Cross-Lingual Transfer in Zero-Shot Cross-Language Entity Linking

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

Abstract

Cross-language entity linking grounds mentions written in several languages to a monolingual knowledge base. We use a simple neural ranking architecture for this task that uses multilingual BERT representations of both the mention and the context as input, so as to explore the ability of a transformer model to perform well on this task. We find that the multilingual ability of BERT leads to good performance in monolingual and multilingual settings. Furthermore, we explore zero-shot language transfer and find surprisingly robust performance. We conduct several analyses to identify the sources of performance degradation in the zero-shot setting. Results indicate that while multilingual transformer models transfer well between languages, issues remain in disambiguating similar entities unseen in training.

1 Introduction

Entity linking grounds named entities mentioned in text, such as *Chancellor*, to a reference knowledge base (KB) or ontology entry, such as *Angela Merkel*. Historically, entity linking work focused on English documents and knowledge bases, but subsequent work expanded the task to consider multiple languages (McNamee et al., 2011). In cross-language entity linking, entities in a set of multilingual documents is linked to a KB in a single language. The TAC KBP shared task (Ji et al., 2015), for example, links mentions in Chinese and Spanish documents with an English KB. Success in building cross-language linking systems can be helpful in tasks such as discovering all documents relevant to an entity, regardless of language.

Successfully linking a mention across languages requires adapting several common entity linking components to the cross-language setting. Consider the example in Figure 1, which contains the Spanish mention *Oficina de la Presidencia*, a reference to the entity *President of Mexico* in an English KB. To link the mention to the relevant entity we must compare the mention text and its surrounding textual context in Spanish to the English entity name and entity description, as well as compare the mention and entity type. Previous work has focused on transliteration or translation approaches for name and context (McNamee et al., 2011; Pan et al., 2015), or leveraging large amounts of cross-language information (Tsai and Roth, 2016) and multilingual embeddings (Upadhyay et al., 2018).

Since this work emerged, there have been major advances in multilingual NLP (Wu and Dredze, 2019; Pires et al., 2019). Mainstream approaches to multilingual learning now use multilingual encoders, trained on raw text from multiple languages (Devlin et al., 2019). These models, such as multilingual BERT or XMLR (Conneau et al., 2019), have achieved impressive results on a range of multilingual NLP tasks, including part of speech tagging (Tsai et al., 2019), parsing (Wang et al., 2019; Kondratyuk and Straka, 2019), and semantic similarity (Lo and Simard, 2019; Reimers and Gurevych, 2019).

We propose to leverage text representations with multilingual BERT (Devlin et al., 2019) for cross-language entity linking to handle the mention text, entity name, mention context and entity description[1]. We use a neural ranking objective and a deep learning model to combine these representations, along with a one-hot embedding for the entity and mention type, to produce a cross-language linker. We use this ranking architecture to highlight the ability of mBERT to perform on this task without a more complex architecture. Although previous work tends to use multilingual encoders for one language at a time, *e.g.,* train a Spanish NER system

[1]Our code is available at https://github.com/elliotschu/crosslingual-el

Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 583–595

with mBERT, we ask: can our model effectively link entities *across* languages? We find that, somewhat surprisingly, our approach does exceedingly well; scores are comparable to previously reported best results that are trained on data not available to our model (they have access to non-English names). Next, we consider a multilingual setting, in which a single system is simultaneously trained to link mentions in multiple languages to an English KB. Previous work (Upadhyay et al., 2018) has shown that multilingual models can perform robustly on cross-language entity linking. Again, we find that, surprisingly, a model trained on multiple languages at once does about as well, or in some cases better, than the same model trained separately on every language.

These encouraging results lead us to explore the challenging task of zero-shot training, in which we train a model to link single language documents (*e.g.,* English) to an English KB, but apply it to unseen languages (*e.g.,* Chinese) documents. While the resulting model certainly does worse on a language that is unobserved, the reduction in performance is remarkably small. This result leads us to ask: 1) Why do zero-shot entity linking models do so well? 2) What information is needed to allow zero-shot models to perform as well as multilingually trained models? Using a series of ablation experiments we find that correctly comparing the mention text and entity name is the most important component of an entity linking model. Therefore, we propose an auxiliary pre-training objective to improve zero-shot performance. However, we find that this text-focused approach does not improve performance significantly. Rather, we find that much of the remaining loss comes not from the language transfer, but from mismatches of entities mentioned across the datasets. This suggests that future work on the remaining challenges in zero-shot entity linking should focus on topic adaptation, instead of improvements in cross-lingual representations.

In summary, this paper uses a simple ranker to explore effective cross-language entity linking with multiple languages. We demonstrate its effectiveness at zero-shot linking, evaluate a pre-training objective to improve zero-shot transfer, and lay out guidelines to inform future research on zero-shot linking.

2 Cross-Language Entity Linking

A long line of work on entity linking has developed standard models to link textual mentions to entities in a KB (Dredze et al., 2010; Durrett and Klein, 2014; Gupta et al., 2017). The models in this area have served as the basis for developing multilingual and cross-language entity linking systems, and they inform our own model development. We define **multilingual** to mean a model that can operate on mentions from more than one language at the same time (link both English and Chinese mentions to an ontology) and **cross-language** to refer to linking mentions in one language (*e.g.,* Spanish) to an ontology in another (*e.g.,* English).

A common approach to cross-language entity linking is to use transliteration data to transform non-English mentions into English strings. Early transliteration work (McNamee et al., 2011) uses a transliteration corpus to train a support vector machine ranker, which uses common entity linking features such as name and context matching, co-occurring entities, and an indicator for NIL (no matching candidate.) Pan et al. (2017) uses transliteration data for a set of 282 languages to generate all possible combinations of mentions. A related approach is to use machine translation to translate a document into English, and then use an English entity linker. However, an MT system may not be available, and it further needs a specialized name module to properly translate entity names. Several systems from the TAC 2015 KBP Entity Discovery and Linking task (Ji et al., 2015) translate non-English documents into English, then use standard Entity Linking systems.

Cross-language Wikification is a closely related task, which uses links within Wikipedia, combined with equivalent pages in other languages to train an entity linker with Wikipedia as the KB. This approach typically uses English Wikipedia as the KB, though it could use a KB in other languages. Tsai and Roth (2016) use a two-step linking approach, first using an IR-based triage system (which we also use). Second, they use a candidate ranking step based on a linear ranking SVM model with several features, including contextual, document, and coreference.

The most closely related work to our own is that of Upadhyay et al. (2018), who use multilingual embeddings as the basis for their representations, and Wikipedia as training data. They use FastText (Bojanowski et al., 2017; Smith et al., 2017)

... el jefe de la **Oficina de la Presidencia** *(m.01p1k, ORG)*, Aurelio Nuño y ...

name	*President of Mexico* (m.01p1k)
desc.	*The President of the United ...*
type	government_office

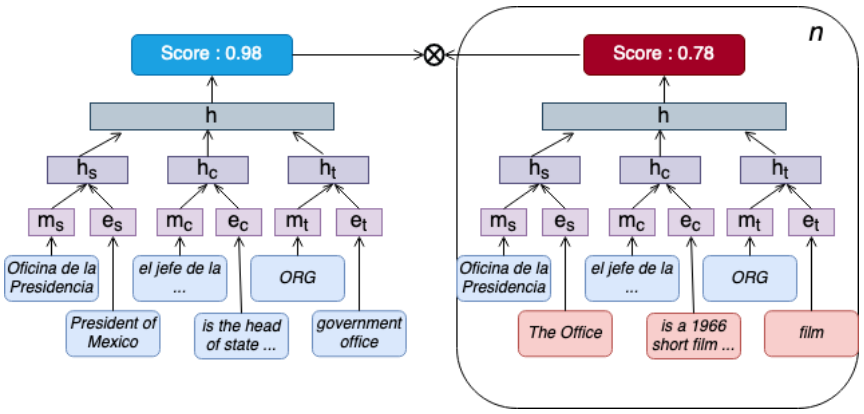

Figure 1: Example Spanish mention *Oficina de la Presidencia*, which is a link to entity *President of Mexico*, and the architecture for our neural ranker, using that example and a negatively-sampled entity *The Office*.

to align embeddings across languages, and a small dictionary to identify alignments. They pass these representations through a convolutional neural network to create a mention representation. They in turn use the other mention representations in the document to create a contextual representation, and also use a separate type vector. They train their network on hyperlinks from multiple languages in Wikipedia. Before the ranking step, they use a triage system similar to that of Tsai and Roth (2016). They evaluate on several entity linking datasets, including **TAC**. As their system only uses English Wikipedia as the KB, they set all mentions that link to a entity outside of Wikipedia to NIL; this results in a different evaluation setup than we need for our work. Their results show that training on all languages, instead of monolingual or bilingual training, generally performs best. For zero-shot entity linking, they train on English language Wikipedia. They find that their performance is heavily dependent on a prior probability derived from the triage system – otherwise, there is a large drop in performance.

Rijhwani et al. (2019) investigate zero-shot entity linking on low-resource languages. They propose a model consisting of a similarity model using encoders separately trained on high-resource language mentions, related to the low-resource language, and English entities. They then use the high-resource language as a pivot language for low resource language mentions, allowing them to score mentions in an unseen language. Raiman and Raiman (2018) consider multilingual entity linking, in which they use a KB in the same language as the mention, but exploit multilingual transfer for the model's type system. They formulate a type system as a mixed integer problem, which they use to learn a type system from knowledge graph relations.

3 Entity Linking Model

We propose a cross-language entity linker based on a pointwise neural ranker that scores a mention m and entity e pair, adapting from an architecture discussed in Dehghani et al. (2017). Unlike a classification architecture, a ranking architecture is able to score previously unseen entities. As is standard, we use a two stage system: triage followed by ranking; this reduces the number of entities that must be ranked, and results in better performance. Our system is shown in Figure 1. We select this architecture so as to focus on the ability of multilingual transformers to handle this task.

The ranker takes as input information about the mention and entity: 1) the mention string and entity name; 2) the context of the mention and entity description; and 3) the types of the mention and entity. We represent the mention string, entity name, mention context and entity description using a pre-trained multilingual deep transformer encoder (Devlin et al., 2019), while the mention and entity type are represented as one-hot embeddings. We describe the multilingual representation, model architecture and training procedure.

3.1 Multilingual Representations

We use multilingual BERT (mBERT) (Devlin et al., 2019)[2], which has been shown to create effective multilingual representations for downstream NLP tasks (Wu and Dredze, 2019). Consider the Spanish example in Figure 1. First, we create a represen-

[2]We found that XLM-R (Conneau et al., 2019) performed similarly and only report results on mBERT.

en	NN	0.195	0.463	0.550	0.502
	Mono	0.586	**0.703**	0.619	**0.658**
	MultiDS	0.509	0.873	0.478	0.618
	Multi	**0.602**	0.691	**0.626**	0.655
	MultiOr	*0.654*	*0.773*	*0.641*	*0.703*
	Tri	—	*0.736*	*0.738*	*0.737*
zh	NN	0.207	0.889	0.449	0.597
	Mono	0.709	**0.867**	0.728	0.791
	MultiDS	**0.733**	**0.867**	**0.746**	**0.801**
	Multi	0.730	0.862	0.735	0.793
	MultiOr	*0.828*	*0.950*	*0.812*	*0.876*
	Tri	—	*0.854*	*0.809*	*0.831*
es	NN	0.214	0.508	0.552	0.529
	Mono	0.595	**0.921**	0.587	0.714
	MultiDS	0.604	0.918	0.590	0.718
	Multi	**0.652**	0.918	**0.625**	**0.744**
	MultiOr	*0.691*	*0.936*	*0.655*	*0.770*
	Tri	—	*0.804*	*0.804*	*0.804*
Model		**micro**	**prec.**	**recall**	**F_1**
ar	NN	0.171	0.414	0.602	0.491
	Mono	**0.660**	**0.683**	**0.816**	**0.743**
	Multi	0.637	0.661	0.778	0.715
fa	NN	0.330	0.694	0.734	0.714
	Mono	0.702	0.780	0.881	0.827
	Multi	**0.762**	**0.817**	**0.919**	**0.863**
ko	NN	0.269	0.816	0.597	0.690
	Mono	0.752	0.832	0.861	0.846
	Multi	**0.805**	**0.850**	**0.902**	**0.875**
ru	NN	0.358	0.841	0.529	0.649
	Mono	0.694	0.834	0.843	0.837
	Multi	**0.740**	**0.865**	**0.876**	**0.871**

Table 1: Micro-avg. precision, precision, recall, and F_1 for **TAC** and **Wiki** datasets. In a majority of languages, the **Multi** model outperforms the **Mono** model.

tation of the mention text m_s, *Oficina de la Presidencia*, by creating an mBERT representation of the entire sentence, selecting the lowest layer representations of each of the mention's sub-words,[3] and form a single representation using max pooling. We create a representation of the entity name e_s, *President of Mexico* in the same way, although there is no surrounding context as in a sentence.

For the mention context m_c we select the surrounding sentences up to BERT's 512 sub-word limit, positioning the mention in the middle, and pass the text to BERT, using the resulting top layer of the `[CLS]` token. We create a similar representation for the entity context e_c from the definition or other text in the KB, using the first 512 subword tokens from that description. For the mention type m_t and entity type e_t we create one-hot embeddings, omitting ones that do not occur more than 100 times in the training set.

3.2 Architecture

We feed the representations of the name (m_s and e_s), context (m_c, e_c) and type (m_t, e_t) into a neural ranker. Each of these three pairs is passed into distinct multilayer perceptrons (MLPs), which each produce an embedding that captures the similarity between each type of information. For example, we input m_s and e_s into a text-specific hidden layer, which produces a combined representation r_s. The same is done for the context and type representations, producing representations r_c and r_t, respectively. These three representations are then fed into a final MLP, which produces a final score ($[-1, 1]$.) The entire network is jointly trained with the ADAM optimizer and a ranking objective. We apply dropout at every layer, use ReLu as the intermediate activation function, and Tanh for the final layer. While additional features such as entity salience are likely useful for this task, we chose to restrict our model as much as possible to use only text features. This focuses on mBERT's multilingual ability, and allows for easier adaptation to new KBs than with KB-specific features.

3.3 Model Training

We learn the parameters θ of our scoring function S using a pairwise approach; this allows us to train our model without annotated scores. Our ranker scores a mention m and positive entity e_+ pair, and separately scores the same mention paired with n sampled negative entities e_-. We apply the hinge loss between our correct entity and the highest scoring negative entity,

$$L(\theta) = \mathbf{max}\{0, \epsilon - (S(\{m, e_+\}; \theta) - \mathbf{max}\{S(\{m, e_{0-}\}; \theta) \ldots S(\{m, c_{n-}\}; \theta)\}\}$$

We jointly train all components of the network, including the positive and negative portions of the network. The major benefit of this pairwise approach is that it does not rely on annotated scores, but instead uses negative sampling to train the ranker. We

[3] We experimented with several BERT layers and found this to be the best performing on the **TAC** development set.

tested random combinations of hidden layer sizes and dropout rates to find the best configuration (see Appendix A for parameter selection details).

4 Datasets

We conduct our evaluation on two cross-language entity linking datasets. We predict NILs by applying a threshold; mentions where all entities are below a given threshold are marked as NIL. We evaluate all models using the evaluation script provided by Ji et al. (2015), which reports Precision, Recall, F_1, and Micro-averaged precision. For implementation details, please see the appendix.

TAC. The 2015 TAC KBP Entity Linking dataset (Ji et al., 2015) consists of newswire and discussion form posts in English, Spanish, and Mandarin Chinese linked to an English KB. We use their evaluation set, and provide a comparison to the numbers noted in Ji et al. (2015). The referenced systems had access to non-English language KB text which we exclude, and thus are a goal rather than a baseline. Later papers, such as Upadhyay et al. (2018), also use this dataset but only for evaluation, instead training on Wikipedia and treating mentions that are linked to TAC entities without Wikipedia links as NIL. Therefore, we cannot compare our evaluation to this work.

Wiki. We created a cross-language entity linking dataset from Wikipedia links (Pan et al., 2017) that includes Korean, Farsi, Arabic, and Russian. A preprocessed version of Wikipedia has links in non-English Wikipedia pages to other non-English pages annotated with that link and an English page link if a corresponding page was available. From these annotations we created a dataset consisting of non-English mentions linked to English-language entities (Wikipedia page) using English Wikipedia as the KB. We consider this to be silver-standard data because–unlike the **TAC** dataset–the annotations have not been reviewed by annotators. Since we do not have a separate development set for this dataset, we apply the hyperparameters selected on **TAC** development data to this dataset.

Triage. We assume gold-standard mention boundaries in our analysis. We use the triage system of Upadhyay et al. (2018), which is largely based on work in Tsai and Roth (2016). This allows us to score a smaller set of entities for each mention as opposed to the entire KB. For a give mention m, a triage system will provide a set of k candidate entities $e_1 \dots e_k$. The system 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 ($k = 10$ for **TAC** and $k = 100$ for **Wiki**).

5 Model Evaluation

We consider several different training and evaluation settings to explore the multilingual ability of transformers on this task. Recent studies suggest that multilingual models can achieve similar or even better performance on cross-language entity linking (Upadhyay et al., 2018). Other work (Mueller et al., 2020) has shown that this is not always the case. Therefore, we begin by asking: does our linker do better when trained on all languages (multilingual cross-language) or trained separately on each individual language (monolingual cross-language)?

We train our model on each of the 7 individual languages in the two datasets (noted as **Mono**). Next, we train a single model for each dataset (3 languages in **TAC**, 4 in **Wiki**, each noted as **Multi**). **Mono** and **Multi** share the exact same architecture - there are no multilingual adjustments made, and the model contains no language-specific features. As **Multi** uses data available in all languages and thus has more training data than **Mono**, we include a model that is trained on a randomly-sampled subset of the multilingual training data that set to match the training size of **Mono** (**MultiDS**) . For **TAC Multi** models, we also report results using a candidate oracle instead of triage (**Multi+Or**), where the correct entity is always added to the candidate list. For all **Mono** and **Multi**-based models we report the average of three runs. The metric-specific standard deviations were all small, with all but one at or below 0.017. We note the best performing architecture from (Ji et al., 2015) as **Tri**, again noting that those systems have access to non-English text. We also evaluate a simple nearest neighbor model (noted as **NN**). This model scores each mention-entity pair using the cosine similarity between the mention name representation m_s and the entity representation e_s, and selects the highest-scoring pair.

Table 1 shows that for **TAC** there is a small difference between the **Mono** and **Multi** models. For **Wiki** the difference is often larger. **Multi** often does better than **Mono**, suggesting that additional training data is helpful specifically for languages (*e.g.,* Farsi) with smaller amounts of data. Over-

Training Setting	Evaluation Language: en	zh	es	ar	fa	ko	ru
Multi	0.66	0.79	0.74	0.72	0.86	0.88	0.87
en	.00	−.03	−.02	+.03	−.08	−.08	−.05
zh	−.05	.00	−.03	−.14	−.04	−.16	−.10
es	−.06	−.06	−.03	−.20	−.13	−.03	−.09
ar	−.06	−.06	−.03	−.20	−.08	−.13	−.03
fa							
ko							
ru							

Table 2: ΔF_1 for each single-language trained model, compared to a multilingually-trained model, for each evaluation language. Each column is an evaluated language, and each row is a training setting. While models trained on the target language perform best, many monolingually-trained models perform well on unseen languages.

	en		zh		es	
	avg	F_1	avg	F_1	avg	F_1
name	0.59	0.70	0.45	0.71	0.42	0.73
+cont	+.12	+.05	+.22	+.05	+.14	+.05
+type	+.03	+.01	+.10	−.02	+.03	−.03
all	+.12	+.05	+.26	+.08	+.19	+.06

Table 3: English-only trained Δmicro-average and ΔF_1 when using a subset of linker features, compared to the name-only model for each language in the Development set. The name component of the model has the highest performance impact, but context also leads to better performance in almost all cases.

BERT	Lang	micro	prec.	recall	F_1
en	en	−.07	+.17	−.13	−.03
en	es	−.01	.00	−.02	−.01
ar	ar	−.08	−.08	−.03	−.06
ar	fa	−.09	−.05	−.08	−.06

Table 4: Change in performance for monolingually-trained models using monolingually-trained BERT models, compared to monolingually-trained models using mBERT.

all, these results are encouraging as they suggest that a single trained model for our system can be used for cross-language linking for multiple languages. This can reduce the complexity associated with developing, deploying and maintaining multiple models in a multilingual environment. For some models, the **Multi** improvement may be due to additional data available, as shown in the difference in performance between **Multi** and **MultiDS** (*e.g.,* Spanish F_1 **Multi** is +.026 over **MultiDS**). However, the small difference in performance shows that even by providing additional out-of-language training data, reasonable performance can be achieved even with reduced in-language training.

6 Zero-shot Language Transfer

Encouraged by the results on multilingual training, we explore performance in a zero-shot setting. How does a model trained on a single language perform when applied to an unseen language? We consider all pairs of languages, *i.e.,* train on each language and evaluate on all others in the same dataset[4].

Table 2 shows the change in F_1 for monolingually-trained models compared to multilingual models. While zero-shot performance does worse than a model with access to within-language training data, the degradation is surprisingly small: often less than 0.1 F_1. For example, a model trained on all 3 **TAC** languages achieves an F_1 of 0.79 on Chinese, but if only trained on English, achieves an F_1 of 0.76. This pattern is consistent across both models trained on related languages (Arabic → Farsi, loss of 0.08 F_1), and on unrelated languages (Russian → Korean, loss of 0.13 F_1).

Analysis. Why does zero-shot language transfer do so well for cross-language entity linking? What challenges remain to eliminate the degradation in performance from zero-shot transfer?

We answer these questions by exploring the importance of each component of our cross-language ranking system: mention string, context, and type. We conduct ablation experiments investigating the performance loss from removing these information sources. We then evaluate each model in an English-trained zero-shot setting. First, we train a zero shot model using only the mention text and entity name. We then compare the performance change that results from adding the context, the type, and both context and type (all features).

Table 3 shows that comparing the name and men-

[4]Work in Cross-language entity linking (Upadhyay et al., 2018; Tsai and Roth, 2016) has done similar evaluations, but focus on using external large data sources (Wikipedia) to train their models.

tion text alone accounts for most of the model's performance, a sensible result given that most of the task involves matching entity names. We find that context accounts for most of the remaining performance, with type information having a marginal effect. This highlights the importance of the multilingual encoder, since both name and context rely on effective multilingual representations.

Separately, how does using a multilingual transformer model, such as mBERT, affect the performance of our ranker? First, it is possible that using a monolingual linker with a BERT model trained only on the target language would improve performance, since such a model does not need to represent several languages as the same time. As shown in Table 4, model performance for these settings is largely worse for English-only and Arabic-only (Safaya et al., 2020) models when compared to using mBERT, with the exception that precision increases significantly for English. Second, perhaps a monolingual linker with a BERT model trained only on a related language – *e.g.,* English BERT for Spanish, Arabic BERT for Farsi – would produce acceptable results. Again, as shown in Table 4, the performance is most often worse, illustrating that mBERT is an important aspect of the linker's performance.

7 Improving Zero-shot Transfer

7.1 Name Matching Objective

Given the importance of matching the mention string with the entity name, will improving this component enhance zero-shot transfer? While obtaining within-language entity linking data isn't possible in a zero-shot setting, we can use pairs of translated names, which are often more easily available (Irvine et al., 2010; Peng et al., 2015). Since Chinese performance suffers the most zero-shot performance reduction when compared to the multilingual setting, we use Chinese English name pair data (Huang, 2005) to support an auxiliary training objective. An example name pair: "巴尔的摩－俄亥俄铁路公司" and *Baltimore & Ohio Railroad*.

We augment model training as follows. For each update in a mini-batch, we first calculate the loss of the subset of the model that scores the mention string and entity name on a randomly selected pair $k = 25,000$ of the Chinese/English name pair corpus. We score the Chinese name z and the correctly matched English name e_+ pair, and separately score the same Chinese name paired with n negatively sampled English names e_-. We create representations for both z and e using the method described for names in §3.1 which are passed to the name-only hidden layer. We add a matching-specific hidden layer, which produces a score. We apply the hinge loss between positive and negative examples,

$$N(\theta) = \mathbf{max}\{0, \epsilon - (S(\{z, e_+\}; \theta) - \mathbf{max}\{S(\{z, e_{0-}\}; \theta) \dots S(\{z, e_{n-}\}; \theta)\}\}$$

The name pair loss is then multiplied by a scalar $\lambda = 0.5$ and added to the loss described in §3.3. The resulting loss $L_{joint}(\theta) = (\lambda * N(\theta)) + L(\theta)$ is jointly minimized. After training, we discard the layer used to produce a score for name matches. This procedure still only uses source language entity linking training data, but makes use of auxiliary resources to improve the name matching component, the most important aspect of the model.

We analyze the resulting performance by considering modifications to our English-only training setting, which are designed to replicate scenarios where there is little training data available. To show the effect of a smaller training corpus, we select a random 50% of mentions, partitioned by document (**Rand**). To show the importance of training on frequently occurring entities, we select 50% of mentions that are linked to the least frequent entities in the English dataset (**Tail**).

Table 5 shows the results on each of the three development **TAC** languages compared to the **Multi** model. For the **Rand** training set, we see a large improvement in Chinese micro-average and a small one in F_1, but otherwise see small reductions in performance. In the **Tail** training setting, a similar pattern occurs, with the exception that Chinese is less improved than in **Rand**. Overall, performance loss remains from zero-shot transfer which suggests that improvements need to be explored beyond just name matching.

7.2 Entities

Another possible source of zero-shot degradation is the lack of information on specific entities mentioned in the target language. For entity linking, knowledge of the distribution over the ontology can be very helpful in making linking decisions. While zero-shot models have access to general domain text, *i.e.,* news, they often lack text discussing the same entities. For example, some entities that only occur in Chinese (231 unique entities in **Dev**),

en		zh		es				en		zh		es	
avg	F_1	avg	F_1	avg	F_1			avg	F_1	avg	F_1	avg	F_1
0.64	0.75	0.51	0.69	0.53	0.75	Rand	Tail	0.53	0.66	0.45	0.66	0.42	0.70
.00	−.01	+.07	+.02	−.02	−.02	w/ Name		−.02	−.02	+.02	+.01	−.01	−.01
.00	+.01	+.06	+.04	+.01	+.02	w/ Pop-Train		−.02	+.04	.00	+.07	−.01	+.06
+.04	+.03	+.12	+.06	+.10	+.06	w/ Pop-All		+.13	+.10	+.20	+.11	+.22	+.10

Table 5: For each proposed Name matching or popularity re-ranking model, the change in performance (ΔF_1 and Δmicro-average) compared to the original **Rand** (left) and **Tail** (right) models. While the name matching increased performance somewhat, the additional of popularity was more impactful.

	en		zh		es	
	avg	F_1	avg	F_1	avg	F_1
Multi	0.70	0.73	0.77	0.81	0.68	0.82
Rand	−.04	-.02	−.26	−.12	−.15	−.07
N-1	+.01	+.02	−.04	−.02	−.08	−.03
N-1U	−.24	-.14	−.49	−.22	−.38	−.19
Tail	−.16	-.08	−.31	−.15	−.26	−.12

Table 6: For each of the English-only training data subsets described in §7.2, ΔMicro-average and ΔF_1 compared to the full **Multi** model. Models that see even a single example of an entity (*e.g.,* **N-1**) outperform models that see a portion (*e.g.,* **Tail**) or none (*e.g.,* **N-1U**).

such as the frequently occurring entity *Hong Kong*, have a number of similar entities and thus are more challenging to disambiguate.

We measure this effect through several diagnostic experiments where we evaluate on the development set for all languages, but train on a reduced amount of English training data in the following ways: In addition to the **Rand** and **Tail** settings, we sample a single example mention for each entity (**N-1**), resulting in a much smaller training as compared to those datasets. We also take **N-1** and remove all evaluation set entities (**N-1U**), leaving all evaluation entities unseen at train time.

Table 6 reports results on these reduced training sets. All languages use a -1 NIL threshold. Compared to the multilingual baseline (**Multi**) trained on all languages, there is a decrease in performance in all settings. Several patterns emerge. First, the models trained on a subset of the English training data containing more example entities - *e.g.,* **N-1** - have much higher performance than the models that do not. This is true even in non-English languages. Unobserved entities do poorly at test time, suggesting that observing entities in the training data is important.

However, a mention training example can improve the performance of a mention in another language if linked to the same entity, which suggests that this provides the model with data-specific entity information. Therefore, the remaining zero-shot performance degradation can be largely attributed not to a change in language, but to a change in topic, *i.e.,* what entities are commonly linked to in the data. This may also explain why although the name matching component is so important in zero-shot transfer, our auxiliary training objective was unable to fully mitigate the problem. The model may be overfitting to observed entities, forcing the name component to memorize specific names of popular entities seen in the training data. This means we are faced with a topic adaptation rather than a language adaptation problem.

We validate this hypothesis by experimenting with information about entity popularity. Will including information about which entities are popular improve zero-shot transfer? We answer this question by re-ranking the entity linker's top ten predicted entities using popularity information, selecting the most most popular entity from the list. Adding this feature into the model and re-training did not lead to a significant performance gain. We define the popularity of an entity to be the number of times it occurred in the training data. We report results for two popularity measures–one using the popularity of the English subset of the data used for training, and one using all of the training data (including for Spanish and Chinese).

Table 5 shows that both strategies improve F_1, meaning that a missing component of zero-shot transfer is information about which entities are favored in a specific dataset. The gain from using popularity estimated from the training data only is smaller than using the popularity data drawn from all of **TAC**. With more accurate popularity information, we can better mitigate loss.

Several patterns emerge from most common corrections made with the Population reranking for **Tail**, included in Table 8. Many errors arise from selecting related entities that are closely related to the correct entity – for example, *United States Congress* instead of the *United States of America*. Additionally, people with similar names are often confused (e.g. *Edmund Hillary* instead of *Hillary Clinton*). Finally, many appear to be annotation decisions – often both the original prediction (e.g. *Islamic State*) and the corrected popular prediction (e.g. *Islamic State of Iraq and Syria*) appear reasonable choices. While most corrections were in Chinese (632), some occurred in both English (419) and Spanish (187). These errors – especially those in English – illustrate that much of the remaining error is in failing to adapt to unseen entities.

8 Conclusion

We demonstrate that a basic neural ranking architecture for cross-language entity linking can leverage the power of multilingual transformer representations to perform well on cross-lingual entity linking. Further, this enables a multilingual entity linker to achieve good performance, eliminating the need for language-specific models. Additionally, we find that this model does surprisingly well at zero-shot language transfer. We find that the zero-shot transfer loss can be partly mitigated by an auxiliary training objective to improve the name matching components. However, we find that the remaining error is *not* due to language transfer, but to topic transfer. Future work that improves zero-shot transfer should focus on better ways to adapt to entity popularity in target datasets, instead of relying on further improvements in multilingual representations. Focusing on adapting to the topic and entities present in a given document is critical. This could be accomplished by adding a document-level representation or by leveraging other mentions in the document. English-focused work on rare entity performance (Orr et al., 2020; Jin et al., 2014) may provide additional direction.

References

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Mostafa Dehghani, Hamed Zamani, Aliaksei Severyn, Jaap Kamps, and W Bruce Croft. 2017. Neural ranking models with weak supervision. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 65–74. ACM.

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.

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.

Shudong Huang. 2005. *Chinese - English Name Entity Lists v 1.0 LDC2005T34*. Linguistic Data Consortium.

Ann Irvine, Chris Callison-Burch, and Alexandre Klementiev. 2010. Transliterating from all languages. In *Proceedings of the Ninth Conference of the Association for Machine Translation in the Americas*.

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

Yuzhe Jin, Emre Kıcıman, Kuansan Wang, and Ricky Loynd. 2014. Entity linking at the tail: Sparse signals, unknown entities, and phrase models. In *Proceedings of the 7th ACM International Conference on Web Search and Data Mining*, WSDM '14, page 453–462, New York, NY, USA. Association for Computing Machinery.

Dan Kondratyuk and Milan Straka. 2019. 75 languages, 1 model: Parsing universal dependencies universally. 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 2779–2795, Hong Kong, China. Association for Computational Linguistics.

Chi-kiu Lo and Michel Simard. 2019. Fully unsupervised crosslingual semantic textual similarity metric based on BERT for identifying parallel data. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 206–215, Hong Kong, China. Association for Computational Linguistics.

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.

David Mueller, Nicholas Andrews, and Mark Dredze. 2020. Sources of transfer in multilingual named entity recognition.

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.

Xiaoman Pan, Taylor Cassidy, Ulf Hermjakob, Heng Ji, and Kevin Knight. 2015. Unsupervised entity linking with abstract meaning representation. In *North American Chapter of the Association for Computational Linguistics*, pages 1130–1139.

Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng Ji. 2017. Cross-lingual name tagging and linking for 282 languages. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1946–1958.

Nanyun Peng, Mo Yu, and Mark Dredze. 2015. An empirical study of Chinese name matching and applications. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 377–383, Beijing, China. Association for Computational Linguistics.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy. Association for Computational Linguistics.

Jonathan Raphael Raiman and Olivier Michel Raiman. 2018. Deeptype: multilingual entity linking by neural type system evolution. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Shruti Rijhwani, Jiateng Xie, Graham Neubig, and Jaime Carbonell. 2019. Zero-shot neural transfer for cross-lingual entity linking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6924–6931.

Ali Safaya, Moutasem Abdullatif, and Deniz Yuret. 2020. KUISAIL at SemEval-2020 task 12: BERT-CNN for offensive speech identification in social media. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 2054–2059, Barcelona (online). International Committee for Computational Linguistics.

Samuel L Smith, David HP Turban, Steven Hamblin, and Nils Y Hammerla. 2017. Offline bilingual word vectors, orthogonal transformations and the inverted softmax. *arXiv preprint arXiv:1702.03859*.

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.

Henry Tsai, Jason Riesa, Melvin Johnson, Naveen Arivazhagan, Xin Li, and Amelia Archer. 2019. Small and practical BERT models for sequence labeling. 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 3632–3636, Hong Kong, China. 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.

Yuxuan Wang, Wanxiang Che, Jiang Guo, Yijia Liu, and Ting Liu. 2019. Cross-lingual BERT transformation for zero-shot dependency parsing. 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 5721–5727, Hong Kong, China. Association for Computational Linguistics.

Shijie Wu and Mark Dredze. 2019. Beto, bentz, becas: The surprising cross-lingual effectiveness of BERT. 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

833–844, Hong Kong, China. Association for Computational Linguistics.

A Architecture information

Parameter	Values
Context Layer(s)	[768], **[512]**, [256], [512,256]
Mention Layer(s)	[768], **[512]**, [256], [512,256]
Type Layer	[128], **[64]**, [32], [16]
Final Layer(s)	**[512,256]**, [256,128], [128,64], [1024,512], [512], [256]
Dropout probability	0.1, **0.2**, 0.5
Learning rate	1e-5, 5e-4, **1e-4**, 5e-3, 1e-3

Table 6: To select parameters for the ranker, we tried 10 random combinations of the above parameters, and selected the configuration that performed best on the TAC development set. The selected parameter is in bold. We report results after training for 500 epochs for TAC and 800 for Wiki. The full TAC multilingual model takes approximately 1 day to train on a single NVIDIA GeForce Titan RTX GPU, including candidate generation, representation caching, and prediction on the full evaluation dataset.

B Dataset Details

The NIL threshold is selected based on the developmentTAC dataset. Unless noted, we use -0.8 for English and -1 otherwise.

TAC: The training set consists of 30,834 mentions (6,857 NIL) across 447 documents. We reserved a randomly selected 20% of these documents as our development set, and will release development splits. The evaluation set consists of 32,459 mentions (8,756 NIL) across 502 documents. A mention is linked to NIL if there is no relevant entity in the KB, and the KB is derived from a version of BaseKB.

TAC Triage: We use the system discussed in for both the **TAC** and **Wiki** datasets. However, while the triage system provides candidates in the same KB as the **Wiki** data, not all entities in the **TAC** KB have Wikipedia page titles. Therefore, the **TAC** triage step requires an intermediate step - using the Wikipedia titles generated by triage ($k = 10$), we query a Lucene database of BaseKB for relevant entities. For each title, we query BaseKB proportional to the prior provided by the triage system, meaning that we retrieve more BaseKB entities for titles that have a higher triage score, resulting in $l = 200$ entities. First, entities with Wikipedia titles are queried, followed by the entity name itself. If none are found, we query the mention string - this provides a small increase in triage recall. This necessary intermediate step results in a lower recall rate for the **TAC** dataset (85.1% for the evaluation set) than the **Wiki** dataset, which was 96.3% for the evaluation set .

Wiki: Some BaseKB entities used in the **TAC** dataset have Wikipedia links provided; we used those links as seed entities for retrieving mentions, retrieving mentions in proportion to their presence in the **TAC** dataset, and to sample a roughly equivalent number of non-TAC entities. We mark 20% of the remaining mentions as NIL. In total, we train and evaluate on 5,923 and 1,859 Arabic, 3,927 and 1,033 Farsi, 5,978 and 1,694 Korean, and 5,337 and 1,337 Russian mentions, respectively.

Original Prediction	Popular Correction	Count
United States Department of State	United States of America	146
united_states_congress	United States of America	121
Soviet Union	Russian	57
Central Intelligence Agency	United States of America	41
healthcare_of_cuba	Cuba	36
islamic_state	Islamic State of Iraq and Syria	33
edmund_hillary	First lady Hillary Rodham Clinton	32
United States Department of Defense	United States of America	32
Tamerlan Tsarnaev	Dzhokhar A. Tsarnaev	27
Carl Pistorius	Oscar Leonard Carl Pistorius	23
CUBA_Defending_Socialism_ ... documentary	Cuba	22
Barack Obama Sr.	Barack Hussein Obama II	18
Iraq War	Iraq	14
Dzhokhar Dudayev	Dzhokhar A. Tsarnaev	13
Sumter County / Cuba town	Cuba	13
United States Army	United States of America	13
military_of_the_united_states	United States of America	13
Republic of Somaliland	Somalian	13
ISIS	Islamic State of Iraq and Syria	13
Islamic_State_of_Iraq_and_Syria	Islamic State of Iraq and Syria	12
National Assembly of People's Power	Cuba	11
Sara Netanyahu	Benjamin Netanyahu	10

Table 8: All pairs of original prediction and popular prediction altered by the reranking procedure described in Section 7.2, for the **Tail** model