On the Surprising Effectiveness of Name Matching Alone in Autoregressive Entity Linking

Elliot Schumacher **James Mayfield** **Mark Dredze**
Johns Hopkins University
eschuma7@jhu.edu mayfield@jhu.edu mdredze@cs.jhu.edu

Abstract

Fifteen years of work on entity linking has established the importance of different information sources in making linking decisions: mention and entity name similarity, contextual relevance, and features of the knowledge base. Modern state-of-the-art systems build on these features, including through neural representations (Wu et al., 2020). In contrast to this trend, the autoregressive language model GENRE (De Cao et al., 2021) generates normalized entity names for mentions and beats many other entity linking systems, despite making no use of knowledge base (KB) information. How is this possible? We analyze the behavior of GENRE on several entity linking datasets and demonstrate that its performance stems from memorization of name patterns. In contrast, it fails in cases that might benefit from using the KB. We experiment with a modification to the model to enable it to utilize KB information, highlighting challenges to incorporating traditional entity linking information sources into autoregressive models.

1 Introduction

Early work in entity linking in Wikipedia (Cucerzan, 2007; Bunescu and Paşca, 2006) followed by the formulation of the task at the TAC KBP shared task (McNamee and Dang, 2009; Ji et al., 2010; Li et al., 2011) has led to more than a decade of research into how to match textual mentions of entities to grounded entities in a knowledge base (KB). This large body of research has led to some clear findings (Dredze et al., 2010; Durrett and Klein, 2014; Gupta et al., 2017; Lample et al., 2016; Francis-Landau et al., 2016; Cao et al., 2018; Wang et al., 2015; Witten and Milne, 2008; Piccinno and Ferragina, 2014). Entity linking is commonly modeled as a ranking task, in which a triaged set of KB entities is ranked by comparison to a textual entity mention. These ranking systems rely on different information sources. First, the entity mention is compared to the entity name in the KB (name matching), with allowances for aliases, acronyms, etc. Second, the context of the mention is compared to entity descriptions in the KB to select the correct entity among a set of similarly named candidates. Third, other relevant information from the KB (type information, links to related entities, popularity, etc.) can help disambiguate between candidates. This information is formulated as features (either engineered or learned) into the ranking system.

The recent emergence of autoregressive large language models as multi-task learners (Radford et al., 2019) has led to numerous new applications of these models. These models have been particularly effective in few-shot learning settings (Brown et al., 2020; Chowdhery et al., 2022), but typically fall behind supervised training of traditional systems that can flexibly incorporate a range of features. Despite this trend, De Cao et al. (2021) presented GENRE, an autoregressive language model that uses supervised training to link textual mentions to entities in a KB. Given a sentence and a previously-identified mention span, the model generates an entity name selected from a set of (triaged) candidates, with the option to generate entities without any constraints (with worse performance). Surprisingly, aside from the entity name, GENRE uses no information from the KB, in contrast to other high-performing entity linking systems that rely on textual entity descriptions (Wu et al., 2020) or type information (Orr et al., 2020). We may expect an autoregressive LM to do well, but how can it beat the best available feature-based entity linking systems?

We explore the benefits and drawbacks of autoregressive entity linking. First, we ask – why GENRE performs so well? Our answer comes from an analysis of the behavior of GENRE across several different entity linking datasets. Specifically, we measure the generalization ability of the

Proceedings of the First Workshop on Matching From Unstructured and Structured Data (MATCHING 2023), pages 58–69
July 13, 2023 ©2023 Association for Computational Linguistics

model by looking at performance on new datasets and knowledge bases. We find that GENRE relies heavily on memorization of name patterns, meaning that it struggles to generalize to new entities and KBs. KB information is often found to be useful in these cases, but its absence from GENRE means it struggles when name matching fails. Therefore, our second question is: can GENRE make use of information from the KB when available? Specifically, we provide descriptive information about an entity from the KB to GENRE and measure its resulting performance in various settings. We find that while it sometimes can make use of this information, it still struggles to learn generalizable patterns. Our analysis shows opportunities for incorporating KB information into an autoregressive entity linker, but also the challenges of doing so given current model architectures.

2 Autoregressive Entity Linking

GENRE (De Cao et al., 2021) is an autoregressive language model that links textual mentions to entities in Wikipedia through text generation. Autoregressive language models, such as BART (Lewis et al., 2020), are trained to generate text, as opposed to other non-autoregressive based models (*e.g.,* BERT (Devlin et al., 2019)), which are better suited for classification or scoring tasks. BART and similar models do very well at text generation tasks (Johner et al., 2021).

GENRE formulates entity linking as text generation as follows. Given the selected entity mention and its left context within the sentence, the model is trained to predict the next tokens as the normalized entity name. Consider the example in Figure 1. The model encodes the context *Two of the party's European*, and is trained to generate the correct normalized entity name *European Parliament* for this context. During training, the model is trained to minimize the smoothed cross-entropy loss between the generated entity name and the correct (normalized) entity name, where the normalized entity name matches the title of the associated node in the KB (Wikipedia page title). In this setup, negative sampling is not required. GENRE starts with a pretrained BART model and continues training on 9 million example entity mentions selected from Wikipedia, where the entity name is appended after each entity mention (see Section 5).

Asking GENRE to freely generate a normalized name is both extremely challenging and unnecessary. In practice, a pre-filtering (triage) step can be used to automatically select the most likely entity candidates for a textual reference via a name matching algorithm.[1] De Cao et al. (2021) evaluated GENRE under several conditions. First, a free decoding step whereby the model could output any string; this did not do well. Second, constraining the model to generate a valid entity name from the KB. Third, constraining the model to generate an entity from the small set of triaged candidates. For the constrained generation case, the authors constructed a trie $\mathcal{T}$, where each node of the trie consists of a vocabulary entry, with a specialized token in the root. For each subword $t \in \mathcal{T}$, its children are allowed subword continuations.

In an evaluation on the several entity linking datasets, including Wikipedia and MSNBC (Derczynski et al., 2015), GENRE achieved state-of-the-art results compared to traditional entity linking systems. Yet the shocking thing about this result is what GENRE lacks. First, GENRE uses no information from the KB. Typical entity linking systems consider contextual overlap between the mention string and the KB entity description; GENRE does not. For example, when linking the textual mention *America*, a system would measure overlap with the KB description *The United States of America is a transcontinental country primarily located in North America* (United States) or *Americans are the citizens and nationals of the United States of America.* (American). Another popular feature is entity type, for example, *country* (United States) or *nationality* (American). Other feature such as entity popularity, entity type, and related entities, are not available to GENRE. This information has long been used to disambiguate entities, and recent systems continue to show their ongoing effectiveness. Orr et al. (2020) use type information to help disambiguate entities that do not occur frequently. BLINK (Wu et al., 2020) build contexualized embeddings for each entity using entity descriptions. None of this information is available to GENRE.

Furthermore, due to the generation nature of BART, GENRE only uses the left context of the entity mention. In sentences such as that in Figure 1, a very limited left context is availble to provide any information. While GENRE can memorize associations between the limited left context and the entity name, it cannot generalize even this limited

[1] This task itself is a challenge, and relying on a candidate set that contains the correct entity is often unrealistic.

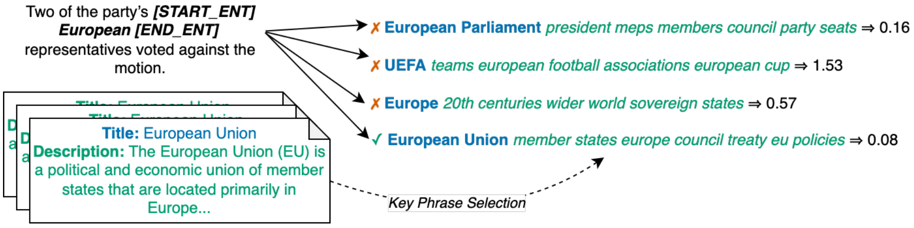
Two of the party's [START_ENT] European [END_ENT] representatives voted against the motion.
✗ European Parliament president meps members council party seats ⇒ 0.16
✗ UEFA teams european football associations european cup ⇒ 1.53
✗ Europe 20th centuries wider world sovereign states ⇒ 0.57
✓ European Union member states europe council treaty eu policies ⇒ 0.08
Title: European Union
Description: The European Union (EU) is a political and economic union of member states that are located primarily in Europe...
Key Phrase Selection

Figure 1: An example mention taken from the TAC training set. In the original GENRE model, constrained decoding would be performed over only the **normalized entity names** (in blue, bolded) in the candidate list, given the mention and the sentence context. In our proposed GENRE-KB, we perform constrained decoding over the **normalized entity names** and *keywords* taken from entity descriptions in the knowledge base.

information to new settings. Despite these limitations, GENRE represents a state-of-the-art entity linker.

3 GENRE and Generalization

How does GENRE achieve great entity linking results with such limited information? We explore this through the issue of generalization: how well does the model do on new unseen data?

Since the model does not have access to the KB, its predictions on new data are based entirely on what it can learn about entities fron training data.

De Cao et al. (2021) suggested that GENRE predicts entities with contextualized name matching by leveraging large amounts of entity linking annotations during training. For example, while the original authors show that the model performs acceptably on rare entities (*e.g.,* approximately 80% accuracy on Wikipedia entities seen once in the training data), the accuracy for entities unseen in the training data is only 50%. Bhargav et al. (2022) show that GENRE is very data-intensive to train; reducing training to 0.01% of the original size performs 11% worse than BLINK. Constrained decoding is also necessary for accurate predictions. Generating without triaged candidates drops the accuracy by 9.2%. However, the importance of training data is clearly central, as triage could be adapted to new settings separately.

What is GENRE learning from the massive training data? One possibility is that it learns how to normalize entity name (*Bill Clinton* to *William Clinton*) from annotated data. Pretraining on massive amounts of unannotated text followed by a large amount of entity linking annotations may also allow it to learn how to normalize certain informal names (*America*) to formal ones (*The United States*). Furthermore, pretraining may allow for robust modeling of the context before mentions. Finally, as in other NLP tasks, the effect of using the encoding of the context provided by the sentence is likely valuable.

If GENRE exhibits these behaviors, it can generalize certain abilities to new domains. However, if instead it is memorizing the training data, e.g. learning specific entities that appear in training, it cannot generalize. For example, Wikipedia titles and mentions follow conventions, which may be learnable by the model, but will not generalize to settings that do not use Wikipedia data or KBs. Additionally, De Cao et al. (2021) report results on examples where the gold entity is found in the triage step, which biases toward lexical matches. Examples that can be lexically matched are likely more likely to be solved by name matching. These links are far more common in Wikipedia than other domains.

In short, while generalization is a challenge for any machine learning model, it may be especially challenging for the mechanisms used by GENRE to learn from the training data. Our first question is: Does GENRE learn generalizable patterns or does it memorize the entities in the training data? We answer by probing how GENRE leverages its training data to perform linking. We evaluate GENRE on new datasets (Section 5) more challenging than those reported in the original paper. We begin with datasets linked to Wikipedia KBs, the proceed to datasets with different KBs. These new KBs contain entities unobserved in training, especially difficult for GENRE because it cannot access the KB.

4 GENRE and the Knowledge Base

GENRE faces challenges in generalization from its lack of access to the KB, which contains information about unseen entities. If GENRE was able to access the KB, could it better generalize to new data? A long line of entity linking research suggests that the answer should be "yes". In this Section, we modify the training data to provide this information to GENRE.

The key idea is to augment the training data with short descriptions of information in the KB. Specifically, we add several keywords that summarize an entity's description in the KB to each training instance. GENRE is then asked (and trained) to generate the entity title followed by these keywords after each entity mention. This approach uses an unchanged GENRE model architecture to both learn to normalize names and bias the model towards entity descriptions (via keywords) that are most triggered by the (left) context of the mention.

We choose to use keywords instead of the full text description for several reasons. First, in many KBs (especially Wikipedia) entity descriptions are quite long, often multiple paragraphs. This stretches the context beyond what GENRE can reasonably model. Even selecting a short snippet, e.g. the first sentence, also pushes the model beyond what is reasonable. Instead, selecting a few important phrases from the description allows us to easily control the length of the produced string. Furthermore, if selected correctly, these keyword can highlight topically related content, signaling a match with the left context of the entity.

Context enables GENRE to match the topic of the context with that of the candidate entity. In Figure 1, which entity best matches the the term *European* is ambigious. Although the correct entity *European Union* has a partial lexical match, other entities do as well (*e.g., European Parliament*), and others are close lexical variants (*Europe*). GENRE's ability to link this mention correctly would likely solely be based on whether it is seen in the training data, given the ambigiuity in the knowledge base. Adding additional keywords can signal that *European Union* and *European Parliament* are potentially related, given political-related keywords such as *party* and *council*, whereas *Europe* is less related. The same approach may be helpful to other mentions that could be amibigously linked in the knowledge base, such as *Washington*. The keywords for *Washington D.C.*, *district city congress united states metropolitan area*, can help differentiate that entity from *Washington (State)*, which is paired with keywords *seattle united states british columbia cascade range*. This idea is in the same spirit as Bevilacqua et al. (2022), which uses autoregressive language models for search, but decodes entire spans from a corpus, as opposed to keywords.

4.1 Keyword Selection

We use the PKE toolkit (Boudin, 2016) to select keywords from the entity description. After a careful examination of several of the unsupervised methods in the toolkit, we found that Topic Rank (Bougouin et al., 2013) produced the most descriptive keywords. We selected the top n keywords (phrases) and multiplied the Topic Rank score s by a frequency factor from the KB. For each keyword in the KB, we took a summation over their inverse rank ($\frac{1}{rank+1}$) within each entity-specific set. The final score for a keyword k for a given entity is

$$s_k * (1 + \log(\sum_{e \in \text{KB}, k \in e} \frac{1}{\text{rank}_k + 1})) \qquad (1)$$

The keywords are ordered by their score. The addition of the frequency factor removed some highly-scored esoteric keywords (*e.g., Punic Wars* for *Spain*) that may not generalize well. We also experimented with the number of keywords to include, and found that adding at least five words was best. Many keywords are phrases with multiple words, which results in some sequences being just over five words. This selection procedure can generalize to other sources of information in KBs.

To avoid GENRE memorizing this training data, we use a different selection method during the training step. During training, we sample five words from the entire keyword list proportional to the Topic Rank score, and resample for each training instance. Scores less than zero are set to a small value (0.0001), then normalized to form a probability distribution. At inference, we use the same top scoring keywords for every instance of an entity. Examples of selected keywords are shown in Appendix Table 4.

4.2 Training and Inference

We closely follow the training procedure in De Cao et al. (2021). Beginning with the pretrained GENRE model, we train GENRE-KB to maximize the entity title and keyword sequence given the sentence context: maximize $logp_\theta(y|x)$ with respect to the model's parameters θ. We closely follow their choices of training methods and parameter selections, and use teacher forcing, dropout, and label smoothing. The authors originally add a special token to the beginning of each target sequence. In

Dataset	Wikipedia		TAC	
	acc.	mrr	acc.	mrr
GENRE	92.1 ±.67	.952	92.4 ±.56	.950
+KB	81.1 ±.97	.874	91.8 ±.58	.950
GENRE*	90.9 ±.69	.943	80.7 ±.75	.856
+KB*	77.5 ±1.0	.846	80.9 ±.75	.862

Table 1: Datasets with Wikipedia as the KB. The first two rows show examples with correct entity in the triaged set. The rows with an asterisk show the oracle setting, where all examples with the correct candidate added if not present. Confidence Intervals (at 95%) are included for accuracy.

addition to using this token, we add special tokens before and after the keywords to indicate where keywords are present. We do not add these as tokens to the vocabulary due to Fairseq (Ott et al., 2019) constraints. We believe the performance difference is likely small.

Similarly, we use GENRE's candidate scoring with constrained beam search. For Wikipedia-based datasets, we use the same beam size (10) as in their work. However, for other datasets, we found that a smaller beam size works better (5). Additionally, since we are scoring longer strings that likely vary much more in length than in the title-only model, we explored normalizing the likelihood of a candidate by its length (in number of byte pair encoding tokens). In some datasets, we found this provided a small improvement. Training these models from scratch exceeded our computational resources, so we initialized training using the existing models. We trained each model on a single NVIDIA GeForce RTX 2080 for 32 hours, iterating over all the data.

5 Data

Wikipedia GENRE was trained on the BLINK-created version of a Wikipedia dataset (Wu et al., 2020) based on a May 2019 English Wikipedia dump with 5.9 million entities. They use a 9 million-sized subset of Wikipedia-linked mentions (*e.g.,* links within Wikipedia pages to other Wikipedia pages). The KB consists of all pages within that snapshot of Wikipedia. We exclusively use this dataset to train GENRE-KB. While we also report evaluation results on this dataset, we primarily target more challenging datasets. For evaluation, we use the provided candidate sets.

TAC The 2015 TAC KBP Entity Linking dataset (Ji et al., 2015) consists of newswire and discussion forum posts linked to an English KB. The discussion forum posts with informal entity mentions are especially challenging. Chinese and Spanish data are also included, but we only consider English. While this dataset does not directly link to Wikipedia, almost all entities linked in the English dataset include a Wikipedia title in their metadata. Therefore, we convert all entities with Wikipedia links to their respective entry in the Wikipedia KB and convert all others to NIL (no relevant entity). To generate a candidate set at inference time, we use the system of Upadhyay et al. (2018), which is largely based on work in Tsai and Roth (2016). This approach uses Wikipedia cross-links to generate a prior probability $\mathtt{P}_{\mathtt{prior}}(e_i|m)$ by estimating counts from those mentions. This prior is used to provide the top k English Wikipedia page titles for each mention.

Wikia To explore how GENRE and GENRE-KB work on datasets where Wikipedia is not the KB, we include the Wikia dataset (Logeswaran et al., 2019). Wikia was constructed from the Wikia.com website (now Fandom), which consists of community-written encyclopedias on a particular subject or theme. This was constructed in the same manner as the Wikipedia dataset – mentions were taken from in-page hyperlinks, and each document served as an entity. The authors collect 16 Wikias, each with a different topic and KB, thus serving as a challenging adaptation for our Wikipedia-trained models. The authors exclude all NIL entities and provide candidate sets for each mention of size 64, retrieved via BM25.

Topics are partitioned across training, validation, and test sets so that each appears in only one set. Each mention is categorized by the amount of token overlap between the mention text and the normalized entity title. The categories include *high overlap* (5% of mentions), which represent exact matches; *multiple categories* (28% of mentions), where the entity title is the mention text plus a disambiguation phrase (*e.g.,* mention *Batman*, entity title *Batman (Lego)*); and *ambiguous substring* (8% of mentions), where the mention is a substring of the title. The category *other* (59% of mentions) includes all remaining mentions. We believe the original label of *low overlap* is misleading, as many examples in that category have a high degree of lexical similarity. For example, of the *other* examples that have a candidate identified in the validation set, 28.96% of mention span - entity title pairs have a Jaro-Winkler lexical similarity (Winkler, 1990) of over 0.794.

6 Experimental Setup

For GENRE-KB, we train all models on the Wikipedia dataset alone and select the best-performing model using the Wikipedia validation set's loss. In all cases, we do not use the Wikia or TAC training data for training but only as a validation set. For Wikia and TAC data, we provide the model with the sentence where the mention occurs. Sentence boundaries are identified with Spacy (Honnibal and Montani, 2017). We adopt the method of reporting results from Logeswaran et al. (2019), which reports normalized accuracy, which is calculated over the set of examples that are non-NIL and have the gold standard entity in their candidate set. As this restricts the types of examples to those that have mentions which are lexically similar to the entity name, we also report oracle results for some datasets, where we add the gold standard entity to all non-NIL examples if not already present.

7 Results

Our experiments address two questions. First, why does GENRE perform so well? We answer this through evaluating generalization to new datasets. Second, can GENRE utilize KB information to improve generalization (GENRE-KB)?

7.1 GENRE Generalization

To probe GENRE's reliance on the mention string matching the normalized entity name, we performed two experiments with the TAC training dataset using the original GENRE model. First, we remove the available context around the tntity and replace it with a generic prompt: *This entity is called **mention***. In this setting, no context is available for linking decisions. Second, we keep the original context but remove the actual mention string. In this setting, GENRE relies on context alone.

How important to GENRE are each type of information: name matching and context? Compared to the normal model's performance of 49.1% on TAC data (unnormalized, *i.e.,* including NIL entities), using only the mention string GENRE did nearly as well (41.6%). By comparison, using only context drops accuracy significantly (26.8%). This suggests that GENRE largely relies on the training data to learn transformations between the mention and the entity name alone. The context adds a bit to the model's ability.

Despite this result, GENRE performs well on the more challenging datasets. Table 1 shows the performance of the GENRE model on the Wikipedia and TAC datasets. While it is unsurprising that GENRE performs well on Wikipedia, the performance on the TAC dataset is surprisingly high for the setting with only retrieved candidates. However, the performance on TAC in the oracle setting is significantly lower. As detailed in Section 6, we add the gold standard entity to the candidate set for any example where it isn't already present. Focusing only on the retrieved candidates restricts examples to those that can be lexically matched, as triage systems frequently rely on surface forms alone. The oracle setting highlights the fact that many of these more challenging matches cannot be linked by GENRE.

The results for Wikia are shown in Table 2. Previous work (Logeswaran et al., 2019) report results on several baselines for the validation set. We include the best-performing baselines that also have not been trained on Wikia data.[2] We report macro accuracy (accuracy is calculated separately on each domain, and divided by the number of domains), and micro accuracy (accuracy is calculated on the corpus as a whole), in addition to mean reciprocal rank (MRR) and top-K accuracy ($k = 5$). In absolute terms, the performance on the Wikia dataset is worse, as it is not trained to link mentions to the Wikia knowledge bases. However, it does outperform two previously reported baselines by a small margin, suggesting that even in this challenging setting GENRE is surprisingly effective.

The reason behind this effectiveness varies in each setting. For linking mentions to the Wikipedia KB, the sheer amount of data GENRE is trained on enables it to recall which entity is likely best. Therefore, when the data allows for such a strategy, memorization can be effective when paired with a model that can also model the context.

7.2 GENRE-KB

We evaluate GENRE-KB (GENRE augmented in training by keywords) on all of our datasets dis-

[2] The authors of that paper also include several baselines that are trained on Wikia data, but are an unfair comparison for this setting.

Method	Validation				Test			
	macro	micro	mrr	top-K	macro	micro	mrr	top-K
TF-IDF*	26.06							
Gupta et al*	27.03							
GENRE	29.09	26.89 ±1.0	.42	52.88	31.99	33.16 ±1.1	.44	43.01
GENRE-KB	29.53	29.63 ±1.0	.46	55.65	28.11	27.83 ±1.1	.42	44.64
Comb. (par)	35.54	35.14 ±1.1	.49	54.48	35.63	36.14 ±1.1	.47	43.89
Comb. (jw)	32.36	30.97 ±1.0	.46	58.82	34.48	35.00 ±1.1	.46	47.00

Table 2: Results on Wikia Datasets. Results for methods marked with an asterisk are taken from Logeswaran et al. (2019). The combination models are built off of the predictions of GENRE-KB and GENRE described in Chapter 4. Confidence Intervals (at 95%) are included for micro accuracy.

degree of similarity	validation accuracy			test accuracy		
	#	GENRE	GENRE-KB	#	GENRE	GENRE-KB
mult. categories	4106	11.93	26.04	2341	16.66	25.72
amb. substring	543	54.70	36.46	419	47.02	28.88
high overlap	501	89.22	71.66	825	91.03	62.30
other	2434	33.07	25.55	3227	28.54	20.42

Table 3: Results on Wikia by degree of similarity category.

cussed in the previous section. For the Wikipedia dataset in Table 1, GENRE performs consistently better than GENRE-KB. This is unsurprising, given the model's ability to memorize training examples and that it has been trained on other Wikipedia data. As reported in the previous section, GENRE relies heavily on name matching, which is sufficient when the model stays within the same domain. In addition, 82.9% of examples in the test set have a Jaro-Winkler score of 0.8 or higher, indicating they are largely lexically similar.

However, performance on the TAC dataset is much closer. On the set of examples where the correct entity is present in the triage candidate set, GENRE performs slightly better on accuracy, while both models tie in MRR. However, in the oracle setting, GENRE-KB performs marginally better in both metrics. This suggests that when trying to link these more challenging examples, which a lexical triage system could not identify, GENRE-KB has an advantage. In short, when context matters, GENRE-KB is better. However, it is still challenging to overcome the memorization capacity of the original GENRE model, and GENRE-KB is still based on the same architecture.

As shown in Table 1, the confidence intervals for accuracy ($\alpha = 0.05$) suggest that the differences in top-predictions are not significant for TAC, but are for Wikipedia. However, to test whether GENRE and GENRE-KB produce rankings that are significantly different, we use a Wilcoxon signed-rank test. For the TAC dataset, the difference between the two models on the Retrieved Candidates setting ($p = 0.005$) and the Oracle setting ($p = 0.005$) are both significant. This suggests the two models produce different rankings despite their similar top-level predictions.

Table 2 shows results on the Wikia validation and test sets. Again, the differences between GENRE and GENRE-KB are small and depend on the dataset. In the validation set, GENRE-KB performs better in all metrics. In test set, GENRE performs better with the exception of top-K accuracy, where GENRE-KB performs better. Comparing the rankings produced by the two models using a Wilcoxon signed-rank test, we find that the difference in the GENRE and GENRE-KB validation rankings is significant ($p = 2.1e - 36$), but not significant for the test rankings ($p = 0.13$). In terms of micro accuracy, the confidence intervals show that the differences between GENRE and GENRE-KB are significant.

At first glance, this suggests that the validation data was overfitted. However, we believe this has more to do with the distribution of examples in each set. Table 3 breaks down accuracy by similarity categories (detailed in Section 5). In the validation set, the largest category is *multiple categories*, which are linked to entities that have a parenthetical in their name. In both sets, GENRE-KB performs consistently better than GENRE, but the portion of these examples is smaller in the test set. Conversely, it is unsurprising that in the cases of *high overlap* and *amb. substring* GENRE performs better since those are categories with high lexical similarity between mention and entity title.

For the *other* category, GENRE performs well on examples with high lexical similarity. For example, in the validation set, while only 28.96% of textitother examples have a high lexical similarity, those examples consist of 52.9% of the examples that GENRE gets correct. GENRE performs better on test and GENRE-KB better on validation because the sets have a different distribution over example types.

GENRE and GENRE-KB are useful for different types of examples. GENRE is excellent when name string alone is sufficient. GENRE-KB improves when context matters. Therefore, we explore combining the two systems. Table 2 shows two methods for model combination. First, we propose a model (labeled *paren*) where we use the prediction from GENRE-KB if it predicts a parenthetical, and GENRE otherwise. Second, we combine scores of GENRE and GENRE-KB with the Jaro-Winkler lexical similarity between the GENRE model's top predicted entity and the mention serving as a scalar between the two scores (labeled as *jw*)[3]. This puts more weight on examples where GENRE thinks there is a lexically similar entity name to the mention, but more weight on GENRE-KB in dissimilar cases.

Neither model changes predictions based on the gold standard entity label – they only operate off of the top prediction of one of the two models. In both cases, across both data sets and metrics, both combination models outperform GENRE-KB and GENRE. The confidence intervals included in Table 2 suggest that while the difference between the *jw* model and the best-performing individual model is not significant, the difference between the *par* model and the best-performing individual model is significant. In summary, adding KB information to GENRE helps, but only where such information is informative to the correct prediction. A simple metric (Jaro Winkler) can successfully identify those cases.

8 Related Work

Entity linking has been broadly studied (Dredze et al., 2010; Durrett and Klein, 2014; Gupta et al., 2017; Lample et al., 2016; Cao et al., 2018; Wang et al., 2015; Wu et al., 2020). Recent work (Bhargav et al., 2022; Orr et al., 2020) highlights the utility of type information in making linking decisions for rarer entities. Other work has applied autoregressive models to other information extraction tasks (De Cao et al., 2022; Josifoski et al., 2022). De Cao et al. (2021) seeks to alleviate some of the performance challenges with GENRE during inference, although initial experiments found this performed worse in new domains. Aghajanyan et al. (2022) proposed a method that allows both the left and right context surrounding an entity mention to be modeled by producing the link at the end of the sequence.

9 Conclusion

Autoregressive transformer-based sequence-to-sequence models, such as BART, have found increasing success in information extraction tasks. The GENRE model, which applies autoregressive sequence-to-sequence approaches to entity linking, has high performance on many datasets linked to the Wikipedia domain. However, its performance on other domains with different challenges produces mixed results.

We suggest that adding previously-explored entity linking features to GENRE can address some of these pitfalls. Specifically, descriptions are a commonly used source of text to make linking decisions. While we see performance decreases in the original Wikipedia datasets, we see some improvements in both newswire text and in applying GENRE-KB to previously unseen knowledge bases for more challenging matches. Yet, the ability of GENRE to work in even challenging settings suggests that it can memorize patterns useful for mention-entity pairs with high lexical similarity.

There are several unexplored directions for our model. Specifically, we used an off-the-shelf keyword selection method. Selecting keywords in a more targeted fashion – perhaps by selecting keywords for an entity that best separates it from another entity – may improve performance. Having the computational resources to train a model from scratch would also likely improve performance, as opposed to training from a GENRE checkpoint. Moreover, we focus on integrating descriptive information within the original GENRE framework. Future work may consider an autoregressive entity linker with a novel architecture that can integrate and learn representations of entities would better utilize this information in learning.

[3] We divide the GENRE score by the candidate's length, to match the length normalization procedure of GENRE-KB, as described in Section 4.1.

10 Ethics and Limitations

Our experiments focus solely on English-language entity linking. Similar models have been trained to perform entity linking in multiple languages (De Cao et al., 2022), but we do not consider performance beyond English. The issues faced in other languages are likely to be similar, but the multilingual element of other models might lead to different results. Further, how to select keywords in the multilingual setting is unclear.

In addition, we are limited by the available annotated entity linking datasets. Given that we need a large amount of data to train these models, they are inherently reliant on Wikipedia. These entity linking datasets are skewed towards specific types of matches, including ones that are frequently exact matches. The effectiveness of this model might change when trained on a dataset with different characteristics, even with a large amount of data.

Finally, the computational resources required to train these models are large, and our final results do not reflect numerous other preliminary experiments. This restricts our ability to run multiple experiments, train models from scratch easily, and potentially leads to underfitting of our final models.

References

Armen Aghajanyan, Bernie Huang, Candace Ross, Vladimir Karpukhin, Hu Xu, Naman Goyal, Dmytro Okhonko, Mandar Joshi, Gargi Ghosh, Mike Lewis, and Luke Zettlemoyer. 2022. Cm3: A causal masked multimodal model of the internet.

Michele Bevilacqua, Giuseppe Ottaviano, Patrick Lewis, Wen tau Yih, Sebastian Riedel, and Fabio Petroni. 2022. Autoregressive search engines: Generating substrings as document identifiers. In *arXiv pre-print 2204.10628*.

G P Shrivatsa Bhargav, Dinesh Khandelwal, Saswati Dana, Dinesh Garg, Pavan Kapanipathi, Salim Roukos, Alexander Gray, and L Venkata Subramaniam. 2022. Zero-shot entity linking with less data. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1681–1697, Seattle, United States. Association for Computational Linguistics.

Florian Boudin. 2016. pke: an open source python-based keyphrase extraction toolkit. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations*, pages 69–73, Osaka, Japan.

Adrien Bougouin, Florian Boudin, and Béatrice Daille. 2013. TopicRank: Graph-based topic ranking for keyphrase extraction. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 543–551, Nagoya, Japan. Asian Federation of Natural Language Processing.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Razvan Bunescu and Marius Paşca. 2006. Using encyclopedic knowledge for named entity disambiguation. In *11th Conference of the European Chapter of the Association for Computational Linguistics*, pages 9–16, Trento, Italy. Association for Computational Linguistics.

Yixin Cao, Lei Hou, Juanzi Li, and Zhiyuan Liu. 2018. Neural collective entity linking. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 675–686, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2022. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*.

Silviu Cucerzan. 2007. Large-scale named entity disambiguation based on Wikipedia data. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 708–716, Prague, Czech Republic. Association for Computational Linguistics.

Nicola De Cao, Wilker Aziz, and Ivan Titov. 2021. Highly parallel autoregressive entity linking with discriminative correction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7662–7669, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. 2021. Autoregressive entity retrieval. In *International Conference on Learning Representations*.

Nicola De Cao, Ledell Wu, Kashyap Popat, Mikel Artetxe, Naman Goyal, Mikhail Plekhanov, Luke Zettlemoyer, Nicola Cancedda, Sebastian Riedel, and Fabio Petroni. 2022. Multilingual autoregressive entity linking. *Transactions of the Association for Computational Linguistics*, 10:274–290.

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

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Mark Dredze, Paul McNamee, Delip Rao, Adam Gerber, and Tim Finin. 2010. Entity disambiguation for knowledge base population. In *Conference on Computational Linguistics (COLING)*, pages 277–285. Association for Computational Linguistics.

Greg Durrett and Dan Klein. 2014. A joint model for entity analysis: Coreference, typing, and linking. *Transactions of the Association for Computational Linguistics*, 2:477–490.

Matthew Francis-Landau, Greg Durrett, and Dan Klein. 2016. Capturing semantic similarity for entity linking with convolutional neural networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1256–1261, San Diego, California. Association for Computational Linguistics.

Nitish Gupta, Sameer Singh, and Dan Roth. 2017. Entity linking via joint encoding of types, descriptions, and context. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2681–2690, Copenhagen, Denmark. Association for Computational Linguistics.

Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Heng Ji, Ralph Grishman, Hoa Trang Dang, Kira Griffitt, and Joe Ellis. 2010. Overview of the tac 2010 knowledge base population track. In *Third text analysis conference (TAC 2010)*, volume 3, pages 3–3.

Heng Ji, Joel Nothman, Ben Hachey, and Radu Florian. 2015. Overview of TAC-KBP2015 Tri-lingual Entity Discovery and Linking. *TAC*.

Timo Johner, Abhik Jana, and Chris Biemann. 2021. Error analysis of using BART for multi-document summarization: A study for English and German language. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 391–397, Reykjavik, Iceland (Online). Linköping University Electronic Press, Sweden.

Martin Josifoski, Nicola De Cao, Maxime Peyrard, Fabio Petroni, and Robert West. 2022. GenIE: Generative information extraction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4626–4643, Seattle, United States. Association for Computational Linguistics.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270, San Diego, California. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Xuansong Li, Joe Ellis, Kira Griffitt, Stephanie M Strassel, Robert Parker, and Jonathan Wright. 2011. Linguistic resources for 2011 knowledge base population evaluation. In *TAC*.

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.

Paul McNamee and Hoa Trang Dang. 2009. Overview of the tac 2009 knowledge base population track. In *Text analysis conference (TAC)*, volume 17, pages 111–113.

Laurel Orr, Megan Leszczynski, Simran Arora, Sen Wu, Neel Guha, Xiao Ling, and Christopher Re. 2020. Bootleg: Chasing the tail with self-supervised named entity disambiguation.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53, Minneapolis, Minnesota. Association for Computational Linguistics.

Francesco Piccinno and P. Ferragina. 2014. From tagme to wat: a new entity annotator. In *ERD '14*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Chen-Tse Tsai and Dan Roth. 2016. Cross-lingual Wikification Using Multilingual Embeddings. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational*

Linguistics: Human Language Technologies, pages 589–598, Stroudsburg, PA, USA. Association for Computational Linguistics.

Shyam Upadhyay, Nitish Gupta, and Dan Roth. 2018. Joint multilingual supervision for cross-lingual entity linking. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2486–2495, Brussels, Belgium. Association for Computational Linguistics.

Han Wang, Jin Guang Zheng, Xiaogang Ma, Peter Fox, and Heng Ji. 2015. Language and domain independent entity linking with quantified collective validation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 695–704, Lisbon, Portugal. Association for Computational Linguistics.

William E Winkler. 1990. String comparator metrics and enhanced decision rules in the fellegi-sunter model of record linkage. *ERIC*.

Ian H Witten and David N Milne. 2008. An effective, low-cost measure of semantic relatedness obtained from wikipedia links.

Ledell Wu, Fabio Petroni, Martin Josifoski, Sebastian Riedel, and Luke Zettlemoyer. 2020. Scalable zero-shot entity linking with dense entity retrieval. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6397–6407, Online. Association for Computational Linguistics.

Entity Title	**Keywords**
Germany	german states country member berlin france
Church of England	local parishes christianity common people bishop
General officer	army air forces countries different systems
Flowering plant	plants families species pollen embryo
Civil liberties	religion european convention constitution personal freedoms
Julia Gillard	leader education australia university labor
1924 World Series	games washington ninth walter johnson giants
John Hodgman	radio episode death role appearance
Humoral immunity	function phagocytosis cellular components presence antibodies
Camino Real (play)	time tennessee williams esmeralda marguerite camille
Bumper Tormohlen	december known seasons nba draft record
Craig Wiseman	tim mcgraw blake shelton songs year
Carroll Gardens Historic District	brooklyn common new york city smith
Dallas	city southern united states universities texas
Phanagoria	town site augustus black sea auxiliary bishop
Pierre Berton	time books canada ontario canadian history
Military advisor	afghanistan capabilities marines infantry vietnam
Francesca Schiavone	fourth round italy semifinals french open
Show Boat (1951 film)	julie stage play characters song magnolia
Los Angeles County, California	pasadena arts san bernardino port cities
Metatheria	years earliest marsupials placentals north america
The New York Times	articles report publisher newspaper paper
Tamil Nadu	india coimbatore parts british chennai
Government of Hong Kong	chief secretary systems chief executive head
Roberto Matta	europe surrealist art life work le corbusier
DC Comics	series line picture stories second title
The Outer Limits (1995 TV series)	tales season science fiction time monster
Marvel Comics	year american comic books titles series
Berkshire Hathaway	years share cash general decline stock
Portugal	lisbon portuguese government country territory spain
Methanosphaera	carbon dioxide taxonomy genus formate methanol

Table 4: Example keywords for the shuffled scoring selection method detailed in Section 4.1.