Multilingual Autoregressive Entity Linking

Nicola De Cao[1,4], **Ledell Wu**[7], **Kashyap Popat**[1], **Mikel Artetxe**[1],
Naman Goyal[3], **Mikhail Plekhanov**[1], **Luke Zettlemoyer**[3,5],
Nicola Cancedda[1], **Sebastian Riedel**[1,6], **Fabio Petroni**[1]

[1]Facebook AI, UK [2]Facebook AI, China [3]Facebook AI, USA [4]University of Amsterdam, NL [5]University of Washington, USA [6]University College London, UK
[7]Beijing Academy of Artificial Intelligence, China

nicola.decao@gmail.com, {ledell, kpopat, artetxe, naman, movb, lsz, ncan, sriedel, fabiopetroni}@fb.com, wuyu.ledell@gmail.com

Abstract

We present mGENRE, a sequence-to-sequence system for the Multilingual Entity Linking (MEL) problem—the task of resolving language-specific mentions to a multilingual Knowledge Base (KB). For a mention in a given language, mGENRE predicts the name of the target entity left-to-right, token-by-token in an autoregressive fashion. The autoregressive formulation allows us to effectively cross-encode mention string and entity names to capture more interactions than the standard dot product between mention and entity vectors. It also enables fast search within a large KB even for mentions that do not appear in mention tables and with no need for large-scale vector indices. While prior MEL works use a single representation for each entity, we match against entity names of as many languages as possible, which allows exploiting language connections between source input and target name. Moreover, in a zero-shot setting on languages with no training data at all, mGENRE treats the target language as a latent variable that is marginalized at prediction time. This leads to over 50% improvements in average accuracy. We show the efficacy of our approach through extensive evaluation including experiments on three popular MEL benchmarks where we establish new state-of-the-art results. Source code available at `https://github.com/facebookresearch/GENRE`.

1 Introduction

Entity Linking (EL, Bunescu and Paşca, 2006; Cucerzan, 2007; Hoffart et al., 2011; Dredze et al., 2010) is an important task in NLP, with plenty of applications in multiple domains, spanning Question Answering (De Cao et al., 2019; Nie et al., 2019; Asai et al., 2020), Dialogue (Bordes et al., 2017; Wen et al., 2017; Williams et al., 2017; Chen et al., 2017; Curry et al., 2018; Sevegnani et al., 2021), and Biomedical systems (Leaman and Gonzalez, 2008; Zheng et al., 2015), to name just a few. It consists of grounding entity mentions in unstructured texts to KB descriptors (e.g., Wikipedia articles).

The multilingual version of the EL problem has been for a long time tied to a purely cross-lingual formulation (XEL, McNamee et al., 2011; Ji et al., 2015), where mentions expressed in one language are linked to a KB expressed in another (typically English). Recently, Botha et al. (2020) made a step towards a more inherently multilingual formulation by defining a language-agnostic KB, obtained by grouping language-specific descriptors per entity. Such a formulation has the power of considering entities that do not have an English descriptor (e.g., a Wikipedia article in English) but have one in some other languages.

A common design choice to most current solutions, regardless of the specific formulation, is to provide a unified entity representation, either by collating multilingual descriptors in a single vector or by defining a canonical language. For the common bi-encoder approach (Wu et al., 2020; Botha et al., 2020), this might be optimal. However, in the recently proposed GENRE model (De Cao et al., 2021), an autoregressive formulation to the EL problem leading to stronger performance and considerably smaller memory footprints than bi-encoder approaches on monolingual benchmarks, the representations to match against are entity names (i.e., strings) and it's unclear how to extend those beyond a monolingual setting.

In this context, we find that maintaining as much language information as possible, hence

Transactions of the Association for Computational Linguistics, vol. 10, pp. 274–290, 2022. https://doi.org/10.1162/tacl_a_00460
Action Editor: Jing Jiang. Submission batch: 7/2021; Revision batch: 10/2021; Published 3/2022.

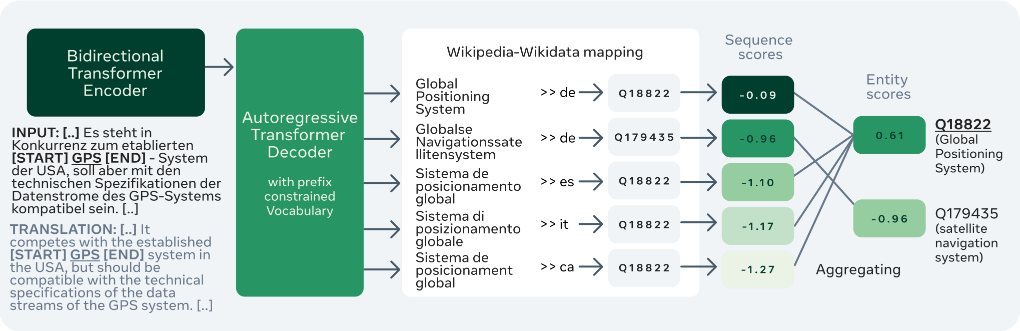

Bidirectional Transformer Encoder
INPUT: [..] Es steht in Konkurrenz zum etablierten [START] GPS [END] - System der USA, soll aber mit den technischen Spezifikationen der Datenstrome des GPS-Systems kompatibel sein. [..]
TRANSLATION: [..] It competes with the established [START] GPS [END] system in the USA, but should be compatible with the technical specifications of the data streams of the GPS system. [..]
Autoregressive Transformer Decoder
with prefix constrained Vocabulary
Wikipedia-Wikidata mapping
Global Positioning System
>> de
Q18822
Globalse Navigationssate llitensystem
>> de
Q179435
Sistema de posicionamento global
>> es
Q18822
Sistema di posizionamento globale
>> it
Q18822
Sistema de posicionament global
>> ca
Q18822
Sequence scores
-0.09
-0.96
-1.10
-1.17
-1.27
Entity scores
0.61
Q18822 (Global Positioning System)
-0.96
Q179435 (satellite navigation system)
Aggregating

Figure 1: mGENRE: the input is text where an entity mention is signaled with special separator tokens and the model outputs predictions for the entity identifier. We use an autoregressive transformer decoder to generate language IDs as well as entity names (i.e., Wikipedia titles). The combination of language ID and a entity name uniquely identify a Wikidata ID (with a N-to-1 mapping). We use Beam Search for efficient inference and we marginalize the probability scores for different languages to score entities. This example is a real output from our system and scores values are length normalized log-probabilities.

providing multiple representations per entity (i.e., one for each available language), helps due to the connections between source language and entity names in different languages. We additionally find that using all available languages as targets and aggregating over the possible choices is an effective way to deal with a zero-shot setting where no training data is available for the source language.

Concretely, in this paper, we present mGENRE, the first multilingual EL system that exploits a sequence-to-sequence architecture to generate entity names in more than 100 languages left to right, token-by-token in an autoregressive fashion and conditioned on the context (see Figure 1 for an outline of our system). While prior works use a single representation for each entity, we maintain entity names for as many languages as possible, which allows exploiting language connections between source input and target name. To summarize, this work makes the following contributions:

- Consider in the catalog of entity names all languages for each entry in the KB. Storing the multilingual names index is feasible and cheap (i.e., 2.2GB for ∼89M names).
- Design a novel objective function that marginalizes over all languages to perform a prediction. This approach is particularly effective in dealing with languages not seen during fine-tuning (∼50% improvements).
- Establish new state-of-the-art performance for the Mewsli-9 (Botha et al., 2020), TR2016$^{\text{hard}}$ (Tsai and Roth, 2016), and TAC-KBP2015 (Ji et al., 2015) MEL datasets.
- Present extensive analysis of modeling choices, including the usage of candidates from a mention table, frequency-bucketed evaluation, and performance on a held out set including low-resource languages.

2 Background

We first introduce Multilingual Entity Linking in Section 2.1, highlighting its difference with monolingual and cross-lingual linking. We address the MEL problem with a sequence-to-sequence model that generates textual entity identifiers (i.e., entity names). Our formulation generalizes the GENRE model by De Cao et al. (2021) to a multilingual setting (mGENRE). Thus in Section 2.2 and 2.3, we discuss the GENRE model and how it ranks entities with Beam Search respectively.

2.1 Task Definition

Multilingual Entity Linking (MEL, Botha et al., 2020) is the task of linking a given entity mention m in a given context c of language $l \in \mathcal{L}_C$ to the corresponding entity $e \in \mathcal{E}$ in a multilingual Knowledge Base (KB). See Figure 1 for an example: There are textual inputs with entity mentions (in bold) and we ask the model to predict the corresponding entities in the KB. A language-agnostic KB includes an entity descriptor (at least the name) of each entity in one or more languages. Note that there is no guarantee that an entity descriptor matching the context language is always available. We assume that descriptors in multiple languages for the same entity are mapped to a unique entry in the KB (e.g., as in Wikidata).

and that each $e \in \mathcal{E}$ has a descriptor in at least a language. Concretely, in this work, we use Wikidata (Vrandečić and Krötzsch, 2014) as our KB. Each item lists a set of Wikipedia pages in multiple languages linked to it and in any given language each page has a unique name (i.e., its title).

The MEL formulation is a generalization of both monolingual Entity Linking EL and cross-lingual EL (XEL, McNamee et al., 2011; Ji et al., 2015). The monolingual EL formulation considers a KB where each entity descriptor is expressed in the context language—mention and KB language always match, descriptors in other languages are discarded. One problem of this formulation is that the KB might miss several entries for languages with limited coverage of entity descriptors. The XEL formulation tries to mitigate this problem by considering the language with the highest descriptors coverage as canonical (typically English)—mentions in multiple languages are mapped to a single canonical language. Therefore, both the MEL and XEL formulations exploit inter-language links to identify entities in other languages. However, given that XEL requires the target KB to be monolingual, it might still miss several entries in the KB. For instance, Botha et al. (2020) reported that ≈25% of hyperlinks in the Japanese Wikinews do not point to a page that have a corresponding one in English.

In this work we assume that each entity descriptor contains a name that concisely describes an entity. In particular, we consider Wikipedia titles (in multiple languages) as entity names. Note that such entity names might not be available for other KBs. We consider the definition of meaningful entity names when not available an interesting future research direction.

2.2 Autoregressive Generation

GENRE ranks each $e \in \mathcal{E}$ by computing a score with an autoregressive formulation: $\text{score}_\theta(e|x) = p_\theta(y|x) = \prod_{i=1}^{N} p_\theta(y_i|y_{<i}, x)$ where y is the sequence of N tokens in the identifier of e, x the input (i.e., the context c and mention m), and θ the parameters of the model. GENRE is based on fine-tuned BART architecture (Lewis et al., 2020) and it is trained using a standard seq2seq objective, namely, maximizing the output sequence likelihood with teacher forcing (Sutskever et al., 2011, 2014) and regularized with dropout (Srivastava et al., 2014) and label smoothing (Szegedy et al., 2016).

2.3 Ranking with Constrained Beam Search

At test time, it is prohibitively expensive to compute a score for all elements in $\mathcal{E}$. Thus, GENRE exploits Beam Search (BS, Sutskever et al., 2014), an established approximate decoding strategy to navigate the search space efficiently. Instead of explicitly scoring all entities in $\mathcal{E}$, it searches for the top-k entities in $\mathcal{E}$ using BS with k beams. BS only considers one step ahead during decoding (i.e., it generates the next token conditioned on the previous ones). Thus, GENRE uses a prefix tree (trie) to enable constrained beam search and then generate only valid entity identifiers.

3 Model

To extend GENRE to a multilingual setting, we need to define what are the unique identifiers of all entities in a language-agnostic fashion. This is not trivial because we rely on text representations that are by their nature grounded in some language. Concretely, for each entity e, we have a set of identifiers $\mathcal{I}_e$ that consists of pairs $\langle l, n_e^l \rangle$ where $l \in \mathcal{L}_{KB}$ indicates a language and n_e^l the name of the entity e in the language l. We extract these identifiers from our KB—each Wikidata item has a set of Wikipedia pages in multiple languages linked to it, and in any given language, each page has a unique name. We identify 3 strategies to employ these identifiers:

i) define a *canonical* textual identifier for each entity such that there is a 1-to-1 mapping between the two (i.e., for each entity, select a specific language for its name—see Section 3.1);

ii) define an N-to-1 mapping between textual identifier and entities concatenating a language ID (e.g., a special token) followed by its name in that language—alternatively concatenating its name first and then a language ID (see Section 3.2);

iii) treat the selection of an identifier in a particular language as a latent variable (i.e., we let the model learn a conditional distribution of languages given the input and we marginalize over those—see Section 3.3).

All of these strategies define a different way we compute the underlining likelihood of our model.

In Figure 1 we show an outline of mGENRE. The following subsections will present detailed discussions of the above 3 strategies.

3.1 Canonical Entity Representation

Selecting a single textual identifier for each entity corresponds to choosing its name among all the available languages of that entity. We use the same data-driven selection heuristic as in Botha et al. (2020): for each entity e we sort all its names n_e^l for each language l according to the number of mentions of e in documents of language l. Then we take the name n_e^l in the language l that has the most mentions of e. In case of a tie, we select the language that has the most number of mentions across all entities (i.e., the language for which we have more training data). Having a single identifier for each entity corresponds to having a 1-to-1 mapping between strings and entities.[1] Thus, $\text{score}_\theta(e|x) = p_\theta(n_e|x)$ where with n_e we indicate the *canonical* name for e. A downside of this strategy is that most of the time, the model cannot exploit the lexical overlap between the context and entity name since it has to translate it in the canonical one (e.g., if the canonical name for the entity potato is ''Potato'' $^{\text{Q10998}}$ and the model encounters ''patata''—that is potato in Spanish—it needs to learn that one is the translation of the other).

3.2 Multilingual Entity Representation

To accommodate the canonical representation issues, we can predict entity names in any language. Concatenating a language ID l and an entity name n_e^l in different orders induces two alternative factorizations. We train maximizing the scores for all our training data: $\text{score}_\theta(e|x) =$

$$\begin{cases} p_\theta(l|x) \cdot p_\theta(n_e^l|x,l) & \text{for 'lang+name'} \\ p_\theta(n_e^l|x) \cdot p_\theta(l|n_e^l,x) & \text{for 'name+lang'} \end{cases} \quad (1)$$

The former corresponds to first predicting a distribution over languages and then predicting a title *conditioning* on the language l where the latter corresponds to the opposite. Predicting the language first conditions the generation to a smaller set earlier during beam search (i.e., all names in a specific language). However, it might exclude some targets from the search too early if the beam size is too small. Predicting the language last does not condition the generation of names in a particular language but it *asks* the model to disambiguate the language of the generated name whenever it is ambiguous (i.e., when the same name in different languages corresponds to possibly different entities). Only 1.65% of the entity names need to be disambiguated with the language. In practice, we observe negligible difference in performance between the two approaches. Both strategies define an N-to-1 mapping between textual identifiers and entities and then at test time we just use a lookup table to select the correct KB item. This N-to-1 mapping is an advantage compared to using canonical names because the model can predict in any available language and therefore exploit synergies between source and target language as well as avoiding translation.

3.3 Marginalization

Differently from the plain generation strategies described above, we can treat the textual identifiers as a latent variable and express $\text{score}_\theta(e|x)$ as the probability of the entity name in all languages and marginalizing over them:

$$\text{score}_\theta(e|x) = \sum_{\langle l, n_e^l \rangle \in \mathcal{I}_e} p_\theta(n_e^l, l|x) \;. \quad (2)$$

Marginalization exposes the model to all representations in all languages of the same entity and it requires a minor modification of the training procedure. Unfortunately, because computing $\text{score}_\theta(e|x)$ requires a sum over all languages, both training, and inference with marginalization are more expensive than with simple generation (scaling linearly with the number of languages). However, at least during inference, we can still apply BS to only marginalize using the top-k generations. For this reason, we test this training strategy only on few languages but we evaluate marginalization even when training with the other generation strategies described above.

3.4 Candidate Selection

Modern EL systems that use cross-encoding between context and entities usually do not score all entities in a KB as it is too computational expensive (Wu et al., 2020). Instead, they first apply candidate selection to reduce the number

[1] As this approach chooses one language per entity it might happen that two entities have the same canonical name in the two different languages. We address this issue appending the used language ID so that the combination of the two is always unique.

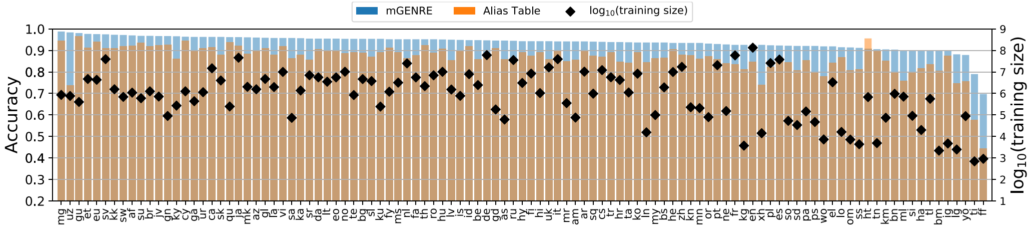


Figure 2: Accuracy of mGENRE on the 105 languages in our Wikipedia validation set. We also report the accuracy of the alias table and the log-training set sizes per each language.

of entities before scoring (with a less expensive method or a non-parametric mention table). In our formulation, there is no need to do that because mGENRE uses BS to generate efficiently. However, using candidates might help, and thus, we also experiment with that. Scoring all candidates might not be always possible (sometimes there are thousands of candidates for a mention) and especially when using an N-to-1 mapping between textual identifiers there will be names to rank in all languages available for each candidate. Then, when we use candidates, it is to constrain BS steps further, rather than to rank all of them. Concretely, candidate selection is made with an alias table. Using the training data, we build a mention table where we record all entities indexed by the names used to refer to them in any language. Additionally, we also use Wikipedia titles as additional mentions (useful for entities that never appear as links), redirects, Wikidata labels, and aliases.

4 Experimental Setting

We use Wikidata (Vrandečić and Krötzsch, 2014) as our KB while exploiting the supervision signal from Wikipedia hyperlinks. For evaluation, we test our model on two established cross-lingual datasets, TR2016$^{\text{hard}}$ and TAC-KBP2015 (Ji et al., 2015; Tsai and Roth, 2016), as well as the recently proposed Mewsli-9 MEL dataset (Botha et al., 2020). Additionally, we propose a novel setting extracted from Wikinews[2] where we train a model on a set of languages, and we test it on unseen ones.

4.1 Knowledge Base: Wikidata

We use Wikidata as the target KB to link to, filtering with the same heuristic as Botha et al. (2020) (see Appendix A for more details). Our entity set $\mathcal{E}$ contains 20,277,987 items (as a reference, English Wikipedia has just ≈6M items). Using the corresponding Wikipedia titles as textual identifiers in all languages leads to a table of 53,849,351 entity names. We extended the identifiers including redirects which leads to a total of 89,270,463 entity names. Although large, the number of entity names is not a bottleneck as the generated prefix tree occupies just 2.2GB for storage (Botha et al.'s [2020] systems need ≈10 times more storage).

4.2 Supervision: Wikipedia

For all experiments, we do not train a model from scratch, but we fine-tune a multilingual language model trained on 125 languages (see Appendix A for more details on the pre-trained model). We exploit Wikipedia hyperlinks as the source of supervision for MEL. We used Wikipedia in 105 languages out of the >300 available. These 105 are all the languages for which our model was pre-trained on that overlaps with the one available in Wikipedia (see full language list in Figure 2 and more details in Appendix A). We extracted a large-scale dataset of 734,826,537 datapoints (i.e., mention-entity pairs). For the plain generation strategy, we selected as the ground truth the name in the source language. When such an entity name is not available[3] we randomly select 5 alternative languages and we use all of them as datapoints. To enable model selection, we randomly selected 1k examples from each language for validation.

4.3 Datasets

For evaluation we use the recent multilingual-EL dataset Mewsli-9 (Botha et al., 2020), the cross-lingual TAC-KBP2015 Tri-Lingual Entity Linking (Ji et al., 2015) and TR2016$^{\text{hard}}$ (Tsai and Roth, 2016). We refer to the original works for details on the data, and we report in Appendix A.3 more details on pre-processing and evaluation.

[2] https://www.wikinews.org.

[3] This happens when there are broken links or links that points to pages in prospect of being created.

	ar	de	en	es	fa	ja	sr	ta	tr	**micro**	**macro**
Alias Table	89.0	86.0	79.0	82.0	87.0	82.0	87.0	79.0	80.0	83.0	83.0
Botha et al. (2020)	92.0	**92.0**	87.0	89.0	92.0	88.0	93.0	88.0	88.0	89.0	90.0
mGENRE	94.7	91.5	86.7	90.0	**94.6**	89.9	94.9	92.9	90.7	90.2	91.8
+ marg.	95.3	91.8	87.0	**90.1**	94.2	90.2	**95.0**	93.1	90.9	90.4	92.0
+ cand.	94.8	91.8	87.1	**90.1**	**94.6**	91.1	94.4	93.3	91.4	90.5	92.1
+ cand. + marg.	**95.4**	**92.0**	**87.2**	**90.1**	94.4	**91.4**	94.5	**93.8**	**91.5**	**90.6**	**92.3**

Table 1: Accuracy on Mewsli-9 dataset with micro and macro averages. The results of mGENRE are with and without top-k candidates from the alias table as well as with and without marginalization.

	TAC-KBP2015			**TR2016$^{\text{hard}}$**				
Method	**es**	**zh**	**macro-avg**	**de**	**es**	**fr**	**it**	**macro-avg**
Tsai and Roth (2016)	82.4	85.1	83.8	53.3	54.5	47.5	48.3	50.9
Sil et al. (2018)*	83.9	85.9	84.9	–	–	–	–	–
Upadhyay et al. (2018)	84.4	86.0	85.2	55.2	56.8	51.0	52.3	53.8
Zhou et al. (2019)	82.9	85.5	84.2	–	–	–	–	–
Botha et al. (2020)	–	–	–	**62.0**	58.0	54.0	56.0	57.5
mGENRE	86.3	64.6	75.5	56.3	57.1	50.0	51.0	53.6
mGENRE + marg.	**86.9**	65.1	76.0	56.2	56.9	49.7	51.1	53.5
mGENRE + cand.	86.5	86.6	86.5	61.8	**61.0**	**54.3**	**56.9**	**58.5**
mGENRE + cand. + marg.	86.7	**88.4**	**87.6**	61.5	60.6	**54.3**	56.6	58.2

Table 2: Accuracy on TAC-KBP2015 Entity Linking dataset (only datapoints linked to FreeBase) and TR2016$^{\text{hard}}$ of mGENRE (trained with 'title+lang') with and without top-k candidates from the table as well as with and without marginalization. *as reported by Upadhyay et al. (2018).

Wikinews-7 For the purpose of testing a model on languages unseen during training, we extract mention-entity pairs from Wikinews in 7 languages that are not in the Mewsli-9 language set.[4] Table 9 in Appendix A.3 reports statistics of this dataset. Wikinews-7 is created in the same way as Mewsli-9, but we used our own implementation to extract data from raw dumps.[5]

5 Results

The main results of this work are reported in Table 1 for Mewsli-9, and in Table 2 for TR2016$^{\text{hard}}$, and TAC-KBP2015 respectively. Our mGENRE (trained with 'title+lang') outperforms all previous works in all those datasets. We show the accuracy of mGENRE on the 105 languages in our Wikipedia validation set against an alias table baseline in Figure 2.

5.1 Performance Evaluation

Mewsli-9 In Table 1 we compare our mGENRE against the best model from Botha et al. (2020) (Model F^{+}) as well as with their alias table baseline. We report results from mGENRE with and without constraining the beam search to the candidates from the table (see Section 3.4) as well as with and without marginalization (see Section 3.3). All of these alternatives outperform Model F^{+} on both micro and macro average accuracy across the 9 languages. Our base model (without candidates or marginalization) has a 10.9% error reduction in micro average and 18.0% error reduction for macro average over all languages. The base model has no restrictions on candidates so it is effectively classifying among all the ≈20M entities. The base model performs better than Model F^{+} on each individual language except English and German. Note that these languages are the ones for which we have more training data (≈134M and ≈60M datapoints each) but also the languages that have the most entities/pages (≈6.1M and ≈2.4M). Therefore these

[4] Chinese, Czech, French, Italian, Polish, Portuguese, and Russian.

[5] Botha et al. (2020) did not release code for extracting Mewsli-9 from a Wikinews dump.

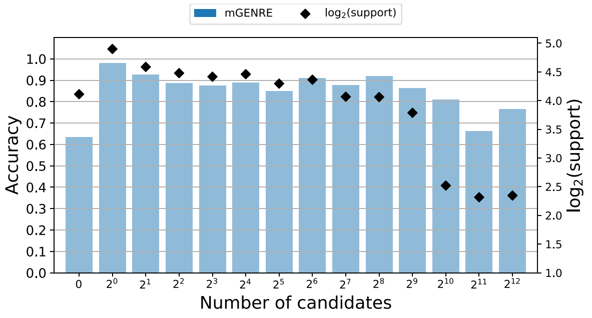


Figure 3: Results of mGENRE on Mewsli-9 by the number of retrieved candidates.

	Botha et al. (2020)		mGENRE	
Bin	Support	Acc.	Support	Acc.
[0, 1)	3,198	8.0	1,244	22.1
[1, 10)	6,564	58.0	5,777	47.3
[10, 100)	32,371	80.0	28,406	77.3
[100, 1k)	66,232	90.0	72,414	89.9
[1k, 10k)	78,519	93.0	84,790	93.2
[10k, +)	102,203	94.0	96,456	96.3
micro-avg	289,087	89.0	289,087	90.6
macro-avg	–	70.0	–	71.0

Table 3: Results on the Mewsli-9 dataset, by entity frequency in training. The support is slightly different because training data differ (i.e., the set of languages from Wikipedia is different).

are the hardest languages to link. When enabling candidate filtering to restrict the space for generation, we further improve error reduction to 13.6% and 21.0% for micro and macro average, respectively. Although candidate selection is not required by our general formulation, it definitely helps to restrict the search space when candidates are available (note that recall@k using all the candidates is >98% for all languages and on average using candidates reduces the search space form ≈20M entities to a few hundreds—e.g., see Figure 3 for a breakout of results by the number of retrieved candidates). Marginalization reduces the error by the same amount as candidate filtering but combining search with candidates and marginalization leads to our best model: It improves error reduction to 14.5% and 23.0% on micro and macro average, respectively. Our best model is also better than Model F$^+$ in English and on par with it in German.

TR2016$^{\text{hard}}$ and TAC-KBP2015 We compared our mGENRE against cross-lingual systems (Tsai and Roth, 2016; Sil et al., 2018; Upadhyay et al., 2018; Zhou et al., 2019) and Model F$^+$ by Botha et al. (2020) in Table 2. Differently from Meswli-9, the base mGENRE model does not outperform previous systems. Using marginalization brings minimal improvements. Instead, using candidates gives +11% absolute accuracy on TAC-KBP2015 and +5% on TR2016$^{\text{hard}}$, effectively making mGENRE state-of-the-art in both datasets. The role of candidates is very evident on TAC-KBP2015, where there is not much of a difference for Spanish but a +22% absolute accuracy for Chinese. TAC-KBP2015 comes with a training set and we used it to expand the candidate set. Additionally, we also included all simplified Chinese versions of the entity names because we used traditional Chinese in pre-training, and TAC-KBP2015 uses simplified Chinese. Many mentions in TAC-KBP2015 were not observed in Wikipedia, so the performance gain mostly comes from this but including the simplified and alternative Chinese names also played an important role (+5% comes from this alone).[6]

5.2 Analysis

By Entity Frequency Table 3 shows a breakdown of Mewsli-9 accuracy by entity frequency in training for Botha et al.'s (2020) Model F$^+$ and mGENRE. Interestingly, our model has much higher accuracy (22% vs 8%) on unseen entities (i.e., the [0,1) bin). This is because our formulation can take advantage of copying names from the source, translating them, or normalizing them. For example, an unseen person name should likely be linked to the entity with the same name. This powerful bias gives the model advantage in these cases. On very rare entities (i.e., the [1,10) bin) our model performs worse than Model F$^+$. Note that Model F$^+$ was trained specifically to tackle those cases (e.g., with hard negatives and frequency-based mini-batches) whereas our model was not. We argue that similar strategies can be applied to mGENRE to improve performance on rare entities, and we leave that to future work. The performance gap between Model F$^+$ and mGENRE

[6] We speculate that including different version (e.g., different dialects for Arabic) of entity names could improve performance in all languages. Since this is not in the scope of this paper, we will leave it for future work.

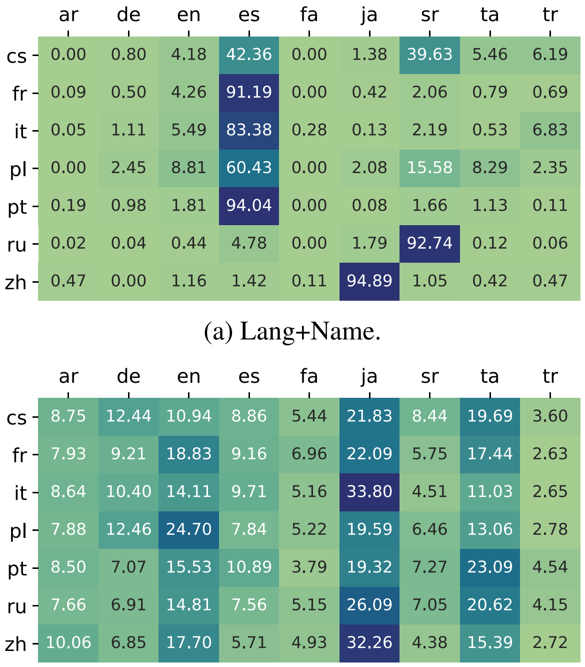
Lang.	**Can.**	**N+L**	**L+N**	**L+N^{M}**
cs	36.3	30.2	34.0	**69.7**
fr	62.9	57.0	53.3	**73.4**
it	44.8	43.7	42.9	**56.8**
pl	31.9	21.2	25.6	**68.8**
pt	60.8	61.7	59.5	**76.2**
ru	34.9	32.4	35.1	**65.8**
zh	35.1	41.1	44.0	**52.8**
micro-avg	41.6	38.3	39.5	**65.9**
macro-avg	43.8	41.0	42.1	**66.2**

Table 4: mGENRE on the Wikinew-7 unseen languages. Models are trained only on the Mewsli-9 languages (1M datapoints per language `ar`, `de`, `en`, `es`, `fa`, `ja`, `sr`, `ta`, and `tr`). 'Can.' is canonical, 'N+L' is 'name+language', and 'L+N' is the opposite. ^{M} indicates marginalization.

on entities that appear more than 100 times in the training set is minimal.

By Candidate Frequency We additionally measure the accuracy on Mewsli-9 by the number of candidates retrieved from the alias table (details in Figure 3). When there are no candidates (≈4% of Mewsli-9), an alias table would automatically fail, but mGENRE uses the entire KB as candidates and has 63.9% accuracy. For datapoints with few candidates (e.g., less than 100), we could use mGENRE as a *ranker* and score all of the options without relying on constrained beam search. However, this approach would be computationally infeasible when there are no candidates (i.e., we use all the KB as candidates) or too many candidates (e.g., thousands). Constrained BS allows us to efficiently explore the space of entity names, whatever the number of candidates.

Unseen Languages We use our Wikinews-7 dataset to evaluate mGENRE capabilities to deal with languages not seen during training (i.e., the set of languages in train and test are disjoint). This zero-shot setting implies that no mention table is available during inference; hence we do not consider candidates for test mentions. We train our models on the nine Mewsli-9 languages and compare all strategies exposed in Section 3. To make our ablation study feasible, we restrict the training data to the first 1 million hyperlinks from Wikipedia abstracts. Results are in Table 4.



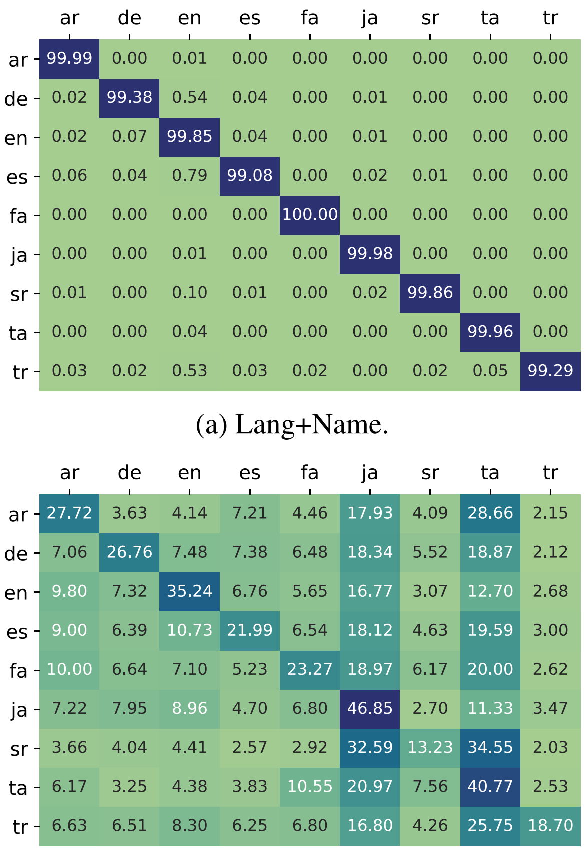
(a) Lang+Name.

(b) Lang+Name^{M}.

Figure 4: Distribution of languages on the top-1 prediction of two mGENRE models on Wikinews-7 (test set). The y-axis indicates the source language (unseen at training time) where the x-axis indicates the language (seen at training time) of the first prediction. Note that the models are only trained on `ar`, `de`, `en`, `es`, `fa`, `ja`, `sr`, `ta`, and `tr`.

Using our novel marginalization strategy that aggregates (both at training and inference time) over all seen languages to perform the linking brings an improvement of over 50% with respect to considering a single language. To investigate more deeeply the behavior of the model in this setting, we compute the probability mass distribution over languages seen at training time for the top-1 prediction (reported in Figure 4). When marginalization is enabled (Figure 4b), the distribution is more spread across languages because the model is trained to use all of them. Hence the model can exploit connections between an unseen language and all seen languages for the linking process drastically increases accuracy.

Marginalization is effective for this zero-shot setting, but it has a minimal impact in the standard setting (e.g., Tables 1 and 2). When a model has seen the source language at training time it mainly makes use of that to perform a prediction (i.e., the target prediction is in the source language most of the time—>99%, see Figure 5a). Instead, when the source language is never seen during



(a) Lang+Name.

(b) Lang+Name$^{\mathrm{M}}$.

Figure 5: Distribution of languages on the top-1 prediction of two mGENRE models on Mewsli-9. The y-axis indicates the source language and the x-axis indicates the language of the top-1 prediction. The models trained on those languages.

training, by marginalization the model can exploit similarities with all seen languages. Indeed, even though marginalization and canonical representation are the top-two systems in the unseen languages setting, they are not on seen languages on the same setting: In Table 6 we report the results of all these strategies also on the seen languages (Mewsli-9 test set). Complementary to Figure 4, we also report the probability mass distribution over languages seen for Mewsli-9 in Figure 5.

Memory Footprint As computational cost and memory footprints are important aspects of modeling, we compared the number of parameters used by mGENRE and the best competing mEL system by Botha et al. (2020). Their model has ≈73M parameters and ≈15B entity parameters for a total memory usage of ≈61GB, where mGENRE has ≈406M model parameters and no entity parameters (i.e., we just have a prefix tree with entity names that occupies ≈2.2GB), for a total of ≈6GB memory usage (i.e., ≈10 times less memory).

Bin	Support	Acc.
[0, 1)	14,741	66.7
[1, 10)	15,279	88.1
[10, 100)	43,169	92.0
[100, 1k)	75,927	91.7
[1k, 10k)	80,329	91.5
[10k, 100k)	47,944	93.6
[100k, 1M)	11,460	93.0
[1M, 10M)	238	73.2

Table 5: mGENRE results on Mewsli-9 dataset by mention frequency in training.

Examples In Table 7 we report some examples of correct and wrong predictions of our mGENRE L+N and N+L on selected datapoints from Mewsli-9. Examples are picked to highlight specific behaviors of our models. We show an example of the copying mechanisms (i.e., the N+L model normalizes the mention but L+N fails to do so) as well as an example where the model memorized an acronym (i.e., ''MDC'' as Movement for Democratic Change) and outputs that correctly in the case of L+N and wrongly for N+L.

By Mention Frequency We show a breakdown of the accuracy of mGENRE on Mewsli-9 by mention frequency in Table 5. The accuracy of unseen mentions is 66.7% and increases up to 93.6% for mentions seen more than 10k times. For extremely common mentions (i.e., seen more than 1M times) the accuracy drops to 73.2%. These mentions correspond to entities that are harder to disambiguate (e.g., 'United States' appears 3.2M times but can be linked to the country as well as any sports team where the context refers to sports).

6 Related Work

The most related work to ours are De Cao et al. (2021), who proposed using an autoregressive language model for monolingual EL, and Botha et al. (2020), who proposed extending the cross-lingual EL task to multilingual EL with a language-agnostic KB. We provide an outline of the GENRE model proposed by De Cao et al. (2021) in Section 2.2 and 2.3. GENRE was applied not only to EL but also for joint mention detection

Lang.	Can.	N+L	L+N	L+N^{M}
ar	90.5	92.8	**92.9**	89.2
de	**84.6**	86.4	86.4	85.3
en	77.6	**79.3**	79.2	76.5
es	83.4	**85.5**	85.2	83.4
fa	91.6	90.7	**91.8**	88.2
ja	81.3	82.3	**82.8**	81.3
sr	91.5	92.7	**92.9**	92.5
ta	**92.8**	91.8	91.9	91.3
tr	**88.0**	87.7	87.3	86.0
micro-avg	83.20	84.77	**84.80**	83.05
macro-avg	86.82	87.68	**87.82**	85.97
		+ candidates		
ar	94.4	94.5	**94.7**	93.0
de	89.4	**89.8**	**89.8**	89.3
en	83.6	83.8	**83.9**	82.4
es	87.7	88.2	**88.3**	87.3
fa	**93.6**	93.3	**93.6**	93.3
ja	87.9	88.0	**88.4**	87.9
sr	93.1	93.4	**93.5**	93.2
ta	**93.0**	92.2	92.5	92.5
tr	**91.1**	90.4	89.9	89.1
micro-avg	87.95	88.22	**88.32**	87.43
macro-avg	90.42	90.41	**90.51**	89.78

Table 6: mGENRE on the Mewsli-9. Models are trained only on the Mewsli-9 languages (1M datapoints per language). 'Can.' is canonical, 'N+L' is 'name+language' and 'L+N' is the opposite. ^{M} indicates marginalization.

and entity linking (still with an autoregressive formulation) as well as to page-level document retrieval for fact-checking, open-domain question answering, slot filling, and dialog (Petroni et al., 2021). Botha et al.'s (2020) Model F^{+} is a *bi-encoder* model: It is based on two BERT-based (Devlin et al., 2019) encoders that output vector representations for contet and entities. Similar to Wu et al. (2020) they rank entities with a dot-product between these representations. Model F^{+} uses the description of entities as input to the entity encoder and title, document, and mention (separated with special tokens) as inputs to the context encoder. Bi-encoder solutions may be memory inefficient because they require keeping in memory large matrices of embeddings, although memory-efficient dense retrieval has recently received attention (Izacard et al., 2020; Min et al., 2021; Lewis et al., 2021).

Input: Police in Zimbabwe have stopped opposition leader Morgan Tsvangirai (**[START] MDC [END]**) en route to a campaign rally. His convoy was then escorted to a police station in Esigodini.
Correct by L+N:
en ≫ Movement for Democratic Change$^{\text{Q6926644}}$
Wrong by N+L:
People's Democratic Party (Zimbabwe) ≫ en$^{\text{Q48798212}}$

Input: Sin embargo, la promoción del **[START] abstencionismo [END]** por parte de los opositores se tradujo en una participación de apenas el 47.32%, alrededor de 9.2 millones de electores. En las municipales de 2013 habían participado 58.92% de los venezolanos con derecho a voto. 61% en los comicios regionales de octubre.
Wrong by L+N: es >> Oposición (política)$^{\text{Q192852}}$
Correct by N+L: Abstención >> es$^{\text{Q345321}}$

Table 7: Examples of correct and wrong predictions of our mGENRE models L+N and N+L on selected samples from Mewsli-9.

Another widely explored line of work is Cross-Language Entity Linking (XEL; McNamee et al., 2011; Cheng and Roth, 2013). XEL considers contexts in different languages while mapping mentions to entities in a monolingual KB (e.g., English Wikipedia). Tsai and Roth (2016) used alignments between languages to train multilingual entity embeddings. They used candidate selection and then they re-rank them with an SVM using these embeddings as well as a set of features (based on the multilingual title, mention, and context tokens). Sil et al. (2018) explored the use of more sophisticated neural models for XEL, and Upadhyay et al. (2018) jointly modeled type information to boost performance. Zhou et al. (2019) propose improvements to both entity candidate generation and disambiguation to make better use of the limited data in low-resource scenarios. Note that in this work we focus on *multilingual* EL, not cross-lingual. XEL is limits to a monolingual KB (usually English), where MEL is more general because it can link to entities that might not be necessarily represented in the target monolingual KB but in any of the available languages.

7 Conclusion

In this work, we propose an autoregressive formulation to the multilingual entity linking problem. For a mention in a given language, our solution generates entity names left-to-right and token-by-token. The resulting system maintains entity names in as many languages as possible to exploit language connections and interactions between source mention context and target entity name. The constrained beam search decoding strategy enables fast search within a large set of entity names (e.g., the whole KB in multiple languages) with no need for large-scale dense indices. We additionally design a novel objective that marginalizes over all available languages to perform a prediction. We show that this strategy is effective in dealing with languages for which no training data are available (i.e., 50% improvement for languages never seen during training). Overall, our experiments show that mGENRE achieves new state-of-the-art performance on three popular multilingual entity linking datasets.

Acknowledgments

The authors thank Patrick Lewis and Aleksandra Piktus for helpful discussions and technical support.

References

Akari Asai, Kazuma Hashimoto, Hannaneh Hajishirzi, Richard Socher, and Caiming Xiong. 2020. Learning to retrieve reasoning paths over wikipedia graph for question answering. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Giuseppe Attardi. 2015. Wikiextractor. https://github.com/attardi/wikiextractor.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: A collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, pages 1247–1250. https://doi.org/10.1145/1376616.1376746

Antoine Bordes, Y-Lan Boureau, and Jason Weston. 2017. Learning end-to-end goal-oriented dialog. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Jan A. Botha, Zifei Shan, and Daniel Gillick. 2020. Entity linking in 100 languages. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7833–7845, Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-main.630

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.

Henry Y. Chen, Ethan Zhou, and Jinho D. Choi. 2017. Robust coreference resolution and entity linking on dialogues: Character identification on TV show transcripts. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 216–225, Vancouver, Canada. Association for Computational Linguistics. https://doi.org/10.18653/v1/K17-1023

Xiao Cheng and Dan Roth. 2013. Relational inference for wikification. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1787–1796, Seattle, Washington, USA. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.acl-main.747

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.

Amanda Cercas Curry, Ioannis Papaioannou, Alessandro Suglia, Shubham Agarwal, Igor Shalyminov, Xinnuo Xu, Ondřej Dušek, Arash Eshghi, Ioannis Konstas, Verena Rieser, and

Oliver Lemon. 2018. Alana v2: Entertaining and informative open-domain social dialogue using ontologies and entity linking. *Alexa Prize Proceedings*.

Nicola De Cao, Wilker Aziz, and Ivan Titov. 2019. Question answering by reasoning across documents with graph convolutional networks. 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 2306–2317, Minneapolis, Minnesota. Association for Computational Linguistics. https://doi.org/10.18653/v1/N19-1240

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

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 *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 277–285, Beijing, China. Coling 2010 Organizing Committee.

Octavian-Eugen Ganea and Thomas Hofmann. 2017. Deep joint entity disambiguation with local neural attention. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2619–2629, Copenhagen, Denmark. Association for Computational Linguistics.

Johannes Hoffart, Mohamed Amir Yosef, Ilaria Bordino, Hagen Fürstenau, Manfred Pinkal, Marc Spaniol, Bilyana Taneva, Stefan Thater, and Gerhard Weikum. 2011. Robust disambiguation of named entities in text. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 782–792, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Gautier Izacard, Fabio Petroni, Lucas Hosseini, Nicola De Cao, Sebastian Riedel, and Edouard Grave. 2020. A memory efficient baseline for open domain question answering. *arXiv preprint arXiv: 2012.15156*.

Heng Ji, Joel Nothman, Ben Hachey, and Radu Florian. 2015. Overview of TAC-KBP2015 tri-lingual entity discovery and linking. In *TAC*.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Robert Leaman and Graciela Gonzalez. 2008. BANNER: An executable survey of advances in biomedical named entity recognition. In *Pacific Symposium on Biocomputing 2008, PSB 2008*, Pacific Symposium on Biocomputing 2008, PSB 2008, pages 652–663. 13th Pacific Symposium on Biocomputing, PSB 2008; Conference date: 04-01-2008 Through 08-01-2008.

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. https://doi.org/10.18653/v1/2020.acl-main.703

Patrick Lewis, Yuxiang Wu, Linqing Liu, Pasquale Minervini, Heinrich Küttler, Aleksandra Piktus, Pontus Stenetorp, and Sebastian Riedel. 2021. PAQ: 65 million probably-asked questions and what you can do with them. *Transactions of the Association for Computational Linguistics*, 9:1098–1115. https://doi.org/10.1162/tacl_a_00415

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742. https://doi.org/10.1162/tacl_a_00343

Paul McNamee, James Mayfield, Dawn Lawrie, Douglas Oard, and David Doermann. 2011. Cross-language entity linking. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 255–263, Chiang Mai, Thailand. Asian Federation of Natural Language Processing.

Sewon Min, Jordan Boyd-Graber, Chris Alberti, Danqi Chen, Eunsol Choi, Michael Collins, Kelvin Guu, Hannaneh Hajishirzi, Kenton Lee, Jennimaria Palomaki, Colin Raffel, Adam Roberts, Tom Kwiatkowski, Patrick Lewis, Yuxiang Wu, Heinrich Küttler, Linqing Liu, Pasquale Minervini, Pontus Stenetorp, Sebastian Riedel, Sohee Yang, Minjoon Seo, Gautier Izacard, Fabio Petroni, Lucas Hosseini, Nicola De Cao, Edouard Grave, Ikuya Yamada, Sonse Shimaoka, Masatoshi Suzuki, Shumpei Miyawaki, Shun Sato, Ryo Takahashi, Jun Suzuki, Martin Fajcik, Martin Docekal, Karel Ondrej, Pavel Smrz, Hao Cheng, Yelong Shen, Xiaodong Liu, Pengcheng He, Weizhu Chen, Jianfeng Gao, Barlas Oguz, Xilun Chen, Vladimir Karpukhin, Stan Peshterliev, Dmytro Okhonko, Michael Schlichtkrull, Sonal Gupta, Yashar Mehdad, and Wen-tau Yih. 2021. Neurips 2020 efficientqa competition: Systems, analyses and lessons learned. In *Proceedings of the NeurIPS 2020 Competition and Demonstration Track,* volume 133 of *Proceedings of Machine Learning Research*, pages 86–111. PMLR.

Yixin Nie, Songhe Wang, and Mohit Bansal. 2019. Revealing the importance of semantic retrieval for machine reading at scale. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2553–2566, Hong Kong, China. Association for Computational Linguistics. https://doi.org/10.18653/v1/D19-1258

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. https://doi.org/10.18653/v1/N19-4009

Fabio Petroni, Aleksandra Piktus, Angela Fan, Patrick Lewis, Majid Yazdani, Nicola De Cao, James Thorne, Yacine Jernite, Vladimir Karpukhin, Jean Maillard, Vassilis Plachouras, Tim Rocktäschel, and Sebastian Riedel. 2021. KILT: A benchmark for knowledge intensive language tasks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2523–2544, Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.naacl-main.200

Karin Sevegnani, David M. Howcroft, Ioannis Konstas, and Verena Rieser. 2021. OTTers: One-turn topic transitions for open-domain dialogue. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2492–2504, Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.acl-long.194

Avirup Sil, Gourab Kundu, Radu Florian, and Wael Hamza. 2018. Neural cross-lingual entity linking. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 5464–5472. AAAI Press.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958.

Ilya Sutskever, James Martens, and Geoffrey E. Hinton. 2011. Generating text with recurrent neural networks. In *Proceedings of the 28th International Conference on Machine Learning, ICML 2011, Bellevue, Washington, USA, June 28 - July 2, 2011*, pages 1017–1024. Omnipress.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 3104–3112.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 2818–2826. IEEE Computer Society. https://doi.org/10.1109/CVPR.2016.308

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, San Diego, California. Association for Computational Linguistics. https://doi.org/10.18653/v1/N16-1072

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. https://doi.org/10.18653/v1/D18-1270

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: A free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85. https://doi.org/10.1145/2629489

Tsung-Hsien Wen, David Vandyke, Nikola Mrkšić, Milica Gašić, Lina M. Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. 2017. A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 438–449, Valencia, Spain. Association for Computational Linguistics.

Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. 2020. CCNet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4003–4012, Marseille, France. European Language Resources Association.

Jason D. Williams, Kavosh Asadi, and Geoffrey Zweig. 2017. Hybrid code networks: Practical and efficient end-to-end dialog control with supervised and reinforcement learning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 665–677, Vancouver, Canada. Association for Computational Linguistics. https://doi.org/10.18653/v1/P17-1062

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.

Ikuya Yamada, Hiroyuki Shindo, Hideaki Takeda, and Yoshiyasu Takefuji. 2016. Joint learning of the embedding of words and entities for named entity disambiguation. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 250–259, Berlin, Germany. Association for Computational Linguistics. https://doi.org/10.18653/v1/K16-1025

Jin G. Zheng, Daniel Howsmon, Boliang Zhang, Juergen Hahn, Deborah McGuinness, James Hendler, and Heng Ji. 2015. Entity linking for biomedical literature. *BMC Medical Informatics and Decision Making*, 15(1):1–9. https://doi.org/10.1186/1472-6947-15-S1-S4, PubMed: 26045232

Shuyan Zhou, Shruti Rijhwani, and Graham Neubig. 2019. Towards zero-resource cross-lingual entity linking. In *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019)*, pages 243–252, Hong Kong, China. Association for Computational Linguistics. https://doi.org/10.18653/v1/D19-6127

A Experimental Details

A.1 Pre-training

We used a pre-trained mBART (Lewis et al., 2020; Liu et al., 2020) model on 125 languages—see Figure 6 for a visual overview of the overlap with these languages, Wikipedia, and the languages used by Botha et al. (2020). mBART has 24 layers of hidden size is 1,024 and it has a total of 406M parameters. We pre-trained on an extended version of the `cc100` (Conneau et al., 2020; Wenzek et al., 2020) corpora available here[7] where we increased the number of common crawl snapshots for low resource languages from 12 to 60. The dataset has ≈5TB of text. We pre-trained for 500k steps with max 1,024 tokens per GPU on a variable batch size (≈3000).

A.2 Data for Supervision

Wikidata Wikidata contains tens of millions of items but most of them are scholarly articles or they correspond to help and template pages in Wikipedia (i.e., not entities we want to retain).[8] Following Botha et al. (2020), we only keep Wikidata items that have an associated Wikipedia page in at least one language, independent of the languages we actually model. Moreover, we filter out items that are a subclass (P279) or instance of (P31) some Wikimedia organizational entities (e.g., help and template pages—see Table 8).

Wikipedia We aligned each Wikipedia hyperlink to its respective Wikidata item using a custom script. Note that each Wikipedia page maps to a Wikidata item. For the alignment we use i) direct reference when the hyperlink point directly to a Wikipedia page, ii) a re-directions table if the hyperlink points to an alias page, and iii) a Wikidata search among labels and aliases of items if the previous two alignment strategies failed. The previous two alignment strategies might fail when i) authors made a mistake linking on a non-existing page, ii) authors linked to a non-existing page on purpose hoping it would be created in the future, or iii) the original title of a page changed over time and no redirection was added to accommodate old hyperlinks. This procedure successfully aligns 91% of the hyperlinks. We only keep unambiguous alignments because, when using Wikidata search (i.e., the third alignment strategy), the mapping could be ambiguous (e.g., multiple items may share the same labels and aliases). We use a standard Wikipedia extractor `wikiextractor`[9] by Attardi (2015) and a redirect extractor.[10] We use both Wikipedia and Wikidata dumps from 2019-10-01.

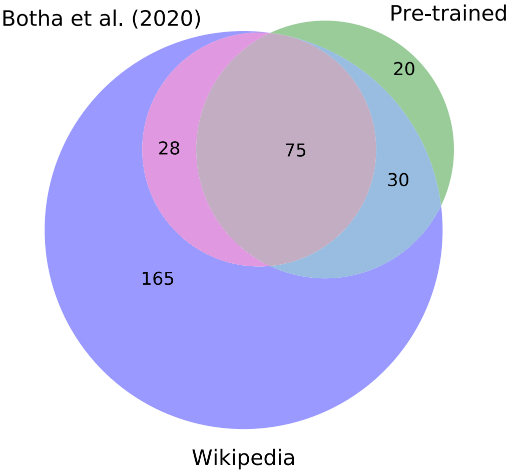


Figure 6: Venn diagram of the overlap of languages used during multilingual language modeling (pre-training), the languages available on Wikipedia (as of 2019-10-01), and the languages used by Botha et al. (2020). After pre-training on 125 languages, we fine-tune on the 105 that overlap with the one available in Wikipedia.

A.3 Data for Test

Mewsli-9 (Botha et al., 2020) contains 289,087 entity mentions appearing in 58,717 originally written news articles from Wikinews, linked to WikiData. The corpus includes documents in 9 languages.[11] Differently from the cross-lingual setting, this is a truly multilingual dataset since 11% target entities in Mewsli-9 do not have an English Wikipedia page.

TR2016$^{\textbf{hard}}$ (Tsai and Roth, 2016) is a Wikipedia based cross-lingual dataset specifically constructed to contain difficult mention-entity pairs. Authors extracted Wikipedia hyperlinks for which the corresponding entity is not the most likely when using an alias table. Because we train on Wikipedia, to avoid an overlap with this test data, we removed all mentions from our

[7] `http://data.statmt.org/cc-100.`

[8] `https://www.wikidata.org/wiki/Wikidata:Statistics.`

[9] `https://github.com/attardi/wikiextractor.`

[10] `https://code.google.com/archive/p/wikipedia-redirect.`

[11] Arabic, English, Farsi, German, Japanese, Serbian, Spanish, Tamil, and Turkish.

Wikidata ID	Label
Q4167836	category
Q24046192	category stub
Q20010800	user category
Q11266439	template
Q11753321	navigational template
Q19842659	user template
Q21528878	redirect page
Q17362920	duplicated page
Q14204246	project page
Q21025364	project page
Q17442446	internal item
Q26267864	KML file
Q4663903	portal
Q15184295	module

Table 8: Wikidata identifiers used for filtering out items from Botha et al. (2020).

			Entities	
Lang.	**Docs**	**Mentions**	Distinct	$\notin$ EnWiki
ru	1,625	20,698	8,832	1,838
it	907	8,931	4,857	911
pl	1,162	5,957	3,727	547
fr	978	7,000	4,093	349
cs	454	2,902	1,974	200
pt	666	2,653	1,313	113
zh	395	2,057	1,274	110
Total	6,187	50,198	26,070	4,068

Table 9: Corpus statistics for the Wikinews unseen languages we use as an evaluation set.

training data that also appear in TR2016$^{\text{hard}}$. Note that this pruning strategy is more aggressive than Tsai and Roth's (2016) and Botha et al.'s (2020) strategies. Tsai and Roth (2016) ensured not having mention-entity pairs overlap between training and test, but a mention (with a different entity) might appear in training. Botha et al. (2020)[12] split at the page-level only, making sure to hold out all Tsai and Roth (2016) test pages (and their corresponding pages in other languages), but they trained on any mention-entity pair that could be extracted from their remaining training page partition (i.e., they have overlap between training and text entity-mention pairs). To compare with previous work (Tsai and Roth, 2016; Upadhyay et al., 2018; Botha et al., 2020) we only evaluate on German, Spanish, French, and Italian (a total of 16,357 datapoints).

TAC-KBP2015 To evaluate our system on documents out of the Wikipedia domain, we experiment on the TAC-KBP2015 Tri-Lingual Entity Linking Track (Ji et al., 2015). To compare with previous work (Tsai and Roth, 2016; Upadhyay et al., 2018; Sil et al., 2018; Zhou et al., 2019), we use only Spanish and Chinese (i.e., we do not evaluate in English). Following previous work, we only evaluate *in-KB* links (Yamada et al., 2016; Ganea and Hofmann, 2017), that is, we do not evaluate on mentions that link to entities out of the KB. Previous work considered Freebase (Bollacker et al., 2008) as KB, and thus we computed a mapping between Freebase ID and Wikidata ID. When we cannot solve the match, our system results in a zero score (i.e., it counts as a wrong prediction). TAC-KBP2015 contains 166 Chinese documents (84 news and 82 discussion forum articles) and 167 Spanish documents (84 news and 83 discussion forum articles) for a total of 12,853 mention-entity datapoints.

A.4 Training

We implemented, trained, and evaluated our model using the `fairseq` library (Ott et al., 2019). We trained mGENRE using Adam (Kingma and Ba, 2015) with a learning rate 10^{-4}, $\beta_1 = 0.9$, $\beta_2 = 0.98$, and with linear warm-up for 5,000 steps followed by liner decay for maximum 2M steps. The objective is sequence-to-sequence categorical cross-entropy loss with 0.1 of label smoothing and 0.01 of weight decay. We used dropout probability of 0.1 and attention dropout of 0.1. We used max 3,072 tokens per GPU and variable batch size (≈12,500). Training was done on 384 GPUs (Tesla V100 with 32GB of memory) and it completed in ≈72h for a total of ≈27,648 GPU hours or ≈1,152 GPU days. Since TAC-KBP2015 contains noisy text (e.g., XML/HTML tags), we further fine-tune mGENRE for 2k steps on its training set when testing on it.

[12] Information provided by private correspondence with the authors.

A.5 Inference

At test time, we use Constrained Beam Search with 10 beams, length penalty of 1, and maximum decoding steps of 32. We restrict the input sequence to be at most 128 tokens cutting the left, right, or both parts of the context around a mention. When applying marginalization, we normalize the log-probabilities by sequence length using $\log p(y|x)/L^{\alpha}$, where $\alpha = 0.5$ was tuned on the development set.