Learning Neural Representation for CLIR with Adversarial Framework

Bo Li, Ping Cheng
School of Computer Science
Central China Normal University
Wuhan, China
libo@mail.ccnu.edu.cn

Abstract

The existing studies in cross-language information retrieval (CLIR) mostly rely on general text representation models (e.g., vector space model or latent semantic analysis). These models are not optimized for the target retrieval task. In this paper, we follow the success of neural representation in natural language processing (NLP) and develop a novel text representation model based on adversarial learning, which seeks a task-specific embedding space for CLIR. Adversarial learning is implemented as an interplay between the generator process and the discriminator process. In order to adapt adversarial learning to CLIR, we design three constraints to direct representation learning, which are (1) a matching constraint capturing essential characteristics of cross-language ranking, (2) a translation constraint bridging language gaps, and (3) an adversarial constraint forcing both language and source invariant to be reached more efficiently and effectively. Through the joint exploitation of these constraints in an adversarial manner, the underlying cross-language semantics relevant to retrieval tasks are better preserved in the embedding space. Standard CLIR experiments show that our model significantly outperforms state-of-the-art continuous space models and approaches the strong machine translation and monolingual baselines.

1 Introduction

Text representation is a crucial problem in most natural language processing (NLP) and information retrieval (IR) tasks. In monolingual IR, early research works mostly use vector space models for query-document semantic matching (Salton et al., 1975), which suffer from the problem of *synonymy* and *polysemy*. In order to bridge the lexical gaps, latent semantic models such as latent semantic analysis (LSA) (Deerwester et al., 1990) have been proposed to abstract away from surface text forms to approximate semantics. More recently, text representation learned with neural networks is attracting increasing attention of the IR community (Mitra and Craswell, 2017) and positive results have been reported on various evaluation data sets (Fan et al., 2018).

Compared to the prosperity in monolingual IR, there have been less advancements in CLIR where documents are written in a language different from that of queries. In addition to document ranking, CLIR models need to cross the language barriers, which makes the task intuitively more difficult than monolingual IR. Traditional approaches reduce CLIR to its monolingual counterpart via performing some way of translation on queries or/and documents. The typical translation process is performed with either off-the-shelf machine translation (MT) systems or multilingual dictionaries (Nie, 2010). However, MT based approaches are far from being a suitable solution for solving CLIR problems (refer to detailed analysis in (Zhou et al., 2012)). Dictionary-based approaches suffer from the same problem of lexical gaps as in the monolingual case (Gupta et al., 2017). An efficient cross-language representation is in need for CLIR, which is expected to be able to cross both the language and lexical gaps.

The most intuitive idea one can have so as to represent text in cross-language settings is to extend those models in monolingual environment. For instance, we note studies such as the extension of LSA in (Littman et al., 1998), the extension of principle component analysis (PCA) in (Platt et al., 2010), the extension of autoencoder model in (Chandar et al., 2014), and the extension of word2vec (Mikolov et al., 2013) in (Vulić and Moens, 2015). These approaches construct cross-language and semantic-rich representation of text, which can be applied to CLIR directly. However, all the models listed here aim to learn general text

Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 1861–1870
Brussels, Belgium, October 31 - November 4, 2018.

representation where the objective is to capture term proximity rather than relevance that is essential for retrieval task (Zamani and Croft, 2017). A recent work (Gupta et al., 2017) tries to learn task-specific representation for CLIR. However, their model only captures ranking signals in monolingual settings, which does not necessarily generalize well in CLIR.

In this paper, we propose to learn task-specific text representation for CLIR via a novel adversarial learning framework. Following the convention in generative adversarial networks (GAN) (Goodfellow et al., 2014), our representation learning model is realized as an interplay between two processes, an embedding generator (G) and an adversarial discriminator (D), conducted as a min-max game. With the GAN framework, we design three constraints to direct the representation learning process. CLIR is essentially a ranking problem and we develop a matching constraint to make sure that documents can be ranked in the right order given a query in another language. The matching constraint considers both cross-language and monolingual pairwise ranking signals, which is superior to previous studies (e.g., (Gupta et al., 2017)) only considering monolingual matching signals. Meanwhile, a translation constraint is imposed on the latent representation to bridge the language gaps. These two constraints direct the encoding networks to generate a language-invariant and task-specific representation in the embedding space. Lastly, an adversarial constraint is proposed to force both language and source invariant to be reached more efficiently and effectively. Through the joint exploitation of these constraints in an adversarial manner, the embedding space being optimal for CLIR will then result through the convergence of this process. Comprehensive CLIR experiments reveal that our model is superior to state-of-the-art continuous space models and approaches the machine translation and monolingual baselines.

2 Related work

Text representation has been a long-standing research question in IR. Classic methods such as vector space model are not able to deal with lexical gaps between queries and documents, resulting in inferior retrieval performance. Latent semantic approaches such as LSA (Deerwester et al., 1990) and latent dirichlet allocation (LDA) (Blei et al., 2003) abstract away from surface text forms to alleviate sparsity and approximate semantics. More recently, learning based approaches with neural networks have gained great success in NLP (Baroni et al., 2014) and started to attract increasing interests of the IR community. In terms of word level embedding, word2vec (Mikolov et al., 2013) and Glove (Pennington et al., 2014) are two models that have been cited frequently in recent literature. These two models provide semantic-rich representations to bridge lexical gaps between queries and documents, which have been used broadly in neural IR studies (Ganguly et al., 2015; Zheng and Callan, 2015; Zamani and Croft, 2016).

The above studies deal with monolingual text representation, which are related to the cross-language models presented below. As for CLIR, typical approaches reduce CLIR to its monolingual counterparts via performing some way of translation. Machine translation systems such as Google translator[1] have been widely used to translate queries or documents, which serve as a default and convenient translation option in CLIR. It is however far from being a suitable solution for solving CLIR problems (a detailed analysis can be found in (Zhou et al., 2012)). An alternative solution is to rely on multilingual dictionaries to perform lexicon-level translation, which is mostly in combination with either language modeling strategy (Kraaij et al., 2003) or query structuring framework (Pirkola, 1998). However, dictionary-based methods still suffer from the lexical gap problem which reduces their performance in CLIR.

In fact, researchers have extended the models in monolingual settings and developed continuous space models for cross-language tasks to capture rich semantics. These cross-language extensions can be applied to CLIR directly. For instance, Littman et al. (1998) extend LSA to its cross-language version CL-LSA by concatenating document-term matrix of parallel data which acts as dual-language documents to be learned by LSA. Such a methodology leads to a dual-language semantic space in which terms from both languages are represented. Vinokourov et al. (2002) use parallel data to find most likely correlations between projected vectors based on canonical component analysis technique. The OPCA model (Platt et al., 2010) starts with the basic model

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

PCA that is then made discriminative by encouraging comparable document pairs to have similar vector representation. Compared to CL-LSA, OPCA avoids the use of artificial concatenated documents. More recently, neural models have been employed to learn cross-language representations. For instance, autoencoder is extended to a bilingual version BAE in (Chandar et al., 2014) which learns vectorial word representations from aligned sentences. Yih et al. (2011) develop S2Net to learn a projection matrix to map the corresponding term vectors into a latent space where similar documents are close. S2Net is implemented with Siamese neural network framework. Vulić and Moens (2015) first merge two documents from the aligned document pair in a comparable corpus and then train word2vec on the pseudo-bilingual document to obtain cross-language embeddings. The above approaches learn general text representation that captures term proximity rather than relevance which is important for retrieval task (Zamani and Croft, 2017). A recent work (Gupta et al., 2017) tries to learn task-specific embeddings for CLIR. However, it learns ranking signals by preserving pairwise ranking in monolingual settings prior to a transfer learning process to another language, which does not necessarily generalize well in CLIR.

One can find from above analysis that, most existing approaches, either based on neural networks or not, learn general embeddings irrelevant to CLIR. We argue that task-specific embeddings are superior, a fact that is inspired by monolingual IR studies and that will actually be validated by CLIR experiments in this paper. To this end, we will learn cross-language and task-specific embeddings for CLIR via a novel text representation model based on adversarial learning (Goodfellow et al., 2014).

3 Representation learning framework

We will present in this section a neural representation learning framework for CLIR. As discussed before, the framework is realized based on adversarial learning as an interplay between the generator process and the discriminator process. We will develop three constraints, namely a matching constraint, a translation constraint and an adversarial constraint, to direct the learning of cross-language and target-specific text embeddings. For ease of presentation, let us assume in CLIR we have a source language query q_s and a target language document d_t. The translation of q_s in the target language is q_t. The learning framework is illustrated in figure 1, which consists of an adversarial network NN_{adv}, three dimension adaptation networks NN_{dim} and three encoding networks respectively for q_t, d_t and q_s.

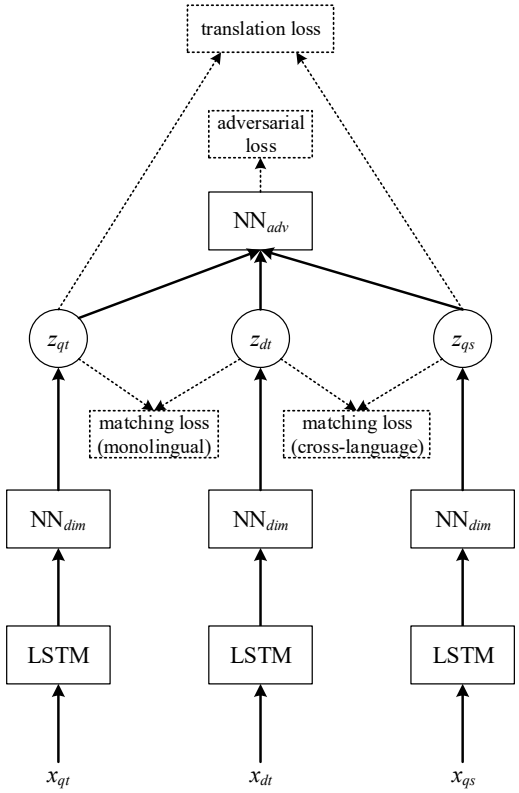

Figure 1: Text representation learning model for CLIR with adversarial framework.

3.1 Text representation networks

There have been various approaches one can use to encode sentences/documents into dense vectors. For instance, models based on convolutional neural networks (Kalchbrenner et al., 2014) and models based on recurrent neural networks (Liu et al., 2016) have been popular choices.

In order to map queries and documents into the embedding space, we make use of recurrent neural network with the long short-term memory (LSTM) architecture that can deal with vanishing and exploring gradient problems (Hochreiter and Schmidhuber, 1997). We present here derivation details of LSTM for clarification sake. The LSTM framework consists of several gates to control the cell state in the network. Firstly, a forget gate f (a

sigmoid layer) functions according to:

$$f^\tau = \sigma(W_f \cdot [h^{\tau-1}, x^\tau] + b_f)$$

Then, an input gate i (a sigmoid layer) and a tanh layer work together as follows:

$$i^\tau = \sigma(W_i \cdot [h^{\tau-1}, x^\tau] + b_i)$$
$$\widetilde{C}^\tau = tanh(W_c \cdot [h^{\tau-1}, x^\tau] + b_c)$$

With the forget gate f, the input gate i and the new value $\widetilde{C}$, one can update the cell state C as:

$$C^\tau = f^\tau * C^{\tau-1} + i^\tau * \widetilde{C}^\tau$$

Lastly, an output gate o (a sigmoid layer) outputs:

$$o^\tau = \sigma(W_o \cdot [h^{\tau-1}, x^\tau] + b_o)$$
$$h^\tau = o^\tau * tanh(C^\tau)$$

In above equations, x^τ is the input at time step τ. h^τ and $h^{\tau-1}$ denote the hidden states at time steps τ and $\tau - 1$. All W and b are parameters. For brevity, we can write the update process as:

$$h^\tau = LSTM(h^{\tau-1}, x^\tau)$$

Given a text sequence $x = (x^1, x^2, \ldots, x^l)$, typical methods take the output h^l of LSTM at the last time step l as the concentrated representation of the whole sequence x (Sutskever et al., 2014). Since queries in IR tasks tend to be short and noisy, we make use of Bidirectional LSTM with pooling (Tan et al., 2015) to obtain a more effective text representation from all the hidden states $h^{1:l}$. The sequence x is fed from left to right into $LSTM_a$ and from right to left into $LSTM_b$. The new hidden state h^τ_{ab} at time step τ is obtained by concatenating the hidden states of $LSTM_a$ and $LSTM_b$ at their respective time step τ. Since max-pooling has been proven to be efficient in similar tasks (Tan et al., 2015), the latent representation z_x of x can be formulated as:

$$z_x = NN_{dim}(MaxPooling(h^{1:l}_{ab}))$$

where x can be q_s, q_t or d_t. NN_{dim} is designed to adapt the output dimension and to allow further flexibility for representation learning.

3.2 Matching constraint and Translation constraint

Document ranking is the central problem in both monolingual IR and CLIR tasks. CLIR differs itself from its monolingual counterpart in that the language gap needs to be crossed prior to the retrieval process. Since the choice of translation strategies (query, document or both) affects the design of other components in our model, we will discuss the translation constraint in section 3.2.1 prior to matching constraints in sections 3.2.2 and 3.2.3.

3.2.1 Translation constraint

The translation constraint is developed to minimize the differences between a pair of parallel texts, which serves as a basic requirement in the translation scenario. Such a constraint directs the learning of language-invariant text representation for CLIR. We follow the arguments in previous studies (Vilares et al., 2016) and choose to translate queries in our model, since it is computationally expensive to translate large-scale document collections in practice. In this paper, we directly employ Google translator to translate queries, which is a popular choice for machine translation that leads to state-of-the-art translation performance. The translation constraint is then imposed on the embedding vectors z_{q_s} and z_{q_t} of the queries q_s and q_t. The translation loss L_{tra} on a set QP of query pairs can be defined with the squared L2 norm, which is:

$$L_{tra} = \sum_{(q_s, q_t) \in QP} \|z_{q_s} - z_{q_t}\|_2^2$$

3.2.2 Cross-language matching constraint

The matching constraint captures essential characteristics of cross-language ranking. Following the practice in learning to rank (Liu, 2009), we model document ranking in the *pairwise* style where the relevance information is in the form of preferences between pairs of documents with respect to individual queries. In the model for CLIR, since we have matching signals from both monolingual text pairs and cross-language text pairs, the model can benefit from complementary knowledge from two resources. The monolingual pairwise matching constraint will be introduced in section 3.2.3.

Similar to neural models in monolingual settings (Huang et al., 2013), the cross-language pairwise matching constraint is placed on top of the embedding vectors of source language query and target language documents. In figure 1, let us assume x_{q_s} has a relevant document $x_{d_{t+}}$ and an irrelevant document $x_{d_{t-}}$ according to annotated text pairs. In training, the positive sample $x_{d_{t+}}$

for x_{q_s} can be chosen as the most relevant texts according to annotation, and the negative sample $x_{d_{t-}}$ is picked randomly from the data collection. The cross-language matching constraint encourages the hidden representation of $x_{d_{t+}}$ to be near to the hidden representation of x_{q_s} in the semantic-rich embedding space. Meanwhile, it asks the hidden representation of $x_{d_{t-}}$ to be far from that of x_{q_s}. We follow typical neural IR models and make use of cosine as the distance measure of hidden vectors. The probability that d_{t+} is ranked higher than d_{t-} given q_s can be derived as:

$$\hat{P}(q_s) = \sigma[\beta_c \cdot (\cos(z_{q_s}, z_{d_{t+}}) - \cos(z_{q_s}, z_{d_{t-}}))]$$

where σ is the sigmoid function with a hyper-parameter β_c controlling its shape. The cross-language matching loss L_{matc} on cross-language triplet set QD_c can be defined with cross-entropy loss as:

$$L_{matc} = \sum_{(q_s, d_{t+}, d_{t-}) \in QD_c} CE[P(q_s), \hat{P}(q_s)]$$

where CE denotes the cross-entropy operator between two distributions and $P(q_s)$ is the actual counterpart of $\hat{P}(q_s)$ estimated from annotation with a strategy similar to that in (Dehghani et al., 2017).

3.2.3 Monolingual matching constraint

The monolingual matching constraint L_{matm} can be built in a way similar to that of L_{matc}. L_{matm} is imposed on a set QD_m of monolingual triplet (q_t, d_{t+}, d_{t-}) as:

$$L_{matm} = \sum_{(q_t, d_{t+}, d_{t-}) \in QD_m} CE[P(q_t), \tilde{P}(q_t)]$$

where $P(q_t)$ is the actual counterpart of $\tilde{P}(q_t)$ estimated from annotation. $\tilde{P}(q_t)$ denotes the probability that d_{t+} is ranked higher than d_{t-} given q_t. It can be computed with the sigmoid function as:

$$\tilde{P}(q_t) = \sigma[\beta_m \cdot (\cos(z_{q_t}, z_{d_{t+}}) - \cos(z_{q_t}, z_{d_{t-}}))]$$

where β_m is a hyper-parameter.

3.2.4 Embedding generator constraint

Since our model is implemented with adversarial framework, we propose to model the representation generator G, which embodies the process of language-invariant and task-specific embedding of queries and documents into a latent subspace, under a combination of three constraints introduced above. The translation constraint aims to guarantee language invariant when translating queries. The cross-language matching constraint explicitly captures cross-language ranking signals from cross-language text pairs. The monolingual matching constraint takes monolingual ranking into account so as to complement the cross-language ranking signals.

Combing the three constraints above, we obtain a comprehensive constraint that should be obeyed by the embedding generator process. With the regularization term L_{reg} equaling to the sum of Frobenius norms of all weight matrices in the text embedding phase, we can write the embedding generator constraint L_G as:

$$L_G(\theta_G) = \gamma_1 \cdot L_{tra} + \gamma_2 \cdot L_{matc} + \gamma_3 \cdot L_{matm} + L_{reg}$$

where θ_G denotes the set of parameters in the generator networks, and γ_1, γ_2, γ_3 are hyper-parameters.

3.3 Adversarial constraint

We will introduce the adversarial constraint in this part. GAN (Goodfellow et al., 2014) simultaneously trains a generative model G and a disriminative model D in a competing way. G generates samples from a source of noise w that satisfies $w \sim P_n(w)$ and tries to capture the real data distribution P_r. D learns to distinguish between the generated samples from G and the true data sampled from P_r (in practice, from training data). The training procedure for G is to try its best to fool D. Let us assume that G generates samples satisfying the distribution P_g that is implicitly decided by $G(w)$. The GAN value function $V(G, D)$ on which D and G play the minmax game can be written as:

$$\min_G \max_D V(D, G) = E_{x \sim P_r}[\log D(x)] + E_{x \sim P_g}[\log(1 - D(x))] \quad (1)$$

Theoretical analysis has indicated that playing the minmax game as above amounts to minimizing the Jensen-Shannon divergence between P_g and P_r.

We follow the general idea of GAN and develop an adversarial component on top of the embedding space in figure 1. We note that GAN has been used in representation learning in a similar way as in (Bousmalis et al., 2016; Liu et al., 2017). In our model in figure 1, the adversarial component NN_{adv} acts as the discriminator D

which tries its best to detect whether the embedding vector z is encoded from x_{q_t}, x_{d_t} or x_{q_s}. In this paper, NN_{adv} is implemented as a neural network with a softmax output layer. The output of NN_{adv} then corresponds to a probability distribution vector over the input sources. Let us denote the ground truth label of the current input z to NN_{adv} as l_z which indicates the source that z is encoded from. We can adjust equation 1 to our settings and obtain the adversarial loss L_{adv} on a query set Q_t and a document set D_t in the target language, as well as a query set Q_s in the source language, which can be written as:

$$L_{adv} = \min_G \max_D \sum_{x \in Q_t, D_t, Q_s} \log NN_{adv}(z_x) \circ l_{z_x}$$

where $\circ$ is the inner product operator.

3.4 Training procedure

Following the training convention of GAN (Goodfellow et al., 2014), the process of learning the language-invariant and task-specific text representation for CLIR should be conducted by jointly minimizing the generator constraint L_G and the adversarial loss L_{adv}, which leads us to the combined objective function L as:

$$L = L_G + L_{adv}$$

According to the rule of playing the minmax game in GAN, G tries its best to maximize the probability that D makes a mistake and D tries its best to distinguish between real data and generated data (in our case, various input sources). The theoretical requirement behind GAN that D is maintained near its optimal solution as long as G changes slowly enough motivates us to update the discriminator part k steps per update of the generator part in the iterative optimization process. Based on these discussions, the minmax optimization process can be derived as:

1. Optimize D when fixing G through:
   $\hat{\theta}_D = \arg\max_{\theta_D} L(\hat{\theta}_G, \theta_D)$

2. Optimize G when fixing D through:
   $\hat{\theta}_G = \arg\min_{\theta_G} L(\theta_G, \hat{\theta}_D)$

The optimization can be implemented with mini-batch gradient ascent (for θ_D) and descent (for θ_G).

4 Experiments and results

In this section, we conduct CLIR experiments so as to compare our text representation model with several other models.

4.1 Data sets

4.1.1 CLIR evaluation sets

To perform CLIR experiments, we rely on broadly used data sets released in the bilingual tasks of the cross-language evaluation forum (CLEF) [2]. We choose to use the data from the year 2000 to 2004. Table 1 lists the characteristics of the data set, which include number of documents (N_d), number of distinct words (N_w), the average document length (DL_{avg}) and the number of queries (N_q) in each task. We use source language queries in French (Fr), German (De) and Italian (It) to retrieve target language documents in English (En). Queries from year 2000 to 2002 are combined to a single task in table 1 since they have the same target set.

Table 1: CLIR dataset statistics (k = thousand).

Dataset	N_d	N_w	DL_{avg}	N_q
CLEF00-02	113k	173k	311	140
CLEF03	169k	233k	284	60
CLEF04	56k	120k	231	50

4.1.2 Training set

In order to train the representation learning model, we need to construct a data set consisting of annotated text pairs. We combine AOL queries (Pass et al., 2006) and a set of news titles downloaded from the news sites[3] to constitute training query set of diversity. Following the previous work (Gupta et al., 2017), we sample a balanced subset (1M) from such query set and use these queries to retrieve the data collection with BM25. For each training query, we take the top retrieved texts as positive samples, and the negative samples are selected randomly from the data collection. In addition to the pseudo-labeled text pairs of low quality, we combine the LETOR4.0 dataset (Qin and Liu, 2013) that is developed for evaluating learning to rank models. The LETOR4.0 dataset consists of relevance judgments of higher quality compared to

[2] http://www.clef-initiative.eu

[3] We fetch 2.8M web pages from several news websites such as ChinaDaily (www.chinadaily.com.cn) and XinhuaNews (www.xinhuanet.com).

pseudo-labeled data. The two data resources can complement each other in the training process.

In our experiments, the pseudo-labeled data is used to train the whole model and the LETOR dataset is employed to fine tune the parameters relevant to the source queries and target documents which are more important for the cross-language retrieval task.

4.2 Experimental settings

4.2.1 Experimental setup

The terms are initialized as the 512d word2vec vectors trained on Wikipedia dump corpus[4]. The term embeddings are fed into the LSTM model of which the hidden unit number is chosen from $\{64, 128, 256, 512\}$. The adversarial network NN_{adv} is as a three-layer feed-forward network with softmax on top of the last layer. NN_{dim} is implemented as a feed-forward network with layer dimension chosen from $\{32, 64, 128, 256\}$ and hidden layer number chosen from $\{1, 2\}$. The values of hyper-parameters γ_1, γ_2 and γ_3 are chosen from $\{0.01, 0.1, 1, 10, 100\}$. The learning rate is selected from $\{10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}\}$. Those hyper-parameters are tuned on the validation set which is 20% of the training queries randomly selected.

For evaluation, we present results in terms of mean average precision (MAP). Statistically significant differences between various models are determined using the paired t-test with $p < 0.05$.

4.2.2 Baseline approaches

We make use of three categories of baselines for CLIR experiments.

1. Monolingual run (MON): a baseline with target language queries that are strictly parallel to source language queries.

2. Machine translation (MT): a baseline with target-language queries translated by machine translation system from source-language queries.

3. Cross-language text representation models: baselines that rely on continuous space models for cross-language text representation. We make use here of S2Net (Yih et al., 2011), BAE (Chandar et al., 2014), and XCNN (Gupta et al., 2017) for the CLIR task.

[4]https://dumps.wikimedia.org

4.3 Results and analysis

4.3.1 Comparisons to state-of-the-art

Table 2 lists the experimental results on CLEF dataset for our model (the column OURS) and all baseline models. There are three data collections and three language pairs, amounting to nine cross-language retrieval tasks. Except the strong baselines MON and MT, our model shows the best overall performance among all CLIR strategies. Indeed, our model outperforms all continuous space baselines (i.e., S2Net, BAE and XCNN) with statistical significance in almost all cases. Our model decreases slightly from the strong MT baseline in most retrieval tasks with only one degradation being significant on 03(De-En). Furthermore, one can find that our model approaches the monolingual baseline very much in all retrieval tasks with all MAP ratios around or over 90%. In our experiments, we have not performed comparisons to CL-LSA (Littman et al., 1998) and its variant OPCA (Platt et al., 2010), because they have been consistently outperformed by other CLIR strategies with a large margin (Schauble and Sheridan, 1997; Nie, 2010; Vulić et al., 2011).

Among all continuous space baselines, the most recent model XCNN shows the best performance. XCNN always outperforms linear projection methods S2Net with significance. It also significantly outperforms the non-linear model BAE in all cases. This is coincident with previous conclusions in (Gupta et al., 2017) due to the fact that XCNN learns target-specific representation for CLIR but the other models do not. Our model also tries to learn task-specific representation for CLIR, which significantly outperforms XCNN in most cases according to the results in table 2. The reasons might be that (1) our method is modeled in a more effective adversarial learning framework. (2) we explicitly capture cross-language ranking signals in embedding generator in addition to monolingual ranking signals used in XCNN. (3) our model can jointly capture the translation knowledge and document ranking knowledge in a unified framework.

4.3.2 Variant of our model

Our model can be customized easily by altering the constraints to direct the representation learning process. Since the specificity of our model comes from the adversarial learning framework that has never been investigated in CLIR, we re-

Table 2: Retrieval performance (MAP scores) of all models on CLEF collections. $+(m/x)$ or $-(m/x)$ indicates that the improvements or degradations with respect to MT/XCNN are statistically significant. The highest value in each row (except the MON and MT baselines) is marked in bold. The percentages in the last column denote the MAP ratio of our model with respect to the MON baseline.

Data	Lang	MON	MT	S2Net	BAE	XCNN	OURS	PROP
00-02	Fr-En	0.469	0.431	0.330^{-m}_{-x}	0.369^{-m}_{-x}	0.401^{-m}	$\mathbf{0.424}_{+x}$	90.4%
	De-En	0.469	0.447	0.341^{-m}_{-x}	0.381^{-m}_{-x}	0.420^{-m}	$\mathbf{0.435}_{+x}$	92.8%
	It-En	0.469	0.439	0.339^{-m}_{-x}	0.374^{-m}_{-x}	0.409^{-m}	$\mathbf{0.426}_{+x}$	90.8%
03	Fr-En	0.498	0.471	0.352^{-m}_{-x}	0.383^{-m}_{-x}	0.431^{-m}	$\mathbf{0.456}_{+x}$	91.6%
	De-En	0.498	0.462	0.358^{-m}_{-x}	0.390^{-m}_{-x}	0.430^{-m}	$\mathbf{0.439}^{-m}$	88.2%
	It-En	0.498	0.468	0.367^{-m}_{-x}	0.395^{-m}_{-x}	0.439^{-m}	$\mathbf{0.467}_{+x}$	93.8%
04	Fr-En	0.517	0.483	0.378^{-m}_{-x}	0.402^{-m}_{-x}	0.442^{-m}	$\mathbf{0.470}_{+x}$	90.9%
	De-En	0.517	0.482	0.382^{-m}_{-x}	0.419^{-m}_{-x}	0.447^{-m}	$\mathbf{0.473}_{+x}$	91.5%
	It-En	0.517	0.477	0.385^{-m}_{-x}	0.411^{-m}_{-x}	0.458	$\mathbf{0.481}_{+x}$	93.0%

move the constraint L_{adv} from the original model M and obtain the variant M_{adv}. In this case, M_{adv} can be optimized with standard mini-batch gradient descent approach, without playing the minmax game. We redo above CLIR experiments with the same settings as above and obtain the retrieval results of M_{adv} in table 3.

Table 3: Retrieval performance (MAP scores) of the variant M_{adv} on CLEF collections. $+$ or $-$ indicates that the improvements or degradations with respect to our original model M are statistically significant. The higher value in each row is marked in bold.

Data	Lang	M	M_{adv}
00-02	Fr-En	**0.424**	0.412^{-}
	De-En	**0.435**	0.418^{-}
	It-En	**0.426**	0.424
03	Fr-En	**0.456**	0.440^{-}
	De-En	**0.439**	0.435
	It-En	**0.467**	0.448^{-}
04	Fr-En	**0.470**	0.453^{-}
	De-En	**0.473**	0.465
	It-En	**0.481**	0.469

From the results one can find that when removing the adversarial component from the original model, M_{adv} decreases from the original model M in all retrieval tasks. The differences that are significant appear in 5 out of 9 retrieval tasks. The results demonstrate that learning generator and discriminator in a competing style within the adversarial learning framework leads to representation of higher quality, which eventually supports efficient CLIR. If we compare the variant M_{adv} with the XCNN model in table 2, we find that M_{adv} still performs better than XCNN in most cases. Such a comparison implicitly indicates that the joint exploitation of monolingual matching constraint, cross-language matching constraint and translation constraint in a single model is more efficient than using them separately as in the XCNN model.

5 Conclusions

In this paper, we propose a novel text representation approach for CLIR based on the adversarial learning framework. The learning framework is implemented as an interplay between an embedding generator process and an adversarial discriminator process, which leads to an optimal representation that is both language invariant and domain specific. The embedding generator is learned such that it explicitly considers both cross-language and monolingual pairwise ranking signals. In this way, it can ensure that the learned embeddings benefit from both sources and are directly optimized for CLIR. To the best of our knowledge, it is the first time adversarial learning has been applied to CLIR. Experiments on various language pairs in CLEF data collection show that our model is significantly better than other latent semantic models for CLIR. Indeed, our model approaches the performance of machine translation and monolingual baselines.

Acknowledgments

We thank the anonymous reviewers for their valuable comments. This work was supported by the Fundamental Research Funds for Central Universities of CCNU (No. CCNU15A05062).

References

Marco Baroni, Georgiana Dinu, and Germán Kruszewski. 2014. Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, ACL, pages 238–247.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

Konstantinos Bousmalis, George Trigeorgis, Nathan Silberman, Dilip Krishnan, and Dumitru Erhan. 2016. Domain separation networks. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS, pages 343–351.

A P Sarath Chandar, Stanislas Lauly, Hugo Larochelle, Mitesh M Khapra, Balaraman Ravindran, Vikas Raykar, and Amrita Saha. 2014. An autoencoder approach to learning bilingual word representations. In *Proceedings of the 27th International Conference on Neural Information Processing Systems*, NIPS, pages 1853–1861.

Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407.

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*, SIGIR, pages 65–74.

Yixing Fan, Jiafeng Guo, Yanyan Lan, Jun Xu, Chengxiang Zhai, and Xueqi Cheng. 2018. Modeling diverse relevance patterns in ad-hoc retrieval. In *Proceedings of the 41st International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR, pages 375–384.

Debasis Ganguly, Dwaipayan Roy, Mandar Mitra, and Gareth J.F. Jones. 2015. Word embedding based generalized language model for information retrieval. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR, pages 795–798.

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Proceedings of the 27th International Conference on Neural Information Processing Systems*, NIPS, pages 2672–2680.

Parth Gupta, Rafael E. Banchs, and Paolo Rosso. 2017. Continuous space models for clir. *Information Processing & Management*, 53(2):359 – 370.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

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 Information and Knowledge Management*, CIKM, pages 2333–2338.

Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. 2014. A convolutional neural network for modelling sentences. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, ACL, pages 655–665.

Wessel Kraaij, Jian-Yun Nie, and Michel Simard. 2003. Embedding web-based statistical translation models in cross-language information retrieval. *Computational Linguistics*, 29(3):381–419.

Michael L. Littman, Susan T. Dumais, and Thomas K. Landauer. 1998. Automatic cross-language information retrieval using latent semantic indexing. In *Cross-Language Information Retrieval*, pages 51–62, Boston, MA. Springer.

Pengfei Liu, Xipeng Qiu, Jifan Chen, and Xuanjing Huang. 2016. Deep fusion lstms for text semantic matching. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, ACL, pages 1034–1043.

Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. 2017. Adversarial multi-task learning for text classification. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, ACL, pages 1–10.

Tie-Yan Liu. 2009. Learning to rank for information retrieval. *Foundations and Trends in Information Retrieval*, 3(3):225–331.

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*, NIPS, pages 3111–3119.

Bhaskar Mitra and Nick Craswell. 2017. Neural models for information retrieval. *CoRR*, abs/1705.01509.

Jian-Yun Nie. 2010. Cross-language information retrieval. *Synthesis Lectures on Human Language Technologies*, 3(1):1–125.

Greg Pass, Abdur Chowdhury, and Cayley Torgeson. 2006. A picture of search. In *Proceedings of the 1st International Conference on Scalable Information Systems*, InfoScale.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, EMNLP, pages 1532–1543.

Ari Pirkola. 1998. The effects of query structure and dictionary setups in dictionary-based cross-language information retrieval. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR, pages 55–63.

John C. Platt, Kristina Toutanova, and Wen-tau Yih. 2010. Translingual document representations from discriminative projections. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP, pages 251–261.

Tao Qin and Tie-Yan Liu. 2013. Introducing LETOR 4.0 datasets. *CoRR*, abs/1306.2597.

G. Salton, A. Wong, and C. S. Yang. 1975. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620.

Peter Schauble and Paraic Sheridan. 1997. Cross-language information retrieval (clir) track overview. In *Proceedings of the sixth Text REtrieval Conference*, TREC.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems*, NIPS, pages 3104–3112.

Ming Tan, Bing Xiang, and Bowen Zhou. 2015. Lstm-based deep learning models for non-factoid answer selection. *CoRR*, abs/1511.04108.

Jesus Vilares, Manuel Vilares, Miguel A. Alonso, and Michael P. Oakes. 2016. On the feasibility of character n-grams pseudo-translation for cross-language information retrieval tasks. *Computer Speech and Language*, 36(C):136–164.

Alexei Vinokourov, John Shawe-Taylor, and Nello Cristianini. 2002. Inferring a semantic representation of text via cross-language correlation analysis. In *Proceedings of the 15th International Conference on Neural Information Processing Systems*, NIPS, pages 1497–1504.

Ivan Vulić, Wim De Smet, and Marie-Francine Moens. 2011. Cross-language information retrieval with latent topic models trained on a comparable corpus. In *Proceedings of the 7th Asia Conference on Information Retrieval Technology*, AIRS, pages 37–48.

Ivan Vulić and Marie-Francine Moens. 2015. Monolingual and cross-lingual information retrieval models based on (bilingual) word embeddings. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR, pages 363–372.

Wen-tau Yih, Kristina Toutanova, John C. Platt, and Christopher Meek. 2011. Learning discriminative projections for text similarity measures. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning*, CoNLL, pages 247–256.

Hamed Zamani and W. Bruce Croft. 2016. Embedding-based query language models. In *Proceedings of the 2016 ACM International Conference on the Theory of Information Retrieval*, ICTIR, pages 147–156.

Hamed Zamani and W. Bruce Croft. 2017. Relevance-based word embedding. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR, pages 505–514.

Guoqing Zheng and Jamie Callan. 2015. Learning to reweight terms with distributed representations. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR, pages 575–584.

Dong Zhou, Mark Truran, Tim Brailsford, Vincent Wade, and Helen Ashman. 2012. Translation techniques in cross-language information retrieval. *ACM Computing Surveys*, 45(1):1:1–1:44.