Cross-lingual Learning-to-Rank with Shared Representations

Shota Sasaki[1], **Shuo Sun**[2], **Shigehiko Schamoni**[3], **Kevin Duh**[2], **Kentaro Inui**[1,4]
[1]Tohoku University, [2]Johns Hopkins University, [3]Heidelberg University, [4]RIKEN AIP
{sasaki.shota,inui}@ecei.tohoku.ac.jp, ssun32@jhu.edu,
schamoni@cl.uni-heidelberg.de, kevinduh@cs.jhu.edu

Abstract

Cross-lingual information retrieval (CLIR) is a document retrieval task where the documents are written in a language different from that of the user's query. This is a challenging problem for data-driven approaches due to the general lack of labeled training data. We introduce a large-scale dataset derived from Wikipedia to support CLIR research in 25 languages. Further, we present a simple yet effective neural learning-to-rank model that shares representations across languages and reduces the data requirement. This model can exploit training data in, for example, Japanese-English CLIR to improve the results of Swahili-English CLIR.

1 Introduction

Multilingual document collections are becoming prevalent. Thus an important application is cross-lingual information retrieval (CLIR), i.e. document retrieval which assumes that the language of the user's query does not match that of the documents. For example, imagine an investor who wishes to monitor consumer sentiment of an international brand in Twitter conversations around the world. She might issue a query string in English, and desire all relevant tweets in any language.

There are two main approaches to building CLIR systems. The *modular approach* involves a pipeline of two components: translation (machine translation or bilingual dictionary look-up) and monolingual information retrieval (IR). These approaches may be further divided into the *document translation* and *query translation* approaches (Nie, 2010). In the former, one translates all foreign-language documents to the language of the user query prior to IR indexing; in the latter, one indexes foreign-language documents and translates the query. In both, the idea is to solve the translation problem separately, so that CLIR becomes document retrieval in the monolingual setting.

A distinctly different way to build CLIR systems is what may be called the *direct modeling approach* (Bai et al., 2010; Sokolov et al., 2013). This assumes the availability of CLIR training examples of the form (q, d, r), where q is an English query, d is a foreign-language document, a r is the corresponding relevance judgment for d with respect to q. One directly builds a retrieval model $S(q, d)$ that scores the query-document pair. While q and d are in different languages, the model directly learns both translation and retrieval relevance on the CLIR training data. Compared to the modular approach, direct modeling is advantageous in that it focuses on learning translations that are beneficial for retrieval, rather than translations that preserve sentence meaning/structure in bitext.

However, there exist no large-scale CLIR dataset that can support direct modeling approaches in a wide variety of languages. To obtain relevance judgments, one typically needs a bilingual speaker who can read a foreign-language document and assess whether it is relevant for a given English query. This can be an expensive process. Here, we present a large-scale dataset that is automatically constructed from Wikipedia: it can support training and evaluation of CLIR systems between English queries and documents in 25 other languages (Section 2). The data is of sufficient size for direct modeling, and can also serve as an wide-coverage evaluation data for the modular approaches.[1]

To demonstrate the utility of the data, we further present experiments for CLIR in low-resource languages. First, we introduce a neural CLIR model based on the direct modeling approach (Section

[1]To facilitate CLIR research, the dataset is publicly available at http://www.cs.jhu.edu/~kevinduh/a/wikiclir2018/.

Proceedings of NAACL-HLT 2018, pages 458–463
New Orleans, Louisiana, June 1 - 6, 2018.

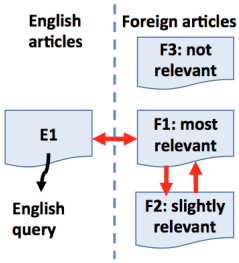

Figure 1: CLIR data construction process: From an English article (E1), we extract the English query. Using the inter-language link, we obtain the *most relevant* foreign-language document (F1). Any article that has mutual links to and from F1 are labeled as *slightly relevant* (F2). All other articles are *not relevant* (F3). The data is a set of tuples: (English query q, foreign document d, relevance judgment r), where $r \in \{0, 1, 2\}$ represents the three levels of relevance.

3.1). We then show how we can bootstrap CLIR models for languages with less training data by an appropriate use of paramater sharing among different language pairs (Section 3.2). For example, using the training data for Japanese-English CLIR, we can improve the Mean Average Precision (MAP) results of a Swahili-English CLIR system by 5-7 points (Section 4).

2 Large-Scale CLIR Dataset

We construct a large-scale CLIR data from Wikipedia. The idea is to exploit *inter-language links*: from an English page, we extract a sentence as query, and label the linked foreign-document pages as relevant. See Figure 1 for an illustration.

This data construction process is similar to (Schamoni et al., 2014) who made an English-German CLIR dataset, but ours is at a larger scale. Specifically, we use Wikipedia dumps released on August 23, 2017. English queries are obtained by extracting the first sentence of every English Wikipedia article. The intuition is that the first sentence is usually a well-defined summary of its corresponding article and should be thematically related for articles linked to it from another language. Similar to (Schamoni et al., 2014), title words from the query sentences are removed, because they may be present across different language editions. This deletion prevents the task from becoming an easy keyword matching task.

For practical purposes, each document is limited to the first 200 words of the article. Empty documents and category pages are filtered. Currently, our dataset consists of more than 2.8 million English queries and relevant documents from 25 other selected languages (see Table 1).

Language	#Doc	#Query	#SR
Arabic	535	324	194
Catalan	548	339	625
Chinese	951	463	462
Czech	386	233	720
Dutch	1908	687	1646
Finnish	418	273	665
French	1894	1089	4048
German	2091	938	4612
Italian	1347	808	2635
Japanese	1071	426	2912
Korean	394	224	343
Norwegian-Nynorsk	133	99	150
Norwegian-Bokmål	471	299	663
Polish	1234	693	1777
Portuguese	973	611	1130
Romanian	376	199	251
Russian	1413	664	1656
Simple English	127	114	135
Spanish	1302	781	2113
Swahili	37	22	35
Swedish	3785	639	1430
Tagalog	79	48	23
Turkish	295	185	195
Ukrainian	704	348	565
Vietnamese	1392	354	257
(All numbers are in units of one thousand)			

Table 1: CLIR dataset statistics. For each language X, we show the total number of documents in language X and the number of English queries. The number of "most relevant" documents is by definition equal to #Query. The number of "slightly relevant" documents is shown in the column #SR.

In sum, we have created a CLIR dataset that is large-scale in terms of both the amount of examples as well as the number of languages. This can be used in two scenarios: (1) one mixed-language collection where an English query may retrieve relevant documents in multiple languages. (2) 25 independent datasets for training and evaluating CLIR on English queries against one foreign language collection. In the experiments in Section 4, we will utilize the dataset in terms of scenario (2).[2]

[2]For extensibility purposes, these experiments use only half of the data, randomly sampled by query (the held-out data is reserved for other uses). Also it only considers binary relevance (*most relevant* vs *not relevant*) for simplicity. The exact data splits will be provided along with the data release.

3 Direct Modeling for CLIR

3.1 Neural Ranking Model

Given an English query q and a foreign-language document d, our models compute the relevance score $S(q, d)$. First, we represent each word as n-dimensional vectors, so q and d are represented as matrices $\mathbf{Q} \in \mathbb{R}^{n \times |q|}$ and $\mathbf{D} \in \mathbb{R}^{n \times |d|}$, where $|q|$ and $|d|$ are the numbers of tokens in q and d:

$$\begin{aligned} \mathbf{Q} &= [E_q(q_1); E_q(q_2); ...; E_q(q_{|q|})] \\ \mathbf{D} &= [E_d(d_1); E_d(d_2); ...; E_d(d_{|d|})] \end{aligned}$$

q_i and d_i denote the i-th term in q and d. E is embedding function which transforms each term to a dense n-dimensional vector as its representation. ; is the concatenation operator. Then, we apply convolutional feature map[3] to these matrices, followed by tanh activation and average-pooling to obtain each representation vector $\hat{q}$ and $\hat{d}$.

$$\hat{q} = CNN_q(\mathbf{Q}); \quad \hat{d} = CNN_d(\mathbf{D}) \tag{1}$$

Next, we define two variations in calculating $S(q, d)$. The first is a *cosine model* which computes cosine similarity between $\hat{q}$ and $\hat{d}$:

$$S_{cos}(q, d) = cossim(\hat{q}, \hat{d}) \tag{2}$$

The second is a *deep model* with a fully connected layer on top of the concatenation of $\hat{q}$ and $\hat{d}$ (a 200-dimensional vector):

$$\begin{aligned} S_{deep}(q, d) &= tanh(O \cdot h_{vec}^{\mathrm{T}}) \\ &= tanh(O \cdot relu(W \cdot [\hat{q}; \hat{d}]^{\mathrm{T}})) \end{aligned} \tag{3}$$

Here, $O \in \mathbb{R}^{1 \times h}$ and $W \in \mathbb{R}^{h \times 200}$ are the deep model parameters, and h is the number of dimensions of the hidden state, $h_{vec} \in \mathbb{R}^{1 \times h}$. For regularization, we set dropout rate as 0.5 (Srivastava et al., 2014) at the hidden layer.

In the training phase, we minimize pairwise ranking loss, which is widely used for learning-to-rank (Pang et al., 2016; Guo et al., 2016; Hui et al., 2017; Xiong et al., 2017; Dehghani et al., 2017), defined as follows:

$$L = max\left\{0, 1 - (S(q, d^+) - S(q, d^-))\right\} \tag{4}$$

where d^+ and d^- are relevant and non-relevant document respectively. We fix only the word embeddings and tune the other parameters.

[3]The $n \times 4$ convolution window has filter size of 100 and takes a stride of 1.

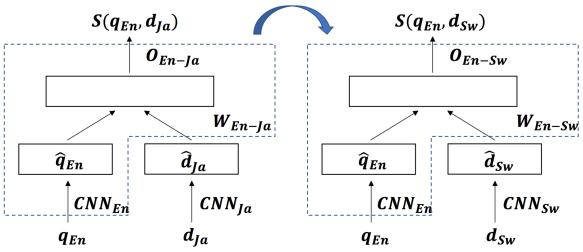

Figure 2: Illustration of the proposed method. On low resource dataset (e.g. Swahili-English), the parameters of the CNN for encoding query (CNN_{En}) and the parameters of the fully connected layer (O_{En-Sw}, W_{En-Sw}) are initialized by the ones pre-trained on high resource dataset (e.g. Japanese-English).

	Ja	De	Fr
$S_{cos}(q, d)$: cos	59/74	49/66	55/70
$S_{deep}(q, d)$: h=100	61/75	64/77	69/81
$S_{deep}(q, d)$: h=200	68/80	67/79	74/84
$S_{deep}(q, d)$: h=300	70/82	70/81	74/84
$S_{deep}(q, d)$: h=400	**73**/83	**71/82**	75/**85**
$S_{deep}(q, d)$: h=500	**73/84**	70/81	**76/85**

Table 2: P@1/MAP performance (0-100 range, in percent) of the cosine model and the deep model with different hidden state size on **high resource datasets**. Best value in each column is highlighted in bold.

We note there are many other ranking models that can be adapted to CLIR (Huang et al., 2013; Shen et al., 2014; Xiong et al., 2017; Mitra et al., 2017); they have a common framework in extracting features from both query and document and optimizing scores $S(q, d)$ via some ranking loss.

3.2 Sharing Representations

Training a network like the deep model generally requires a nontrivial amount of data. To address the data requirement for low-resource languages, we propose a simple yet effective method that shares representations across CLIR models trained in different language-pairs. Basically, we use the same architecture as the deep model ($S_{deep}(q, d)$, Equation 3). However, we use the parameters trained on a high-resource dataset (e.g Japanese-English) to initialize the parameters for a low-resource language-pair (e.g. Swahili-English).

Figure 2 illustrates the idea: Concretely, we initialize the parameters of the CNN for encoding query (CNN_q) and the parameters of the fully connected layer (O, W) by using the pre-trained parameters. When training on low-resource data,

	Tl			Sw			De (subsample)			Fr (subsample)		
	In	Sh	Δ	In	Sh	Δ	In	Sh	Δ	In	Sh	Δ
cos	51/68	50/67	-/-	51/67	49/65	-/-	40/59	38/56	-/-	46/63	43/60	-/-
h=100	34/50	48/62	+/+	46/62	46/62	=/=	39/55	46/62	+/+	40/57	46/62	+/+
h=200	44/58	55/67	+/+	47/63	52/67	+/+	41/57	48/63	+/+	43/60	51/66	+/+
h=300	42/57	49/63	+/+	50/65	58/70	+/+	44/60	50/65	+/+	49/65	51/66	+/+
h=400	49/63	**57/69**	+/+	51/66	**60/73**	+/+	45/61	**51/66**	+/+	47/64	**56/70**	+/+
h=500	51/64	54/67	+/+	53/68	56/69	+/+	44/60	49/65	+/+	47/63	51/66	+/+

Table 3: P@1/MAP performances on **low resource datasets**. Δ columns show the comparison between the basic deep models with in-language training (In) and the deep models with sharing parameters (Sh); + indicates Sh outperforms In, and - indicates the In outperforms Sh. Best value in each dataset is highlighted in bold.

we fix only the word embedding, and tune the parameters of CNNs and the fully connected layer.

The intuition behind this is that our direct modeling approach enforces $\hat{q}$ and $\hat{d}$ to become language-independent representations of the query and document. The parameters O and W in the deep layer can therefore be used for any language-pair. Note for the cosine model, we can also share parameters for CNN_q.

4 Experiment Results

Setup: We use datasets of 3 high-resource languages (Japanese [Ja], German [De], French [Fr]) and 2 low-resource languages (Tagalog [Tl], Swahili [Sw]). We also subsample German and French data to be equivalent to the size of Swahili, in order to compare training size effects. Word embedding with dimension 100 for each language is trained on Wikipedia corpus, using word2vec SGNS (Mikolov et al., 2013). The size of hidden states in the deep model is {100, 200, 300, 400, 500}. We adopt Adam (Kingma and Ba, 2014) for optimization, train for 20 epochs and pick the best epoch based on development set loss. For the proposed method of parameter sharing, we use the weight parameters pre-trained on Japanese-English dataset to initialize parameters.

High-resource results: Table 2 shows the P@1 (precision at top position) and MAP (mean average precision) for datasets consisting of on the order of 100k+ training queries. The deep models outperformed the cosine models under all conditions, suggesting that the fully connected layer can exploit the large training set in learning more expressive scoring functions.

Low-resource results: Table 3 shows the results on the low resource datasets under two conditions: training on only the language-pair of interest (in-language), or additionally sharing parameters using a pre-trained Japanese-English model. For the in-language case, we observe the cosine model outperforms the deep model. In contrast to the high-resource results, this implies that deep models, which have a lot of parameters, only become effective if provided with sufficient training data.

For the sharing case, the deep models with parameter sharing outperformed the basic deep models trained only on in-language data under almost all conditions. This indicates that our sharing method reduces training data requirement. Importantly, by sharing parameters, the deep models are now able to outperform the cosine model and achieve the best results on all datasets.[4]

5 Conclusion and Future Work

We introduce a large-scale CLIR dataset in 25 languages. This enables the training and evaluation of direct modeling approaches in CLIR. We also present a neural ranking model with shared representations, and demonstrate its effectiveness in bootstrapping CLIR in low-resource languages.

Future work includes: (a) expansion of the dataset to more languages, (b) extraction of different types of queries and relevant judgments from Wikipedia, and (c) development of other ranking models. Importantly, we also plan to evaluate our models on standard CLIR test sets such as TREC (Schäuble and Sheridan, 1997), NTCIR (2007), FIRE (2013) and CLEF (2016). This will help answer the question of whether knowledge

[4]Sharing representations with the cosine models did not help; we hypothesize that cross-lingual sharing only works if given sufficient model expressiveness. We also tried the shared deep models on high resource datasets (e.g. using Japanese parameters on the full French dataset without subsampling). As expected, results did not change significantly.

learned from automatically-generated datasets can be transferred to a wide range of CLIR problems.

Acknowledgments

This work was supported by Cooperative Laboratory Study Program (COLABS-Outbound), JSPS KAKENHI Grant Number 15H0170 and JST CREST Grant Number JPMJCR1513, Japan. Sun and Duh are supported in part by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via contract # FA8650-17-C-9115. 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 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

2007. NII test collection for IR systems project. http://research.nii.ac.jp/ntcir/ntcir-ws6/ws-en.html.

2013. Forum for information retrieval evaluation. https://www.isical.ac.in/~fire/2013/index.html.

2016. Conference and labs of the evaluation forum. http://clef2016.clef-initiative.eu/.

Bing Bai, Jason Weston, David Grangier, Ronan Collobert, Kunihiko Sadamasa, Yanjun Qi, Olivier Chapelle, and Kilian Weinberger. 2010. Learning to rank with (a lot of) word features. *Information Retrieval* 13(3):291–314.

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*. ACM, pages 65–74.

Jiafeng Guo, Yixing Fan, Qingyao Ai, and W. Bruce Croft. 2016. A deep relevance matching model for ad-hoc retrieval. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*. ACM, pages 55–64.

Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*. ACM, pages 2333–2338.

Kai Hui, Andrew Yates, Klaus Berberich, and Gerard de Melo. 2017. PACRR: A position-aware neural ir model for relevance matching. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, pages 1049–1058.

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* .

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*. Curran Associates Inc., pages 3111–3119.

Bhaskar Mitra, Fernando Diaz, and Nick Craswell. 2017. Learning to match using local and distributed representations of text for web search. In *Proceedings of the 26th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, pages 1291–1299.

Jian-Yun Nie. 2010. *Cross-Language Information Retrieval*. Morgan & Claypool Publishers.

Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, and Xueqi Cheng. 2016. A study of match pyramid models on ad-hoc retrieval. In *Neu-IR16 SIGIR Workshop on Neural Information Retrieval*.

Shigehiko Schamoni, Felix Hieber, Artem Sokolov, and Stefan Riezler. 2014. Learning translational and knowledge-based similarities from relevance rankings for cross-language retrieval. In *Proceedings of the 52 Annual Meeting of the Association for Computational Linguistics*.

P. Schäuble and P. Sheridan. 1997. Cross-language information retrieval (CLIR) track overview. In *Proceedings of TREC Conference*.

Yelong Shen, Xiaodong He, Jianfeng Gao, Li Deng, and Grégoire Mesnil. 2014. A latent semantic model with convolutional-pooling structure for information retrieval. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*. ACM, pages 101–110.

Artem Sokolov, Laura Jehl, Felix Hieber, and Stefan Riezler. 2013. Boosting cross-language retrieval by learning bilingual phrase associations from relevance rankings. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, pages 1688–1699.

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(1):1929–1958.

Chenyan Xiong, Zhuyun Dai, Jamie Callan, Zhiyuan Liu, and Russell Power. 2017. End-to-end neural ad-hoc ranking with kernel pooling. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, pages 55–64.