We derive a ground truth of entities as follows: All entities in the true aspect are relevant for the context whereas all other entities in all other candidate aspects are non-relevant. We use the sentence context as query. As documents, we use the top-ranked pseudo-relevant paragraph retrieved for every candidate entity. We use the collection of English Wikipedia paragraphs available with the TREC Complex Answer Retrieval [12] dataset.

Our systems. We refer to our proposed entity ranking method as **BERT (PRF-Psg)**, to our derived aspect ranking as **EntAsp**, and to our Learning-To-Rank combination of EntAsp with the lexical features from previous work as **LTR-EntLex**.

Baselines. We compare our systems to the following baselines. All neural models fine-tuned using pairwise margin ranking loss.[4]

(1) **Benchmark [40].** Benchmark results provided by Ramsdell.
(2) **LTR-Lex.** A list-wise Learning-To-Rank combination of the aspect rankings obtained using lexical similarity measures.
(3) **BERT [10].** BERT model fine-tuned directly for aspect ranking. We use the context as query and aspect as document.
(4) **KNRM [53].** Kernel based neural model for document ranking. Model fine-tuned for aspect ranking as in (3) above. We use the implementation from OpenMatch [26].
(5) **Conv-KNRM [8].** Convolutional kernel-based neural ranking model that models n-gram soft matches for ad-hoc search. Model fine-tuned for aspect ranking as in (3) above. We use the implementation from OpenMatch
(6) **EntityMatch (Overlap) [32].** Exact entity matching using logical ids of entities in the aspect and context.
(7) **EntityMatch (BM25) [32].** Exact entity matching using BM25 on a bag-of-entity representation of context and aspect.

Details of BERT fine-tuning for entity ranking. Our entity ranking model is implemented in PyTorch using HuggingFace and based on the `bert-base-uncased` version of BERT. We use the pairwise margin ranking loss to fine-tune our model using the Adam [22] optimizer, learning rate of $2e-5$, and batch size of 8.

Details of Learning-To-Rank (LTR). We use Coordinate Ascent optimized for Mean Average Precision with 5-fold cross-validation.

Evaluation metrics. Mean Average Precision (MAP), Precision at 1 (P@1) and Normalized Discounted Cumulative Gain at cut-off 20 (NDCG@20). Significance testing using paired-t-tests.

[4]https://pytorch.org/docs/stable/generated/torch.nn.MarginRankingLoss.html

Token (Railway Signalling)/Token Systems

Context: During rationalisation, on 30 June 1987, the ***staff and ticket*** between Korong Vale and ***Charlton/Wycheproof*** was replaced with ***electric staff***.

Entity Mention: staff and ticket
True Aspect: Token (Railway Signalling)/Token Systems
The token system was developed in ***Britain*** [...] following the ***Armagh rail disaster*** of 1889, ***block working*** became mandatory. Each single-line section is provided with a pair of token instruments, one at the ***signal box*** [...] In the ***Abermule train collision*** of 1921 [...]

Figure 2: Example context with entity mention helped by our approach. Entities in bold italics. We find that due to the small length of the context, *LTR-Lex* that is based on word-overlap fails to identify the correct aspect for this example as there are no overlapping words in the context and aspect. However, our entity ranking method *BERT (PRF-Psg)* ranks many entities from the true aspect (e.g., "Armagh Rail Disaster", "Abermule Train Collision", etc.) among the top-10 relevant entities for the given context. These top relevant entities guide us to the correct aspect.

5 RESULTS AND DISCUSSIONS

RQ1: How does the proposed approach compare to other state-of-the-art methods for the task? From Table 1, we observe that our proposed approach outperforms all baselines in terms of all evaluation measures on both Test and Nanni-Test. For example, on Test, the best performing baseline *LTR-Lex* obtains P@1 $=$ 0.64 whereas our system *LTR-EntLex* obtains P@1 $=$ 0.89 (39% improvement). We also outperform all neural ranking methods.

RQ2: Why does *LTR-EntLex* obtain better performance than the state-of-the-art in EAL? Does the proposed entity ranking method *BERT (PRF-Psg)* contribute to this performance improvement? We analyze the results for every entity mention in Test. We find that *LTR-Lex* cannot identify the correct aspect for 1428 entity mentions; however, our system *LTR-EntLex* helps these entity mentions by identifying the correct aspect. In contrast, the best neural baseline *KNRM* helps 404 mentions whereas the best semantic baseline *EntityMatch (BM25)* helps 395 mentions. One example entity mention helped by *LTR-EntLex* is shown in Figure 2. From the example, we see that *BERT (PRF-Psg)* helps our system *LTR-EntLex* and plays an important role in the performance improvements we achieve. We elaborate on this in RQ3 below.

RQ3: Does the overall performance change if we replace *BERT (PRF-Psg)* with an alternative entity ranking method? We replace *BERT (PRF-Psg)* with the following alternative methods:

- **Relatedness.** Entities e are ranked for context C as follows: $\text{Score}(e, C) = \sum_{E \in C} \text{Relatedness}(e, E)$, where E is an entity from context. To calculate $\text{Relatedness}(e, E)$, we use the cosine similarity between the embeddings of e and E obtained using Wikipedia2Vec [55] and E-BERT [37].
- **BERT (LeadText).** We rank guiding entities by fine-tuning BERT using the lead text of entities instead of the BM25 passages.

The results from our experiments are shown in Table 2. We observe that the aspect ranking derived from our proposed entity

Table 1: Results for entity aspect linking. Train data: *Train-Small*. Test data: *Test* and *Nanni-Test*. ▲ denotes significant improvement and ▼ denotes significant deterioration compared to *Benchmark* (denoted ⋆) using a paired-t-test at $p < 0.05$.

		Test			Nanni-Test		
		P@1	MAP	NDCG@20	P@1	MAP	NDCG@20
Retrieval Baselines	Benchmark [40]⋆	0.62⋆	0.77⋆	0.82⋆	0.66⋆	0.80⋆	0.85⋆
	LTR-Lex	0.64	0.78	0.83	0.73▲	0.84▲	0.88▲
Neural Baselines	BERT [10]	0.37▼	0.58▼	0.68▼	0.34▼	0.55▼	0.66▼
	KNRM [53]	0.41▼	0.62▼	0.71▼	0.40▼	0.61▼	0.70▼
	Conv-KNRM [8]	0.35▼	0.57▼	0.67▼	0.19▼	0.45▼	0.58▼
Semantic Baselines	EntityMatch (Overlap) [32]	0.31▼	0.53▼	0.64▼	0.22▼	0.47▼	0.59▼
	EntityMatch (BM25) [32]	0.34▼	0.56▼	0.66▼	0.49▼	0.66▼	0.74▼
Our Systems	EntAsp	0.78▲	0.83▲	0.84▲	0.72▲	0.73▼	0.73▼
	LTR-EntLex	**0.89**▲	**0.94**▲	**0.95**▲	**0.89**▲	**0.93**▲	**0.95**▲

Table 2: Results on *Test* for replacing our entity ranking with entity rankings obtained using other methods. ▲ denotes significant improvement and ▼ denotes significant deterioration compared to *BERT (PRF-Psg)* (denoted ⋆) using a paired-t-test at $p < 0.05$.

	Entity Ranking			Derived Aspect Ranking			LTR (Derived + Lexical)		
Entity Ranking Method	MAP@100	P@R	NDCG@100	P@1	MAP	NDCG@20	P@1	MAP	NDCG@20
BERT (PRF-Psg)⋆	**0.51**⋆	**0.51**⋆	**0.51**⋆	**0.78**⋆	**0.83**⋆	**0.84**⋆	**0.89**⋆	**0.94**⋆	**0.95**⋆
BERT (LeadText)	0.33▼	0.31▼	0.57▲	0.35▼	0.56▼	0.66▼	0.64▼	0.78▼	0.83▼
Relatedness (Wikipedia2Vec [55])	0.30▼	0.28▼	0.53▲	0.35▼	0.55▼	0.65▼	0.64▼	0.78▼	0.83▼
Relatedness (E-BERT [37])	0.30▼	0.28▼	0.53▲	0.35▼	0.55▼	0.65▼	0.64▼	0.78▼	0.83▼

ranking method *BERT (PRF-Psg)* achieves P@1 $= 0.78$ and outperforms all other aspect rankings. On combining with lexical features using LTR, we obtain further performance improvements. However, on replacing *BERT (PRF-Psg)* with an alternative method, the quality of both the derived aspect ranking as well as the LTR system deteriorates. This shows that *BERT (PRF-Psg)* plays an important role in the performance improvements obtained by us.

From Table 2, we observe that all LTR systems using aspect rankings derived from alternative entity ranking methods perform the same as *LTR-Lex* in Table 1 (P@1 $= 0.64$), i.e., removing these aspect rankings from the feature set does not affect performance. In contrast, from Table 1, we observe that removing our aspect ranking *EntAsp* from the feature set deteriorates the performance of the LTR system by 28%, from P@1 $= 0.89$ for *LTR-EntLex* to P@1 $= 0.64$ for *LTR-Lex*. This shows that *BERT (PRF-Psg)* contributes significantly to the better performance demonstrated by *LTR-EntLex*.

RQ4: Why does *EntAsp* outperform *EntityMatch (BM25)*? From Table 1, we observe that our aspect ranking *EntAsp* obtained using our proposed entity ranking approach outperforms *EntityMatch (BM25)* (used in previous work [32, 40]) on both Test and Nanni-Test. For example, on Test, *EntityMatch (BM25)* obtains P@1 $= 0.34$ whereas *EntAsp* obtains P@1 $= 0.78$.

We find that *EntAsp* helps 2427 entity mentions for which *EntityMatch (BM25)* fails. On further investigation, we find that few entities are shared between the context and an aspect. Instead of the *hard* entity matching of *EntityMatch (BM25)*, we perform a *soft* matching by using the top-k context-relevant entities to match the context and an aspect. This makes *EntAsp* successful. Moreover, our entity ranking model *BERT (PRF-Psg)* uses a context-relevant BM25 passage instead of the generic lead text as an entity's description. From Table 2, we observe that the entity ranking and the derived aspect ranking obtained using *BERT (PRF-Psg)* outperforms those obtained using *BERT (LeadText)*. Using a better context-specific entity description, our model can capture the context-relevant connections between an entity and the context. This helps our model to better identify context-relevant guiding entities which makes *EntAsp* successful.

6 CONCLUSION

In this work, we demonstrate the benefits of using entities as guides for a long-form text similarity problem: Entity aspect linking (EAL). We propose a novel approach for EAL that given a link context, learns to predict entities from the correct aspect using a ground truth of entities linked to the correct aspect. We use the top-k of these predicted entities as guides to the correct aspect: We rank aspects by the number of guiding entities they mention. Finally, we combine our aspect ranking with some lexical aspect rankings explored in related work. Our approach outperforms several hard neural and non-neural baselines, and alleviates the issue with entity matching for semantic similarity from previous work. We show that our proposed entity ranking approach contributes significantly to this performance improvement.

We envision this work on entity aspect linking to be useful for other systems (e.g., document clustering) that model the similarity between two long texts. By demonstrating the effectiveness of predicting entities as guides for entity aspect linking, we hope to motivate more research on using supervised entity ranking for tasks that require an explicit semantic understanding of text.

REFERENCES

[1] Krisztian Balog, Marc Bron, and Maarten De Rijke. 2011. Query Modeling for Entity Search Based on Terms, Categories, and Examples. *ACM Trans. Inf. Syst.* 29, 4 (2011).

[2] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Durán, Jason Weston, and Oksana Yakhnenko. 2013. Translating Embeddings for Modeling Multi-Relational Data. In *NIPS*.

[3] Marc Bron, Krisztian Balog, and Maarten de Rijke. 2013. Example Based Entity Search in the Web of Data. In *ECIR*.

[4] Shubham Chatterjee and Laura Dietz. 2021. Entity Retrieval Using Fine-Grained Entity Aspects. In *SIGIR*.

[5] Shubham Chatterjee and Laura Dietz. 2022. BERT-ER: Query-Specific BERT Entity Representations for Entity Ranking. In *SIGIR*.

[6] Marek Ciglan, Kjetil Nørvåg, and Ladislav Hluchý. 2012. The SemSets Model for Ad-Hoc Semantic List Search. In *WWW*.

[7] Hongliang Dai, Donghong Du, Xin Li, and Yangqiu Song. 2019. Improving Fine-grained Entity Typing with Entity Linking. In *EMNLP-IJCNLP*.

[8] Zhuyun Dai, Chenyan Xiong, Jamie Callan, and Zhiyuan Liu. 2018. Convolutional Neural Networks for Soft-Matching N-Grams in Ad-Hoc Search. In *WSDM*.

[9] Jeffrey Dalton, Laura Dietz, and James Allan. 2014. Entity Query Feature Expansion Using Knowledge Base Links. In *SIGIR*.

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL*.

[11] Laura Dietz. 2019. ENT Rank: Retrieving Entities for Topical Information Needs through Entity-Neighbor-Text Relations. In *SIGIR*.

[12] Laura Dietz and John Foley. 2019. TREC CAR Y3: Complex Answer Retrieval Overview.. In *Proceedings of Text REtrieval Conference (TREC)*.

[13] Faezeh Ensan and Ebrahim Bagheri. 2017. Document Retrieval Model Through Semantic Linking. In *WSDM*.

[14] Paolo Ferragina and Ugo Scaiella. 2010. TAGME: On-the-Fly Annotation of Short Text Fragments (by Wikipedia Entities). In *CIKM*.

[15] Besnik Fetahu, Katja Markert, and Avishek Anand. 2015. Automated News Suggestions for Populating Wikipedia Entity Pages. In *CIKM*.

[16] Jianfeng Gao, Patrick Pantel, Michael Gamon, Xiaodong He, and Li Deng. 2014. Modeling Interestingness with Deep Neural Networks. In *EMNLP*.

[17] Darío Garigliotti and Krisztian Balog. 2017. On Type-Aware Entity Retrieval. In *ICTIR*.

[18] Emma J Gerritse, Faegheh Hasibi, and Arjen P de Vries. 2020. Graph-Embedding Empowered Entity Retrieval. In *ECIR*.

[19] Faegheh Hasibi, Krisztian Balog, and Svein Erik Bratsberg. 2016. Exploiting Entity Linking in Queries for Entity Retrieval. In *ICTIR*.

[20] Hiroaki Hayashi, Prashant Budania, Peng Wang, Chris Ackerson, Raj Neervannan, and Graham Neubig. 2021. WikiAsp: A Dataset for Multi-domain Aspect-based Summarization. *TACL* 9 (03 2021).

[21] K Sparck Jones, Steve Walker, and Stephen E. Robertson. 2000. A Probabilistic Model of Information Retrieval: Development and Comparative Experiments: Part 2. *Information Processing and Management* 36, 6 (2000).

[22] Diederik P Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *arXiv preprint arXiv:1412.6980* (2014).

[23] Victor Lavrenko and W. Bruce Croft. 2001. Relevance-Based Language Models. *SIGIR Forum* (2001).

[24] Xitong Liu and Hui Fang. 2015. Latent Entity Space: A Novel Retrieval Approach for Entity-bearing Queries. *Information Retrieval Journal* 18, 6 (2015).

[25] Zhenghao Liu, Chenyan Xiong, Maosong Sun, and Zhiyuan Liu. 2018. Entity-Duet Neural Ranking: Understanding the Role of Knowledge Graph Semantics in Neural Information Retrieval. In *ACL*.

[26] Zhenghao Liu, Kaitao Zhang, Chenyan Xiong, Zhiyuan Liu, and Maosong Sun. 2021. OpenMatch: An Open Source Library for Neu-IR Research. In *SIGIR*.

[27] Jarana Manotumruksa, Jeff Dalton, Edgar Meij, and Emine Yilmaz. 2020. Cross-BERT: A Triplet Neural Architecture for Ranking Entity Properties. In *SIGIR*.

[28] Pablo N. Mendes, Max Jakob, Andrés García-Silva, and Christian Bizer. 2011. DBpedia Spotlight: Shedding Light on the Web of Documents. In *I-Semantics*.

[29] Donald Metzler and W. Bruce Croft. 2005. A Markov Random Field Model for Term Dependencies. In *SIGIR*.

[30] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. In *NIPS*.

[31] Bhaskar Mitra, Fernando Diaz, and Nick Craswell. 2017. Learning to Match Using Local and Distributed Representations of Text for Web Search. In *WWW*.

[32] Federico Nanni, Simone Paolo Ponzetto, and Laura Dietz. 2018. Entity-Aspect Linking: Providing Fine-Grained Semantics of Entities in Context. In *JCDL*.

[33] Fedor Nikolaev, Alexander Kotov, and Nikita Zhiltsov. 2016. Parameterized Fielded Term Dependence Models for Ad-Hoc Entity Retrieval from Knowledge Graph. In *SIGIR*.

[34] Hamid Palangi, Li Deng, Yelong Shen, Jianfeng Gao, Xiaodong He, Jianshu Chen, Xinying Song, and Rabab Ward. 2016. Deep Sentence Embedding Using Long Short-Term Memory Networks: Analysis and Application to Information Retrieval. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.* 24, 4 (2016).

[35] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation. In *EMNLP*.

[36] Francesco Piccinno and Paolo Ferragina. 2014. From TagME to WAT: A New Entity Annotator. In *ERD*.

[37] Nina Poerner, Ulli Waltinger, and Hinrich Schütze. 2020. E-BERT: Efficient-Yet-Effective Entity Embeddings for BERT. In *Findings of EMNLP*.

[38] Jay Michael Ponte and W Bruce Croft. 1998. *A language modeling approach to information retrieval*. Ph. D. Dissertation.

[39] Marco Ponza, Paolo Ferragina, and Soumen Chakrabarti. 2017. A Two-Stage Framework for Computing Entity Relatedness in Wikipedia. In *CIKM '17*.

[40] Jordan Ramsdell and Laura Dietz. 2020. A Large Test Collection for Entity Aspect Linking. In *CIKM*.

[41] Hadas Raviv, Oren Kurland, and David Carmel. 2016. Document Retrieval Using Entity-Based Language Models. In *SIGIR*.

[42] Ridho Reinanda, Edgar Meij, and Maarten de Rijke. 2016. Document Filtering for Long-Tail Entities. In *CIKM*.

[43] Petar Ristoski and Heiko Paulheim. 2016. Rdf2vec: Rdf Graph Embeddings for Data Mining. In *ISWC*.

[44] Livia Ruback, Claudio Lucchese, Alexander Arturo Mera Caraballo, Grettel Monteagudo García, Marco Antonio Casanova, and Chiara Renso. 2018. Computing Entity Semantic Similarity by Features Ranking. *arXiv preprint arXiv:1811.02516* (2018).

[45] Gerard Salton and Christopher Buckley. 1988. Term-weighting approaches in automatic text retrieval. *Information processing and management* 24, 5 (1988).

[46] Michael Schuhmacher, Laura Dietz, and Simone Paolo Ponzetto. 2015. Ranking Entities for Web Queries Through Text and Knowledge. In *CIKM*.

[47] Yelong Shen, Xiaodong He, Jianfeng Gao, Li Deng, and Grégoire Mesnil. 2014. A Latent Semantic Model with Convolutional-Pooling Structure for Information Retrieval. In *CIKM*.

[48] Karen Sparck Jones. 1988. *A Statistical Interpretation of Term Specificity and Its Application in Retrieval*.

[49] Ian H Witten and David N Milne. 2008. An Effective, Low-Cost Measure of Semantic Relatedness Obtained from Wikipedia Links. (2008).

[50] Chenyan Xiong and Jamie Callan. 2015. EsdRank: Connecting Query and Documents through External Semi-Structured Data. In *CIKM*.

[51] Chenyan Xiong, Jamie Callan, and Tie-Yan Liu. 2016. Bag-of-Entities Representation for Ranking. In *ICTIR*.

[52] Chenyan Xiong, Jamie Callan, and Tie-Yan Liu. 2017. Word-Entity Duet Representations for Document Ranking. In *SIGIR*.

[53] Chenyan Xiong, Zhuyun Dai, Jamie Callan, Zhiyuan Liu, and Russell Power. 2017. End-to-End Neural Ad-Hoc Ranking with Kernel Pooling. In *SIGIR*.

[54] Chenyan Xiong, Russell Power, and Jamie Callan. 2017. Explicit Semantic Ranking for Academic Search via Knowledge Graph Embedding *(WWW)*.

[55] Ikuya Yamada, Akari Asai, Jin Sakuma, Hiroyuki Shindo, Hideaki Takeda, Yoshiyasu Takefuji, and Yuji Matsumoto. 2020. Wikipedia2Vec: An Efficient Toolkit for Learning and Visualizing the Embeddings of Words and Entities from Wikipedia. In *EMNLP*.

[56] Weixin Zeng, Jiuyang Tang, and Xiang Zhao. 2019. Measuring Entity Relatedness via Entity and Text Joint Embedding. *Neural Processing Letters* 50, 2 (2019).

[57] Nikita Zhiltsov, Alexander Kotov, and Fedor Nikolaev. 2015. Fielded Sequential Dependence Model for Ad-Hoc Entity Retrieval in the Web of Data. In *SIGIR*.