Entity and Event Topic Extraction from Podcast Episode Title and Description Using Entity Linking

Christian Siagian
Amazon
Los Angeles, CA, USA
siagian@amazon.com

Amina Shabbeer
Amazon
San Francisco, CA, USA
aminashabbeer@gmail.com

ABSTRACT

To improve Amazon Music podcast services and customer engagements, we introduce Entity-Linked Topic Extraction (ELTE) to identify well-known entity and event topics from podcast episodes. An entity can be a person, organization, work-of-art, etc., while an event, such as the Opioid epidemic, occurs at specific point(s) in time. ELTE first extracts key-phrases from episode title and description metadata. It then uses entity linking to canonicalize them against Wikipedia knowledge base (KB), ensuring that the topics exist in the real world. ELTE also models NIL-predictions for entity or event topics that are not in the KB, as well as topics that are not of entity or event type. To test the model, we construct a podcast topic database of 1166 episodes from various categories. Each episode comes with a Wiki-link annotated main topic or NIL-prediction. ELTE produces the best overall Exact Match *EM* score of .84, with by-far the best *EM* of .89 among the entity or event type episodes, as well as NIL-predictions for episodes without entity or event main topic (*EM* score of .86).

CCS CONCEPTS

• **Computing methodologies** → **Information extraction**; **Natural language processing**.

KEYWORDS

Natural Language Understanding, topic extraction, entity linking

ACM Reference Format:
Christian Siagian and Amina Shabbeer. 2023. Entity and Event Topic Extraction from Podcast Episode Title and Description Using Entity Linking. In *Companion Proceedings of the ACM Web Conference 2023 (WWW '23 Companion), April 30-May 4, 2023, Austin, TX, USA.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3543873.3587648

1 INTRODUCTION

Podcast is a long-form spoken-word discussion or narrative from various categories, such as comedy and politics. Because of the length of a typical episode (45 minutes on average) there is a need to succinctly summarize the content. Topics, which are short phrases, are useful for search terms, topic recommendation, metadata enrichment, as well as custom presentation and experience. We define a topic as succinct, significant, and relevant phrase that describes the corresponding episode audio recording. Per knowledge-base nomenclature [24], a topic can be an entity (abstract or concrete, fictional or reality), event, or subject matter/general concepts. Entities (or EN-type) include people, organization places, creative works (books, songs, etc.), and other self-contained objects. Events (or EV-type) are named happenings at least at one point in time, such as "the 2020 U.S. presidential election". Subject matters (or SM-type) are studied thoughts such as empathy and Buddhism.

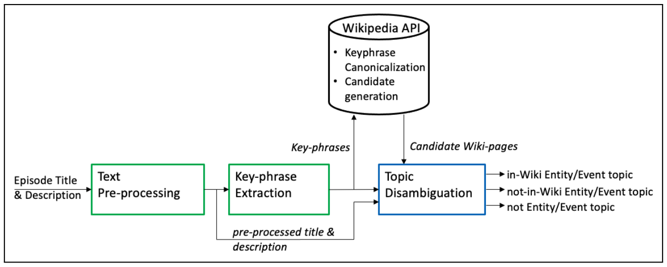


Figure 1: Entity-Linked Topic Extraction model diagram.

Topics, however, are not readily available as metadata. Thus, we draw from two advancements in Natural Language Understanding (NLU): topic extraction [4, 14, 20] and Entity Linking (EL) [19]. Topic extraction processes input text to find representative key-phrases. Entity linking, on the other hand, looks for phrases (called mentions) that can be linked to a knowledge base (KB), e.g. Wikipedia. Entity linking is comprised of three steps: mention detection, candidate generation, and entity disambiguation. The first step identifies phrases in the text that may potentially be entries in the KB. These mentions are then matched with the KB entries to produce candidates (second step) for further matching. The candidates then undergo entity disambiguation to finalize the best match.

The presented model Entity-Linked Topic Extraction (ELTE) combines topic extraction and entity linking to produce topic that both represents the input text, as well as significant by the virtue of being linked to Wikipedia. This significance step is what is missing in the current topic extraction models, making them susceptible to random/vague phrases. ELTE draws inspiration from the EL pipeline, modifying its three-step process into: key-phrase extraction, KB candidate generation, and topic disambiguation (observe figure 1). The last step, which replaces entity disambiguation in EL, outputs a wiki-page that not only fits the mention, but is also the main topic of the episode.

We start with detecting just EN and EV -type topics because they are self-contained, and only have a few synonyms (e.g. nicknames). This enables ELTE to employ topic canonicalization, which enforces

WWW '23 Companion, April 30-May 4, 2023, Austin, TX, USA

ACM ISBN 978-1-4503-9419-2/23/04.
https://doi.org/10.1145/3543873.3587648

single entry in Wikipedia despite name variants. Subject matter topics (for example: love), on the other hand, are conceptual, making them describable in many ways, or be further sub-divided. It is important to note that ELTE can recognize that a main episode topic is not an entity or event, and returns no result. Moreover, if a topic is not in Wikipedia, it is considered as not notable enough. In entity disambiguation literature [19], these cases are known as un-linkable mention or NIL-prediction. Consequently, ELTE also returns no result. Finally, ELTE extracts key-phrases from metadata episode title and description, and not ASR (Automatic Speech Recognition) transcripts. This is because recognizing out-of-vocabulary named entities in speech (via ASR) consistently is still difficult as ASR often mistranscribe them to common words [5, 23].

2 RELATED WORKS

Topic Extraction, which extract key-phrases from a text, can be done in two ways. One is abstractive [15], where a model generates phrases that may not be explicitly mentioned in the text. The other is extractive [4, 14, 20], where the model simply highlights phrases in the text. In our settings, we focus on extractive techniques because topic disambiguation may need the coordinate of the mentions. Extractive techniques can be statistically-based [4, 18], primarily operating on word counts, with semantic heuristics. It can be graph-based [3, 8, 16, 20], which expands upon simple counts by creating word connectedness network. It can also be embedding-based [2, 11, 14, 21] to increase semantic power. In addition, with the focus on entity and event type topic, Named Entity Recognition (NER), a standard task for NLU architectures (e.g. BERT [7], XL-NET [29], RoBERTa [13]), is also appropriate.

Entity disambiguation takes generated wiki-page candidates and selects the best match for the corresponding mention in the text. There are two entity disambiguation approaches: individually score the mention-Wiki-page pairs [9], or make use of the relationships between mentions [1, 6, 28]. For the former, a model computes semantic embeddings (e.g. BERT [7], Wikipedia2Vec [26]) for both the contextual mention and Wiki page, before measuring proximity e.g., cosine similarity. The latter emphasizes dependencies between mentions to maintain consistency throughout the text (e.g. theme, co-references). [28] iteratively solves mentions from most confident to the least, while re-computing inference after each passing Wiki-page selection. [6] reformulates the problem into an auto-regressive model by having a decoder generate the Wiki-page title for each mention. [1] concatenates the page titles and turn entity disambiguation into title selection.

3 DESIGN AND IMPLEMENTATION

The presented model Entity-Linked Topic Extraction (ELTE) takes in a concatenation of episode title and description text and outputs an episode main topic EN/EV Wiki page, if exists, otherwise NIL-prediction. As shown in figure 1, it starts with text pre-processing to remove html tags and advertising. It then performs key-phrase extraction, candidate generation, and ends with topic disambiguation. We use the term "key-phrase" for the extraction step, prior to disambiguation. Once the knowledge base (KB) information has been accounted for, and the phrase is confirmed to represent the episode, it becomes a topic.

For advertising identification, the model first performs sentence tokenization. It then removes sentences that are in more than a quarter of the episodes of each show. These repeated sentences are likely to be ads, or not episode-specific. Given an input episode e, the Text Pre-processing module outputs cleaned metadata text m_e.

The Key-phrase Extraction module uses BERT-NER [7] to extract a set of key-phrases $K_e = \{k_{e,i}\}, i = 1, 2, 3$, or the (maximum of) first three mentions in m_e. Three is sufficient because EN/EV episodes tend to mention the main topics in the title or early in the description for clarity. Note that the module filters out key-phrases that occur in more than half of each show. These repeated phrases are likely to be show names or hosts.

ELTE uses Wikipedia API [10] to generate Wiki-page candidate set $P_{e,i} = \{p_{e,i,j}\}, j = 1, 2, 3$ for each key-phrase mention $k_{e,i}$. Thus, for topic disambiguation, there is a maximum of nine pages to consider: three candidate Wiki pages times three mentions. Note that Wikipedia can return zero matches if there are no good ones, or errored out when there are too many matches (for vague queries). For those, ELTE simply ignores it and moves on to the next phrase.

Then, topic disambiguation encodes m_e using BERT-based sentence transformer (T) to produce embedding $\mu_e = T(m_e)$. Using the same process, it also encodes each page summary $s_{e,i,j}$ (an attribute of page $p_{e,i,j}$) to produce $\sigma_{e,i,j} = T(s_{e,i,j})$. It then semantically compares μ_e and $\sigma_{e,i,j}$ using cosine similarity:

$$cos(\mu_e, \sigma_{e,i,j}) = \frac{\mu_e \cdot \sigma_{e,i,j}}{\|\mu_e\| \, \|\sigma_{e,i,j}\|} \tag{1}$$

The page with the highest similarity score is the most likely episode topic page:

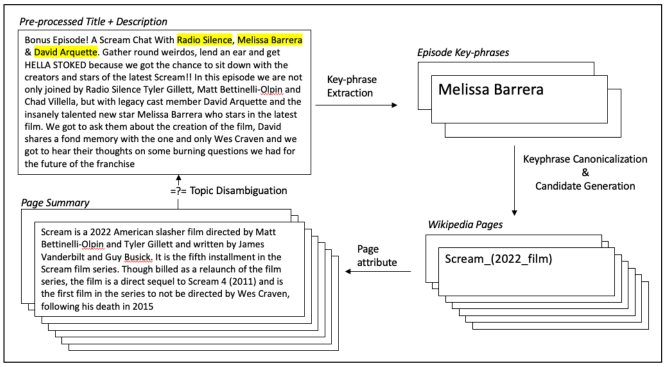
$$\pi_e = \underset{p_{e,i,j}}{\arg\max} \; (cos(\mu_e, \sigma_{e,i,j}) + boost(i, j)) \tag{2}$$

We boost the score of $p_{e,0,0}$ by an empirically-established $boost(0, 0) = .25$ to emphasize the best Wikipedia search match. That is, lower ranked pages have to be significantly more aligned with the input text to be selected. We also threshold the score π_e to test for NIL-prediction (for both not-in-Wiki-EN/EV and not-EN/EV) using threshold $t = .5$ for $p_{e,0,0}$, and $t = .4$ for all other pages.

Figure 2 illustrates an episode about the 2002 Scream movie. Top-left box is the episode title + description text, with the first three key-phrases highlighted. The phrases are then canonicalized and linked. One is a link between key-phrase "Melissa Barrera" and the movie Wiki page "Scream (2002)". Even though keyphrase extraction missed the phrase "Scream", ELTE is able to obtain it via the star of the movie. It also picked the correct one among the identical titled movies. The Wiki page summary (bottom-left box), is then compared with the metadata text. Topic disambiguation deemed they are close enough semantically, given mentions of director and cast. By definition, the subject of the page is the main topic of the episode.

4 TESTING AND DISCUSSION

To test the model, we created a human-annotated dataset of 1166 episodes from entity and event-topic (EN/EV)-dominant shows, primarily from the Comedy, True Crime, and History categories. The dataset is made of selected title and description texts of podcast show episodes, available freely on the internet. For each episode, an



Bonus Episode! A Scream Chat With Radio Silence, Melissa Barrera & David Arquette. Gather round weirdos, lend an ear and get HELLA STOKED because we got the chance to sit down with the creators and stars of the latest Scream!! In this episode we are not only joined by Radio Silence Tyler Gillett, Matt Bettinelli-Olpin and Chad Villella, but with legacy cast member David Arquette and the insanely talented new star Melissa Barrera who stars in the latest film. We got to ask them about the creation of the film, David shares a fond memory with the one and only Wes Craven and we got to hear their thoughts on some burning questions we had for the future of the franchise

Scream is a 2022 American slasher film directed by Matt Bettinelli-Olpin and Tyler Gillett and written by James Vanderbilt and Guy Busick. It is the fifth installment in the Scream film series. Though billed as a relaunch of the film series, the film is a direct sequel to Scream 4 (2011) and is the first film in the series to not be directed by Wes Craven, following his death in 2015

Figure 2: ELTE Example. Refer to the text for details.

annotator reads the respective text to determine a single appropriate phrase as a main topic, along with the corresponding ground truth Wiki page. If the phrase or Wiki page does not exist, none will be assigned to the episode (NIL). That is, even though the selected shows uses a main subject format (e.g. celebrities, songs, or events), they may have specials: a "best of" show, or a discussion on an issue. Thus, even within an EN/EV-dominant shows, there are episodes with not-EN/EV main topic.

The dataset consists of: 846 (or 72.6%) in-Wiki EN/EV episodes, 192 (16.4%) not-in-Wiki EN/EV episodes, and 128 (11.0%) not EN/EV episodes. For EN/EV episodes, there are mostly only one ground truth EN/EV main topic. For our use case, there are times when multiple Wikipedia pages are appropriate, e.g. pages for Jon Stewart vs. The Daily Show, or the band vs. the front-man. Thus, we add acceptable substitutes as alternatives. Note that the models are only expected to output one of them.

Sections below report evaluation details for key-phrase extraction and topic disambiguation. We also evaluated the Wikipedia search [10] candidate generation step. We found that it generates the correct Wiki-page within its top three 99.3% of the time. For this reason ELTE only passes the first three candidate for each key-phrase input to Topic Disambiguation.

4.1 Key-phrase Extraction

We evaluate statistic-based YAKE! [4], graph-based RaKUn [20], transformer-based TNT-KID [14], and BERT-NER [7] on the 846 in-Wiki EN/EV episodes. This is because, technically, there are no correct key-phrase for NIL-prediction cases. Given an episode title and description, an algorithm must output a key-phrase linkable to the ground truth EN/EV main topic. Ideally, an algorithm should only produce one, the correct, key-phrase. However, it usually outputs a list of key-phrases. Therefore, table 1 reports percentage accuracy of how early in the list the target key-phrase appears. The earlier, the better.

As shown, the ELTE utilized BERT-NER performs the best. Within top 3, it already includes the correct key-phrase in 96.7% (818 of 846) of the episodes. The results also confirm that the EN/EV topic key-phrases are mostly mentioned near the front of the text. Yake! is second, but with a substantial difference of 5.8% lower. In addition, Yake! also has many more mis-segmentations in its key-phrases. Upon further analysis, we found a number of grammatical and semantic understanding mistakes, which is less common in Transformer-backed models.

Table 1: Keyphrase Extraction Comparison

Algorithm	Top 1	Top 2-3	Top 4-5	Beyond 5	Not Found
Yake! [4]	74.23%	16.67%	5.20%	1.89%	2.01%
RaKUn [20]	34.16%	10.52%	2.01%	1.30%	52.01%
TNT-KID [14]	46.93%	13.12%	0.59%	0.00%	39.36%
BERT-NER [7]	83.22%	13.48%	1.77%	0.71%	0.83%

The other two models, RaKUn and TNT-KID, perform worse than Yake!. They both have many more Not Founds, but because of different reasons. RaKUn produces a lot of phrases that are not of topical value for the respective texts. This is due to its graph mechanism unable to accumulate importance on the main key-phrase above other common words. Conversely, the transformer-based TNT-KID returns very few key-phrases because it does not find enough semantic evidence that any phrase is the main topic. This is because the metadata text can be short and not stay on-topic. Hence both algorithms often failed because they rely on the main topic being repeated often. In addition, TNT-KID tends to prefer common-word main topics (e.g. Simulation Theory), which is not the emphasis of our EN/EV dataset. This is due to TNT-KID not having encyclopedic knowledge, unlike entity-enriched Transformer-based models [22, 28].

BERT-NER, on the other hand, is unaffected by the loose-narrative metadata text. It strictly looks for named entity and does not purposely compute for topics. Within ELTE framework, that is the job of topic disambiguation.

4.2 Topic Disambiguation

Because, our dataset only needs one or zero topic (for NIL-prediction) to register a positive outcome, it is more akin to Question and Answering evaluation. Therefore, we report Exact Match EM with null reject option [17]. That is, the percentage of correct answers whether it be a wiki-page or NIL-prediction: $EM = \frac{TP}{TP+FP+FN}$.

Here, we define true positive TP as when the model output is equal to ground truth Wiki-page or NIL-prediction. False positive FP is when the model outputs a Wiki-page different than the ground truth, both Wiki page or NIL-prediction. False negative FN is when the model produces no output, and the ground truth is a Wiki-page. In other words, false negatives only occur when the model produces an incorrect NIL-prediction. We also add Answered Accuracy $AA = \frac{TP}{TP+FP}$, or percentage of correct answers, excluding no answers for non-NIL-prediction episodes. Note that, for NIL-prediction cases, EM and AA are equal as there are no false negatives in those cases.

We implement a few models for topic disambiguation comparison. Some are entity disambiguation models as there is no, to the best of our knowledge, entity-linking model that is used for topic extraction model like ELTE.

The first model we call "Wikipedia search" keeps the rest of ELTE intact, but disambiguate topic by selecting the first wiki-page output of the first mention. The second model is the entity-disambiguation portion of the Radboud Entity Linker (REL) service [22]. It takes in key-phrases (from Key-phrase Extraction), and metadata text as

Table 2: Topic Disambiguation Results for In-Wikipedia Entity and Event Episodes

System	Total	*TP*	*FP*	*FN*	*AA*	*EM*
Wikipedia search	846	644	158	44	0.80	0.76
REL [22]	846	506	138	202	0.79	0.60
ELTE	846	755	50	41	0.94	0.89
Luke-ED [28]	20	116	15	75	0.89	0.56

Table 3: Topic Disambiguation Results for Not-In-Wikipedia Entity and Event Episodes

System	Total	*TP*	*FP*	*FN*	*AA*	*EM*
Wikipedia search	192	26	166	0	0.14	0.14
REL [22]	192	145	47	0	0.76	0.76
ELTE	192	118	74	0	0.61	0.61
Luke-ED [28]	28	27	1	0	0.96	0.96

Table 4: Topic Disambiguation Results for Not Entity and Event Episodes

System	Total	*TP*	*FP*	*FN*	*AA*	*EM*
Wikipedia search	128	52	76	0	0.41	0.41
REL [22]	128	74	54	0	0.58	0.58
ELTE	128	110	18	0	0.86	0.86
Luke-ED [28]	13	9	4	0	0.69	0.69

Table 5: Topic Disambiguation Results for the Full Dataset

System	Total	*TP*	*FP*	*FN*	*AA*	*EM*
Wikipedia search	1166	722	400	44	0.64	0.62
REL [22]	1166	725	239	202	0.75	0.62
ELTE	1166	983	142	41	0.87	0.84
Luke-ED [28]	247	152	20	75	0.88	0.62

context. It has its own candidate generation. REL uses the Latent Relation algorithm [12], which models relations between mentions. For the main topic, we also take the result of the first mention. The third one is Luke-ED [28], an entity disambiguation model based on Luke [27], the default entity-enriched architecture in HuggingFace [25]. The model takes metadata text, key-phrases (from Key-phrase Extraction), and the wiki-page candidates. It outputs matching scores for all candidates across the key-phrases. If the highest score is above a threshold, we select the corresponding wiki-page as the episode topic (else NIL-prediction).

Tables 2, 3, 4, 5 report the topic disambiguation results, respectively, for in-Wiki-EN/EV episodes, not-in-Wiki-EN/EV episodes, not-EN/EV episodes, and the full dataset. For each, we list the total, True Positive *TP*, False Positive *FP*, False Negative *FN*, Exact Match *EM*, and Answered Accuracy *AA*. Note that total number of episodes for [28] is smaller because it is only trained on the 500K most frequent Wiki-pages in EL benchmarking.

The scores for EN/EV episodes show high *AA* for all algorithms. Even Wikipedia search, where it just takes the best first mention match from Wikipedia, the *AA* is .80. However, its *AA* results for the NIL-prediction cases (not-in-Wiki-EN/EV and not-EN/EV) are quite bad: .14 and .41, respectively. This is because Wikipedia search results are designed for display. Instead of discarding bad matches, it opts to share the results for customers to decide.

REL [22], on the other hand, does model NIL-prediction. In turn, it performs better in those episodes. However, its false negatives (incorrect NIL-predictions) for EN/EV episodes (202) spikes up. What is worse is REL's *AA* is still below Wikipedia search. One reason for the spike is that the REL service has not kept its knowledge-base up-to-date since 2019. As it is, well-known entities such as the Tiger King TV show (2020) is not in the KB it is using. However, the main issue is probably because podcast metadata texts are less on-topic than the usual EL benchmark, making REL not want to force the issue.

The last two models, ELTE and [28], show markedly higher *AA* for EN/EV episodes (.94 and .89, respectively). The biggest reason is because they produce topic disambiguation scores for page-candidates that are comparable across all first three mentions. Wikipedia and REL [22], on the other hand, either produces no such scores, or only scores that are valid for within-mention comparison. By design, [28]‘s entity disambiguation scores are not intended for topic disambiguation. However, because it computes more globally-influenced scores (from the whole text), as opposed to just a few words adjacent to the mention, we feel it is appropriate for the task. And, for the most part, the results demonstrate it.

For the two NIL prediction cases (not-in-Wiki-EN/EV and not-EN/EV), both ELTE and [28] perform reasonably well. ELTE has the best not-EN/EV EM score of .86, with most mistakes being close to borderline. This goes to show that for this type of episode, many times, it is clear that no single entity can explain the full episode. For not-in-Wiki-EN/EV episodes, ELTE is less successful (EM of .61) while [28] is nearly perfect EM score of .96, albeit in limited numbers of 28 episodes. Among the 74 ELTE false positives (of 192), only four are non-crime episodes. This is because many of the discussed crimes do not have corresponding Wiki pages. However, coarsely similar events with Wiki pages (in terms of details, even with first or last name overlaps) do exist.

5 CONCLUSION

Overall, ELTE produces the main entity or event topic from episode title and description at .84 exact match score, demonstrating the importance of entity linking for improving topic extraction. Key-phrase canonicalization helps resolve synonyms, partial names, and miss-segmented phrases. Also, the utilization of knowledge ensures the significance of each topic, and allows ELTE to discard vague or tangential phrases. As far as we know there is no topic extraction model that implements entity-linking or knowledge integration to improve accuracy like ours. In our production pipeline, an ELTE system similar to the core scientific approach described in this paper has been implemented in production, with modifications for amenable commercial usage. That is, ELTE is paired with human-in-the-loop manual operation. We see improvements leading to reduction in manual effort by 86.7%.

REFERENCES

[1] Edoardo Barba, Luigi Procopio, and Roberto Navigli. 2022. ExtEnD: Extractive Entity Disambiguation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2478–2488.

[2] Kamil Bennani-Smires, Claudiu Musat, Andreea Hossmann, Michael Baeriswyl, and Martin Jaggi. 2018. Simple unsupervised keyphrase extraction using sentence embeddings. *arXiv preprint arXiv:1801.04470* (2018).

[3] Adrien Bougouin, Florian Boudin, and Béatrice Daille. 2013. Topicrank: Graph-based topic ranking for keyphrase extraction. In *International joint conference on natural language processing (IJCNLP)*. 543–551.

[4] Ricardo Campos, Vítor Mangaravite, Arian Pasquali, Alípio Jorge, Célia Nunes, and Adam Jatowt. 2020. YAKE! Keyword extraction from single documents using multiple local features. *Information Sciences* 509 (2020), 257–289.

[5] Nilaksh Das, Duen Horng Chau, Monica Sunkara, Sravan Bodapati, Dhanush Bekal, and Katrin Kirchhoff. 2022. Listen, Know and Spell: Knowledge-Infused Subword Modeling for Improving ASR Performance of OOV Named Entities. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 7887–7891.

[6] Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. 2020. Autoregressive entity retrieval. *arXiv preprint arXiv:2010.00904* (2020).

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[8] Corina Florescu and Cornelia Caragea. 2017. Positionrank: An unsupervised approach to keyphrase extraction from scholarly documents. In *Proceedings of the 55th annual meeting of the association for computational linguistics (volume 1: long papers)*. 1105–1115.

[9] Matthew Francis-Landau, Greg Durrett, and Dan Klein. 2016. Capturing semantic similarity for entity linking with convolutional neural networks. *arXiv preprint arXiv:1604.00734* (2016).

[10] Jonathan Goldsmith. 2013. Wikipedia API for python. https://pypi.org/project/wikipedia/

[11] Maarten Grootendorst. 2020. BERTopic: Leveraging BERT and c-TF-IDF to create easily interpretable topics. *Zenodo, Version v0* 9 (2020).

[12] Phong Le and Ivan Titov. 2018. Improving entity linking by modeling latent relations between mentions. *arXiv preprint arXiv:1804.10637* (2018).

[13] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692* (2019).

[14] Matej Martinc, Blaž Škrlj, and Senja Pollak. 2022. TNT-KID: Transformer-based neural tagger for keyword identification. *Natural Language Engineering* 28, 4 (2022), 409–448.

[15] Rui Meng, Xingdi Yuan, Tong Wang, Peter Brusilovsky, Adam Trischler, and Daqing He. 2019. Does order matter? an empirical study on generating multiple keyphrases as a sequence. *arXiv preprint arXiv:1909.03590* (2019).

[16] Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*. 404–411.

[17] Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. *arXiv preprint arXiv:1806.03822* (2018).

[18] Stuart Rose, Dave Engel, Nick Cramer, and Wendy Cowley. 2010. Automatic keyword extraction from individual documents. *Text mining: applications and theory* 1, 1-20 (2010), 10–1002.

[19] Özge Sevgili, Artem Shelmanov, Mikhail Arkhipov, Alexander Panchenko, and Chris Biemann. 2022. Neural entity linking: A survey of models based on deep learning. *Semantic Web* Preprint (2022), 1–44.

[20] Blaž Škrlj, Andraž Repar, and Senja Pollak. 2019. Rakun: Rank-based keyword extraction via unsupervised learning and meta vertex aggregation. In *International Conference on Statistical Language and Speech Processing*. Springer, 311–323.

[21] Si Sun, Zhenghao Liu, Chenyan Xiong, Zhiyuan Liu, and Jie Bao. 2021. Capturing global informativeness in open domain keyphrase extraction. In *CCF International Conference on Natural Language Processing and Chinese Computing*. Springer, 275–287.

[22] Johannes M van Hulst, Faegheh Hasibi, Koen Dercksen, Krisztian Balog, and Arjen P de Vries. 2020. 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*. 2197–2200.

[23] Haoyu Wang, Shuyan Dong, Yue Liu, James Logan, Ashish Kumar Agrawal, and Yang Liu. 2020. ASR Error Correction with Augmented Transformer for Entity Retrieval.. In *Interspeech*. 1550–1554.

[24] Gerhard Weikum, Xin Luna Dong, Simon Razniewski, Fabian Suchanek, et al. 2021. Machine knowledge: Creation and curation of comprehensive knowledge bases. *Foundations and Trends® in Databases* 10, 2-4 (2021), 108–490.

[25] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771* (2019).

[26] Ikuya Yamada, Akari Asai, Jin Sakuma, Hiroyuki Shindo, Hideaki Takeda, Yoshiyasu Takefuji, and Yuji Matsumoto. 2018. Wikipedia2Vec: An efficient toolkit for learning and visualizing the embeddings of words and entities from Wikipedia. *arXiv preprint arXiv:1812.06280* (2018).

[27] Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. 2020. LUKE: deep contextualized entity representations with entity-aware self-attention. *arXiv preprint arXiv:2010.01057* (2020).

[28] Ikuya Yamada, Koki Washio, Hiroyuki Shindo, and Yuji Matsumoto. 2022. Global Entity Disambiguation with BERT. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 3264–3271.

[29] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems* 32 (2019).