# Zero-Shot Cross-lingual Name Retrieval for Low-Resource Languages

**Kevin Blissett***, **Heng Ji**†‡
* Computer Science Department, Rensselaer Polytechnic Institute
blissk@rpi.edu
† Department of Computer Science ‡ Department of Electrical and Computer Engineering
University of Illinois at Urbana-Champaign
hengji@illinois.edu

## Abstract

In this paper we address a challenging cross-lingual name retrieval task. Given an English named entity query, we aim to find all name mentions in documents in low-resource languages. We present a novel method which relies on zero annotation or resources from the target language. By leveraging freely available, cross-lingual resources and a small amount of training data from another language, we are able to perform name retrieval on a new language without any additional training data. Our method proceeds in a multi-step process: first, we pre-train a language-independent orthographic encoder using Wikipedia inter-lingual links from dozens of languages. Next, we gather user expectations about important entities in an English comparable document and compare those expected entities with actual spans of the target language text in order to perform name finding. Our method shows 11.6% absolute F-score improvement over state-of-the-art methods.

## 1 Introduction

Disasters happen all over the world, not just in the places where language experts are readily available. During these disasters, governments and aid organizations must be able to rapidly understand what is being said online and reported in the news. Extracting such information requires tools that can perform basic Natural Language Processing (NLP) tasks on all languages without language-specific annotations.

Finding names in documents is a critical part of extracting structured information from unstructured natural language documents. Therefore, it is an essential component for applications including Information Retrieval, Question Answering and Knowledge Base Population. Typical name finding methods rely on supervised learning and require training data from the target language. This makes name finding on languages that do not have annotated data available a useful and challenging problem.

We propose a novel approach for name finding that requires no training data from the language to be tagged. Our approach is based on the observation that the mentions of named entities often "look the same" across languages, even when those languages are not related. This "looks the same" relation is difficult to capture with traditional metrics such as edit distance and soundex. Nevertheless, when combined with user expectations about which entities will likely appear in a particular text, this relation provides enough information to identify named entities across the world's languages. To illustrate, let's consider the sentence, "Bill Gates and Paul Allen founded Microsoft in 1975.", as translated into Hindi and romanized by Google Translate[1]: "bil gets aur pol elan ne 1975 mein maikrosopht kee sthaapana kee." Even without any knowledge of Hindi, an English speaker told to identify the entities "Bill Gates", "Paul Allen", and "Microsoft" can easily match them to the spans "bil gets", "pol elan" and "maikrosopht" respectively by relying on this relation. By leveraging pre-training in a cross-lingual setting with freely available data from Wikipedia, we train a Convolutional Neural Network (CNN) model (Krizhevsky et al., 2012) that captures the orthographic similarity of names across languages. This model is trained to encode name mentions into fixed length vectors such that names which refer to the same entities across a large number of languages are close to one another in the encoding space. Because this cross-lingual encoder model is trained in a highly multilingual setting, it can serve as a metric to compare name

[1] https://translate.google.com/



*Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo)*, pages 275–280
Hong Kong, China, November 3, 2019. 
https://doi.org/10.18653/v1/P17

similarity across all of the world's languages, not just those in the training set. We encourage the model to learn more general similarity features across languages by using a large number of training samples and languages relative to the size of the model. After learning these general similarity features, the same encoder model can be applied to new languages without any additional training.

After learning a cross-lingual model of name similarity, we ask a user to provide query names in their native language. We can also extract such queries automatically when comparable corpora are available. Using our language-independent encoder model, these query names can then be compared to spans of text in any language. When those spans of text are similar to the queries provided by the user, we tag them as names. We train a Multilayer Perceptron (MLP) to perform this comparison step using annotations from a language for which we have ground truth name tagging information. Once this comparison model is trained, it can also be applied to find names in new languages without the need for any additional training data.

## 2 Approach

### 2.1 Training the Cross-lingual Encoder

The first component of our method is an encoder model that captures name similarity across languages. We first train this model and use it to generate representations of names as fixed length vectors. To train this model, we employ the method proposed by (Blissett and Ji, 2019) which is in turn adapted from (Schroff et al., 2015). In this approach, a neural network is used to encode names into vectors such that names referring to the same entity are close to one another in the vector space. A triplet loss is employed and the negative example in each training instance is sampled dynamically in order to provide consistently challenging and informative samples to the model.

Our encoder model is trained in a cross-linguistic setting using data from Wikipedia inter-lingual links. Wikipedia inter-lingual links are strings of text in various languages which all refer to a single entity's Wikipedia page. Clusters of these strings of text which refer to the same entity in various languages are easily recoverable using Wikipedia metadata. Our model is then trained to minimize the distance between the representations of names which refer to the same entity.

We make a change from the method employed by (Blissett and Ji, 2019) by using a convolutional neural network (CNN) for our encoder rather than a recurrent model. We use a CNN in this case rather than an RNN because we find that CNNs can be trained faster, require fewer parameters, and provide similar overall performance. We apply our encoder network to character embeddings trained jointly with the rest of the encoder. We then use max pooling to derive a fixed length vector from the encoder filter values.

### 2.2 Applying the Encoder to Name Finding

After the language independent encoder module is trained, we freeze the model and use it as a feature extractor for encoding strings of text both from a source language and from a target unknown language.

To perform name finding, the user is asked for a set of names (queries) the system will search for in the unknown language text. Because we can use our encoder module to derive representations of these queries that are comparable across languages, we can use these encoded queries in order to find their unknown language representation among the rest of the unknown language text.

Typically Recurrent Neural Networks (RNN) are used to perform name tagging. However, recurrent networks become sensitive to the word order of the language or languages that they are used to train them. This makes an RNN unsuitable for our task since we do not know the word order of our unknown target language. Instead, we enumerate the set $S$ of all spans of tokens of a sentence of length $l$

$$S = \{(i, j) \,|\, 0 < i < l, i \leq j < l\}$$

These substrings referred to by these spans are then encoded by our cross-lingual encoder and compared to the queries. Their similarities are computed using a simple Multi-layer Perceptron (MLP). We select an MLP since it is well suited to comparing pairs of vectors and requires relatively little training data. This MLP can be trained using labels from a language for which we have ground truth annotations. Since the encoding model providing input vectors to the MLP is language independent, the trained MLP can also be effectively applied to new, previously unseen languages as we show in our results in Table 1.

A problem arises when converting these similarity scores into a sequence of name tags. This

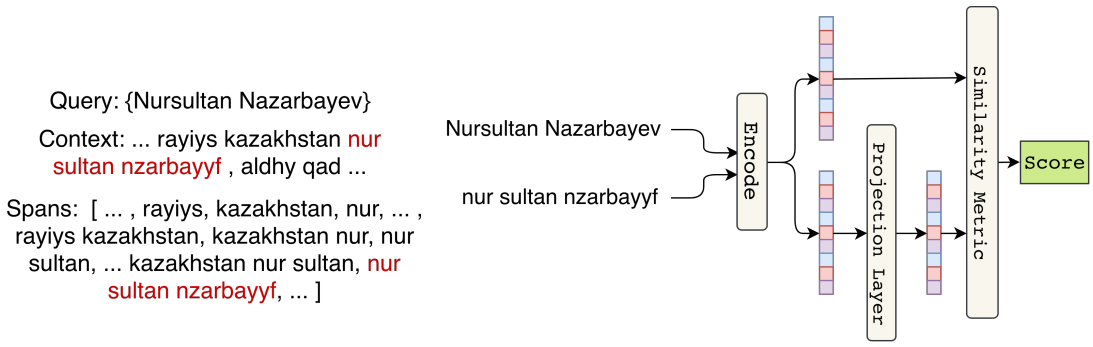


Figure 1: Overview figure for the approach. User provided queries are compared against contexts from the target language. Contexts are broken into individual spans of length 1, 2, etc. for comparison. Both queries and contexts are fed into the pre-trained cross-lingual name encoder and then their similarity is measured using a Multi-layer Perceptron.

| Train – Test | Precision | Recall | F-Score |
|---|---|---|---|
| Oromo – Oromo | 0.280 | 0.441 | 0.343 |
| Oromo – Tigrinya | 0.236 | 0.159 | 0.190 |
| Tigrinya – Tigrinya | 0.745 | 0.590 | 0.658 |
| Tigrinya – Oromo | 0.360 | 0.211 | 0.266 |

| | Precision | Recall | F-Score |
|---|---|---|---|
| Tigrinya | 0.429 | 0.002 | 0.004 |
| Oromo | 0.133 | 0.008 | 0.015 |

(a) string match baseline

| | Precision | Recall | F-Score |
|---|---|---|---|
| Tigrinya | 0.438 | 0.040 | 0.074 |
| Oromo | 0.190 | 0.232 | 0.209 |

(b) soundex-based baseline

Table 1: Performance statistics for our model (top) and baseline approaches (bottom).

problem is best illustrated with an example. Suppose our query is *nur sultan nzarbayyf* and our context sentence is *Kazakhstan's President Nursultan Nazarbayev has led the country since independence from the Soviet Union in 1991*. Our expectation is that the spans *Nursultan*, *Nazarbayev*, and *Nursultan Nazarbayev* will all have a high similarity to the query, but our model must select which of the spans is the best match with the query since each will lead to a different final sequence of tags (e.g. if we select the first span we will assign ['B', 'O'] to this subsequence of tags while selecting the last will lead to the subsequence ['B', 'I'], where 'B' indicates the beginning of a name, 'I' indicates inside a name and 'O' indicates outside of a name ).

When faced with a situation where multiple overlapping spans have a high similarity to the query (as calculated by our MLP) we need a tie-breaker which will tell us which of the spans we ought to ultimately select. We train our model to select the correct span automatically by linking this selection directly to our model's loss function during training.

For each token $t_n$ in the sentence, we assign a score representing the probability that $t_n$ should be assigned the tag 'B' and a score for the probability that the token should be tagged 'I'. To assign a score for the probability that $t_n$ should be assigned the tag 'B', we first collect a subset of spans $B_n$ from the set of all spans $S$ such that the first word in the span is $t_n$. That is,

$$B_n = \{s_i | s_i \in S \wedge s_i[0] = n\}$$

The score assigned for the probability that $t_n$ should be tagged 'B' is the highest score among the all the scores calculated by comparing the spans in $B_n$ with each query in $Q$. That is,

$$BScore_n = \max_{s_i \in B_n, q_i \in Q} f(tokens(s_i), q_i)$$

where $f$ represents our trainable similarity function and $tokens$ retrieves the tokens referred to by the span $s_i$.

Likewise, the score assigned for 'I' is the highest score among spans which include this token,

but in which it is not the first token. We turn these scores into probabilities using a sigmoid function and then compute the Binary Cross-Entropy Loss for the 'B' tags and the 'I' tags separately. For example,

$$l_B = -w_n(y_n \cdot \log BScore_n + (1 - y_n) \cdot \log(1 - BScore_n))$$

where $y_n$ is a label indicating if $t_n$ should be assigned the tag 'B' and $w_n$ is a weight such that

$$w_n = \begin{cases} 1 & \text{where } y_n = 0 \\ r \cdot \frac{\text{\# 'O' labels}}{\text{\# non-'O' labels}} & \text{where } y_n = 1 \end{cases}$$

where $r$ is a parameter of the model which can be selected to trade off between precision and recall. The number of non-'O' labels is either the number of 'B' or 'I' tags depending on which score we are currently computing. Typical values for $r$ in our models were 0.3 to 0.5. This weighting factor allows us to compensate for the fact that positive labels are rare in the data compared to negative labels.

These two losses are then averaged together to provide our final loss for this sentence.

$$l = \frac{(l_B + l_I)}{2}$$

## 3 Experiments

We use for our datasets an Oromo and Tigrinya news corpus from the DARPA LORELEI[2] program. Both are low-resource languages spoken primarily in Africa for which we have human annotated ground truth annotations for evaluation. Although the languages are both members of the Afro-Asiatic language family, they differ significantly in phonology, morphology, and vocabulary and are not mutually intelligible. We will use these languages as examples of unrelated languages in order to show that our model transfers well even without training data in languages closely related to the target language.

Our dataset includes annotations for the following types of entities: person, location, and geo-political entities. We exclude organizations since the names of organizations are commonly translated based on meaning rather than transliterated. We use the top 30 most common names in the dataset as queries to simulate a user who only knows about the most important entities involved in some event. The model is trained on one language using several hundred sentences from that language with the top 30 entities of that language's dataset as the queries. Since the CNN calculating cross-lingual encodings is pre-trained separately and frozen, model training at this point consists only of training our MLP to calculate span similarity scores. We then test by running the model using context sentences from a separate language and the top 30 entities from that language's dataset. For this experiment, the model is scored only on how many of the query entities identified in the context sentences, ignoring other entities. We only assign credit when the tag perfectly matches the correct spans including boundaries. We use simple "BIO" tags in which the first token of a name is tagged 'B', other tokens in the name are tagged 'I', and all other tokens are tagged 'O'. Our scores show that the model can transfer across languages.

We also compare our performance to two baselines. The first baseline tags names that are exact string matches with the query entities. The second applies the New York State Identification and Intelligence System (NYSIIS) phonetic code algorithm to both queries and target language text and then tags spans of target language text that match the queries. The NYSIIS approach performs significantly better than exact string matching, but our own method outperforms both. Results are summarized in Table 1.

## 4 Related Work

The problem of name tagging in low-resource languages has had real attention within the last few years. For example, (Zhang et al., 2016) use a variety of non-traditional linguistic resources in order to train a name tagger for use in low-resource languages. (Pan et al., 2017) and (Tsai et al., 2016) both rely on Wikipedia to provide data for training name tagging models for all Wikipedia languages. Much work has also been pursued for systems that rely on very limited silver-standard training data annotated from the target language by non-speakers (e.g., (Ji et al., 2017)). Our method differs from the above in that we do not require our target language to be present in Wikipedia or any other additional resources.

Cross-linguistic name tagging systems have also been pursued. For example, (Curran and

[2]LDC2017E57 and LDC2017E58 in the LDC Catalog

Clark, 2003) develop a feature-based model using a maximum entropy tagger to achieve good results in English, Dutch and German. Because we do not assume access to capitalization which does not exist in many languages, many of their most valuable features are not suitable for our setting. (Bharadwaj et al., 2016) demonstrates cross-lingual transfer for name tagging using phonologically grounded word representations. In particular, the authors demonstrate 0-shot transfer for their name tagging system between Uzbek and Turkish. While this approach requires monolingual word embeddings in the target language and benefits greatly from capitalization information, our method makes no such assumptions.

(Ji et al., 2008) used a phonetically based method to match English person names in Mandarin audio segments. This method uses an English-to-pinyin transliteration model and then applies fuzzy matching to the transliterated output. This is similar to our work in that it also exploits the phonetics underlying the spelling of names in order to produce matches, but differs in that we use the underlying learned representation directly rather than string matching.

Our approach differs primarily from all those outlined above in that we require no resources or information about the target unknown language. We also require no additional time for training our method in order to tag new languages.

## 5 Conclusions and Future Work

We propose a method to perform name tagging on an unknown languages using a pre-trained cross-lingual name encoder and user expectations about what names may appear in a given dataset. Our method requires no resources from the new language to be tagged. Future work may include performing graph-based query expansion on the target entities provided by the user. This could provide coverage of additional names not specifically searched for by the user.

## Acknowledgments

This research is based upon work supported in part by U.S. DARPA LORELEI Program HR0011-15-C-0115, the Office of the Director of National Intelligence (ODNI), and Intelligence Advanced Research Projects Activity (IARPA), via contract FA8650-17-C-9116. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of DARPA, ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.

## References

Akash Bharadwaj, David Mortensen, Chris Dyer, and Jaime Carbonell. 2016. Phonologically aware neural model for named entity recognition in low resource transfer settings. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1462–1472.

Kevin Blissett and Heng Ji. 2019. Cross-lingual NIL entity clustering for low-resource languages. In *Proceedings of the Second Workshop on Computational Models of Reference, Anaphora and Coreference*, pages 20–25.

James Curran and Stephen Clark. 2003. Language independent NER using a maximum entropy tagger. In *NAACL-HLT*.

Heng Ji, Ralph Grishman, and Wen Wang. 2008. Phonetic name matching for cross-lingual spoken sentence retrieval. In *2008 IEEE Spoken Language Technology Workshop*, pages 281–284. IEEE.

Heng Ji, Xiaoman Pan, Boliang Zhang, Joel Nothman, James Mayfield, Paul McNamee, Cash Costello, and Sydney Informatics Hub. 2017. Overview of TAC-KBP2017 13 languages entity discovery and linking. In *TAC*.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2012. Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems 25 (NIPS 2012)*.

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.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. FaceNet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823.

Chen-Tse Tsai, Stephen Mayhew, and Dan Roth. 2016. Cross-lingual named entity recognition via wikification. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 219–228.

Boliang Zhang, Xiaoman Pan, Tianlu Wang, Ashish Vaswani, Heng Ji, Kevin Knight, and Daniel Marcu. 2016. Name tagging for low-resource incident languages based on expectation-driven learning. In *NAACL-HLT*, pages 249–259.