Training Effective Neural CLIR by Bridging the Translation Gap

Hamed Bonab, Sheikh Muhammad Sarwar, and James Allan
Center for Intelligent Information Retrieval
College of Information and Computer Sciences
University of Massachusetts Amherst
{bonab,smsarwar,allan}@cs.umass.edu

ABSTRACT

We introduce *Smart Shuffling*, a cross-lingual embedding (CLE) method that draws from statistical word alignment approaches to leverage dictionaries, producing dense representations that are significantly more effective for cross-language information retrieval (CLIR) than prior CLE methods. This work is motivated by the observation that although neural approaches are successful for monolingual IR, they are less effective in the cross-lingual setting. We hypothesize that neural CLIR fails because typical cross-lingual embeddings "translate" query terms into *related* terms – i.e., terms that appear in a similar context – in addition to or sometimes rather than synonyms in the target language. Adding related terms to a query (i.e., query expansion) can be valuable for retrieval, but must be mitigated by also focusing on the starting query. We find that prior neural CLIR models are unable to bridge the *translation gap*, apparently producing queries that drift from the intent of the source query.

We conduct extrinsic evaluations of a range of CLE methods using CLIR performance, compare them to neural and statistical machine translation systems trained on the same translation data, and show a significant gap in effectiveness. Our experiments on standard CLIR collections across four languages indicate that Smart Shuffling fills the translation gap and provides significantly improved semantic matching quality. Having such a representation allows us to exploit deep neural (re-)ranking methods for the CLIR task, leading to substantial improvement with up to 21% gain in MAP, approaching human translation performance. Evaluations on bilingual lexicon induction show a comparable improvement.

ACM Reference Format:
Hamed Bonab, Sheikh Muhammad Sarwar, and James Allan. 2020. Training Effective Neural CLIR by Bridging the Translation Gap. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '20), July 25–30, 2020, Virtual Event, China.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3397271.3401035

1 INTRODUCTION

Cross-lingual information retrieval (CLIR) is the task of retrieving documents in a language different from that of the query. The recent success of neural network approaches for monolingual IR [33, 37, 42] suggests that neural techniques could readily be applied in the CLIR context, too. However, although there are a very few studies of neural approaches to CLIR, none of them targets or evaluates the standard CLIR task such as CLEF collections [7]. This study explores the complex cross-lingual issues to start developing an understanding of challenges that someone designing a neural CLIR system will need to address.

Existing CLIR systems are usually implemented as a two-step process: (1) query translation followed by (2) monolingual IR (search in the target language) [39]. Although an eventual end-to-end neural CLIR system is likely to intertwine the translation and relevance (ranking) steps of CLIR, we keep them separated for this exploration. That is, we use existing state-of-the-art neural and non-neural approaches for retrieval and focus our attention on the impact of different translation approaches because of the fundamental importance of translation to those approaches [34, 58].

Most existing work for (monolingual) neural IR relies on word2vec or GloVe for text representation [14, 24, 37, 54]. More recently, the use of pre-trained language models based on transformer neural networks (e.g., BERT) for ad-hoc retrieval in English resulted in high gain in the performance [33, 40, 53]. Building on that idea, Vulić and Moens [51] proposed a *cross-lingual* word embedding (CLE) construction method by randomly shuffling the parallel translation corpora and using word2vec [35] on the outcome. This process resulted in a single dense vector representation of words in both languages, where the vector for a word in one language is close to the vector for its translation. They used that CLE for query translation and also used semantic ranking based on cosine similarity. Litschko et al. [31] used off-the-shelf pre-trained CLEs to combine query translation and semantic space rankings and provided a fully unsupervised framework.

Ruder et al. [45] surveys several other recent studies on pre-trained cross-lingual representation learning, varying by learning paradigm and translation data availability, and usually incorporated in some downstream task other than CLIR. Some of these even extend the contextualized language models for jointly learning to encode many (over 100) languages into a shared CLE space (e.g., mBERT [52] and XLM [27]). We used a subset of these neural text representation approaches as translation resources to perform CLIR query translation. Surprisingly, in our initial experiments we found that these approaches fail to match the performance of a statistical machine translation (SMT) system for query translation.

We hypothesize that this failure is mostly due to the poor translation quality of CLEs. An ideal CLE for CLIR should encode the meaning of a word in such a way that it is first close to its synonym and then related words in a semantic space. However, we observe that existing CLE methods often translate query terms into related

SIGIR '20, July 25–30, 2020, Virtual Event, China
© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-8016-4/20/07…$15.00
https://doi.org/10.1145/3397271.3401035

terms – i.e., terms that appear in a similar context – instead of translating them into synonyms in the target language. Adding related terms to a query (i.e., query expansion) can be valuable for retrieval, but must be mitigated by simultaneously focusing on the starting query. For example, the most similar English translation of a French query term "automobile" is "industry" in the distributed representation trained using one of our baseline CLE methods. However, a word alignment based machine translation method [41], trained on the same data, correctly provides "car" as the English translation term.

This tendency to find related rather than synonymous translations creates a performance failure that we call the *translation gap*. It prevents a neural CLIR system from being as effective as existing state-of-the-art non-neural CLIR systems. This is because training deep neural (re-)ranking models with noisy and limited amounts of data becomes more challenging when dealing with non-English languages, particularly with resource-lean languages [3, 32]. For example, in a bilingual scenario, the vocabulary size is almost doubled and there is scarce publicly available training data (e.g., bilingual query log or click data). We present an effective CLE construction method, *Smart Shuffling*, which shows that effort to bridge the translation gap results in substantial improvements in effectiveness. We break down our study into the following concrete research questions:

RQ1. How does search query translation performance vary across SMT, NMT and existing CLE approaches when all these models are trained using same parallel data?

RQ2. How does Smart Shuffling CLE compare to existing CLE baselines as a query translator? Is there any qualitative and quantitative translation gap among the query translations provided by Smart Shuffling CLE, MT, and a state-of-the-art CLE approach?

RQ3. If we can achieve an equivalent translation quality with CLE compared to MT methods – i.e., filling the translation gap – can we use it for neural (re-)ranking methods and get improved retrieval performance?

RQ4. Do our improvements generalize to other tasks such as bilingual lexicon induction (BLI) [19], which is a prominent CLE evaluation task?

In the rest of this study, we investigate query translation systems, and present an extrinsic evaluation of state-of-the-art CLE techniques for CLIR. We compare them to an effective statistical machine translation system, trained on the same translation data, and show a significant translation gap. Our experiments on CLEF, a standard CLIR collections across four languages indicate that Smart Shuffling CLE fills the translation gap for query translation and provides significantly improved semantic matching quality. Having such a representation, we exploit a deep neural re-ranking method and show substantial improvement with up to 21% gain in MAP, approaching human translation performance. Our BLI evaluations also show significant improvements with up to 17% gain in MAP.

2 BACKGROUND AND RELATED WORK

We briefly survey the existing translation models as well as neural CLIR models. Note that for translation models one could translate source into target language (or vice versa), or translate both into a shared representation. Here with machine translation (MT) models, we only consider the query translation given its low-cost and flexibility [34, 46], and leave other variations (e.g., document translation) as future work.

2.1 Query Translation Models

Several studies show that the translation task for CLIR is different than that of MT [39]. In the retrieval context, looser translations are generally acceptable to find relevant documents. This is mainly because approaches for information retrieval are usually based on the bag-of-words assumption and rely on query translation with synonyms and expansion through adding related words. The most vital criterion when considering query translation is the selection of proper translation words [39]. For that reason, SMT systems, particularly probabilistic word-level translation systems based on word alignment techniques, are still the most widely used query translation techniques.

- **Statistical Machine Translation (SMT).** One of the fundamental components of SMT is word alignment [41]. The alignments are usually phrase-based and constructed using various heuristic or statistic alignment methods. The probability of a source sentence, s^F, given the target sentence, s^E, is calculated as: $P(s^F \mid s^E) = \sum_a P(s^F, a \mid s^E)$, where a is a "hidden" alignment from foreign language to target language and a word alignment model can estimate $P(s^F, a \mid s^E)$. For query translation, usually only IBM model 1 is performed [39]. Zbib et al. [55] proposed a neural model to better estimate word translation probabilities for resource-lean CLIR settings compared to statistical alignment methods by incorporation context and character-level encoding.
- **Neural Machine Translation (NMT).** Even though NMT systems have received increasing interest in recent years, there appears to be limited study of the application of NMT for the CLIR task. Sarwar et al. [47] proposed a relevance-based NMT model using a multi-task learning architecture for the CLIR task. Even though they found reasonable performance with NMT for CLIR with their own implementation, we achieved better results by training an NMT architecture from Fairseq [58] with the same parallel corpus. Fairseq is a stable sequence to sequence learning library from Facebook; we use the convolutional NMT architecture [17] from that library. We attribute the performance gain to the much larger number of training epochs and the stability of Fairseq. However, we still found that NMT struggles to match the performance of SMT.
- **Dictionary-based Translation.** A typical query translation approach is machine-readable bilingual dictionary. Query terms are translated by a look up in the dictionary and taking some or all of the translations in the target language. However, mostly due to the vocabulary coverage problem corpus-based SMT models are usually preferred [39].

2.2 Cross-Lingual Embedding (CLE)

Following a broad use of monolingual dense representation pretraining methods for downstream tasks, extensive efforts have been made toward developing cross-lingual representations to support multi-linguality. Such representations are designed to learn vectors in a shared space for two or more languages such that words with similar meanings obtain similar vectors regardless of language [19].

Ruder et al. [45] presents a comprehensive typology of various CLE models based on the data requirements and objective functions. Here, we use CLEs in two ways: (1) query translation to the target language, (2) obtaining shared representation for query and documents. CLE methods are generally more efficient in terms of time and resource compared to MT models.

In terms of data requirements, there are three main types of methods [45]: *(a) Projection-based:* monolingual word embeddings are trained independently and then a transformation matrix is learned; (*b*) *Pseudo-bilingual:* a corpus of the mixture of source and target words is constructed and then a monolingual method is applied to learn the cross-lingual representations; and, *(c) Joint:* a bilingual corpus is used for jointly minimizing the loss function for monolingual and bilingual term pairs. Due to the number of CLE methods in the literature, covering all of them here is almost impossible. Following Glavas et al. [19]'s findings on the strong correlation of CLIR performance with BLI task as well as our initial explorations, we use the following four strong baselines as indicative reference points on each type of methods described above.

- **PRJ-UNSUP** is an unsupervised projection-based approach based on Conneau et al. [9]. After building monolingual representations, it uses a three-step procedure to train a rotation matrix, W. *First,* it exploits a domain-adversarial learning setting to learn a roughly aligned space using W. *Second,* the Procrustes solution [49] is applied to further refine the first step alignments. *Third,* A process called cross-domain similarity local scaling is conducted to expand the space where there is high density of points. This method uses a self-learning procedure to iteratively augment the lexicons, relaxing the initial dictionary requirement.
- **PRJ-SUP** is a supervised projection-based approach based on Artetxe et al. [1]. Again, after building monolingual representations for source and target languages, an iterative solution applied to get the alignments, similar to Conneau et al. [9], using a word-level dictionary, such that frequent words are aligned in the first step. Until convergence, the latest alignment is used to improve the projection.
- **PSD** is a pseudo-bilingual corpora-based approach based on Levy et al. [28]. In line with some earlier methods like BilBOWA [20], [28] investigate the use of Dice statistical word aligner [41] for CLE construction and report a significant improvement on several benchmarks. PSD uses skip-gram with negative sampling [36] to construct cross-lingual representations.
- **JNT** is a joint training approach based on Duong et al. [15]. In line with some earlier approaches like Vulić and Moens [51]'s BWESG, using the contextual bag of words (CBOW) [35, 36] embedding construction and similar to Gouws and Søgaard [21] for translation replacement in the correct context, Duong et al. [15] use a bilingual dictionary to exploit the context in one language to predict the translation of the pivot word in the other language. Using an expectation-maximization (EM) algorithm for selecting the correct translation from the dictionary, each pivot word is replaced with a translation on-the-fly during CBOW training. The algorithm is forced to predict the word and its translation from a monolingual context.

We use these CLE methods in a bilingual scenario that is typically observed in machine translation. We use parallel corpus to train MT models as well as our baseline CLEs. We observe that training our baseline CLEs with parallel corpus results in improved CLIR performance when compared to their pre-trained versions trained using Wikipedia-based comparable corpora. In addition to these CLEs and because of the recent success of transformer-based contextualized dense representations [13] for monolingual IR [33, 40, 53], we also examine the representation quality of an extended model, XLM [27], that jointly learns to encode over 100 languages into a shared space. XLM extends the Masked Language Model (MLM) training objective of BERT for multilingual scenarios and uses a Translation Language Model (TLM) objective for sentence-level parallel data. XLM-R [8] improves upon XLM by incorporating more training data and languages, particularly low-resource languages. In our CLIR experiments, we observe that XLM-R does not provide much of performance gain. Here, we only use XLM's pre-trained models.

2.3 Existing Neural CLIR Models

There appear to be few studies of neural CLIR (re-)ranking methods. Most of the existing work is focused on representation construction. The global methods rely on external translation resources and are pre-trained for retrieval. Vulić and Moens [51]'s unified monolingual and cross-lingual framework is probably one of the earliest global representation construction methods. They proposed an unsupervised semantic ranking method based on *cosine* similarity. Litschko et al. [31] extended the framework and used off-the-shelf pre-trained CLEs to combine query translation and semantic space rankings. For local representation learning, Dadashkarimi et al. [10, 11] used pseudo-relevant documents for CLIR.

Li and Cheng [29] employed an adversarial framework to learn relevance-based dense representations. Even though Google Translate is used for query log translations, the retrieval performance is lower compared to the MT. Sasaki et al. [48] constructed synthetic cross-lingual retrieval data based on Wikipedia comparable corpora, and proposed a shallow learning-to-rank method. Zhao et al. [57] leverages the sentence-aligned parallel data as a weak supervision signal for training neural CLIR models for low-resource languages. However, none of these target the *standard* CLIR tasks with deep architectures.

3 SMART SHUFFLING CLE

We introduce *Smart Shuffling*, a CLE method that interleaves tokens from sentences in two different languages that are translations of each other. Inspired by statistical word alignment approaches [20, 41], it uses a dictionary to guide this re-ordering process – in contrast to other approaches that interleave (shuffle) the sentences randomly – so that words are placed near their translation's context. After the data has been shuffled this way, a monolingual word embedding approach can be applied to generate the CLE. We train Smart Shuffling CLE using bilingual parallel text and dictionary data[1]. As mentioned above, we hypothesize (and our experiments show) that neural CLIR fails because typical cross-lingual embeddings "translate" query terms into *related* terms in addition to or sometimes rather than synonyms in the target language. A similar

[1] With the development of open source dictionaries (e.g., Panlex and Wiktionary) word-level parallel data is available for almost every language pair.

phenomenon has been shown for monolingual neural IR as the gap between query and document words [38].

The underlying algorithm for CLE construction is based on the monolingual embedding methods, i.e., CBOW and Skip-Gram [35]. CBOW predicts a pivot word from its context over a window of size *ws* centered around the pivot, while Skip-Gram predicts the context for a given pivot. Our Smart Shuffling is in line with Duong et al. [15] and Gouws and Søgaard [21] CLE methods. All these exploit CBOW as the embedding learning method and replace each word in the source or target language with its translations during the training. As a result, for a pivot word in the target language the context is only from the source language, i.e., monolingual context.

Smart Shuffling provides bilingual context for a given pivot word, containing the related monolingual words, the translation of context words, and the translation of the pivot word. With Smart Shuffling, contrary to similar methods, we can lower the context window size *ws* and enforce the joint training with a tighter bilingual context. Our departure point is a recent experimental study on the importance of *ws* for high quality word embedding construction in monolingual text [23]. More specifically, Lison and Kutuzov [30] investigate window sizes of $ws = \{1, 2, 5, 10\}$, along with some other factors and show that for different tasks *ws* can affect the quality of the learned representations. Comparing these results with the analysis of Vulić and Moens [51] on the window size reveals an important point: randomly shuffling the raw text of two languages *multiple* times flattens the contextual information across languages and eliminates the natural ordering of terms. This possibly explains Vulić and Moens [51]'s sensitivity analysis showing indifferent results with $ws = \{10, 20, \cdots, 100\}$.

To explain Smart shuffling we start by formally describing our parallel data. Let the source language, F, and the target language, E, be the pair of languages (e.g., French and English). We have sentence-level parallel data as $P = \{(s_1^F, s_1^E), (s_2^F, s_2^E), \cdots, (s_m^F, s_m^E)\}$ with m pairs of sentences in the source and target languages, and a bilingual dictionary as $\bar{w}_i = D(w_i)$, providing translations of term w_i in the other language. For the k^{th} sentence pair, let $\ell_k^F = |s_k^F|$ and $\ell_k^E = |s_k^E|$ be the number of words in the source and target sentences, respectively. Let $s_k^F =< f_1, f_2, \cdots, f_{\ell_k^F} >$ and $s_k^E =< e_1, e_2, \cdots, e_{\ell_k^E} >$ be the sequence within the source and target sentences, respectively.

The alignment, $A \in \mathbb{R}^{\ell_k^F \times \ell_k^E}$, is defined as a Cartesian product of the word positions [41]. We consider four possible scenarios for the words $f_i \in s_k^F$ and $e_j \in s_k^E$, in ordered priority. Typically, this alignment representation is restricted to assign *one* target word for each source word. The cell $(i, j) \in A$ has two scores normalized row-wise and column-wise (i.e., forward and backward normalized scores): the row-wise normalized scores of i^{th} row shows the probability of assigning the i^{th} word in the source language sentence to j^{th} word in the target sentence, i.e., $t(f_i|e_j)$—similarly for $t(e_j|f_i)$. We provide the following heuristic approach to compute a deterministic association or alignment of terms:

(I) ***IF*** $(f_i, e_j) \in D$ ***THEN*** Fill $(i, j) \in A$ with $(1, 1)$,

(II) ***IF*** $f_i = e_j$ ***THEN*** Fill $(i, j) \in A$ with $(1, 1)$,

(III) ***IF*** $(f_i, .)$ or $(., e_j) \in D$ but $(f_i, e_j) \notin D$ ***THEN*** Get the set of translations of f_i as $\bar{f}_i = D(f_i)$. Using the character 3-grams of each token of $\bar{f}_i$ and e_j, we calculate the translation probability as $t(e_j|f_i) = \dfrac{|\text{ngram}(e_j) \cap \text{ngram}(\bar{f}_i)|}{|\text{ngram}(\bar{f}_i)|}$ for forward translation—similar for backward with $\bar{e}_j = D(e_j)$,

(IV) ***ELSE*** Give equal probabilities for the remaining words from s_k^F and s_k^E.

Algorithm 1 Smart Shuffling CLE

```
Input: $s_k^F$: source sentence, $s_k^E$: target sentence, $A_k$: alignment,
    $SI$: #shuffled sentences, $D$: Dictionary
1: for $dir \in \{fw, bw\}$ do
2:     for $idx \in range(SI)$ do
3:         $trans = sample\_alignment(A_k, dir)$
4:         $shuff\_sent = smart(s_k^F, s_k^E, trans)$
5:         $construct\_emb(shuff\_sent, D)$        // See Eq. 1
6:     end for
7: end for
```

For example, for calculating the character 3-gram similarity score, assume e_j = '*exchange*' and $\bar{f}_i$ is D('*transfert*')= {'*changement*'}.

$$\text{ngram}(e_j) = \{\text{ exc, xch, cha, han, ang, nge }\}$$

$$\text{ngram}(\bar{f}_i) = \{\text{ cha, han, ang, nge, gem, ement, men, ent }\}$$

$$\Rightarrow t(e_j|f_i) = \frac{4}{9}$$

Our presented heuristic is a simple yet effective technique that relies on existing dictionary and character 3-gram similarity score, given its effectiveness for sub-word identification across languages [18, 50]. We leave analyzing the other possible variations of the alignment techniques as future work. We believe that using a dictionary to eliminate some otherwise possible shufflings helps to better align terms that are not covered in the dictionary.

Algorithm 1 presents the pseudo-code of Smart Shuffling CLE. In order to respect the ordering of both languages, which might be completely different, we consider both forward and backward directions (line 1). *sample_alignment()* samples a possible translation for a given source word, f_i, based on the alignment translation probability $t(e_j|f_i)$, and construct a one-to-one relation between source and target words (line 3). Based on the direction, *smart()* provides an insertion of source words into target sentence (or vice versa). In this step, we randomly insert the translation before or after the source term, and obtain a bilingual sentence $s_k =< e_2, f_1, f_2, e_1, \cdots, e_{\ell_k^E}, \cdots, f_{\ell_k^F} >$ (line 4). The shuffled sentence is then provided to the embedding construction method (line 5). Similar to the Duong et al. [15]'s modification of the original CBOW, we use the following objective function for *construct_emb()*.

$$\sum_{i \in s_k} (\alpha \log \sigma(\mathbf{u}_{w_i}^{\intercal} \mathbf{h}_i) + (1-\alpha) \log \sigma(\mathbf{u}_{\bar{w}_i}^{\intercal} \mathbf{h}_i) + \sum_{w_j \in NS()} \log \sigma(-\mathbf{u}_{w_j}^{\intercal} \mathbf{h}_i)) \tag{1}$$

where w_i is a token at position i of s_k, $\mathbf{h}_i$ is the context vector averaging vectors of bilingual words in the range of *ws* of the pivot word, and $NS()$ provides negative samples from the combined vocabulary. As discussed before, providing a bilingual context containing synonyms of the pivot word and limiting the access to other context words forces the model to predict synonyms rather than related terms. The dictionary replacement procedure is borrowed from Duong et al. [15] and we use the default $\alpha = 0.5$. We repeat the process SI times (line 2). In our experiments, we set $SI = 10$.

Example. For a French text, *"point transfert ambulancier"* and its corresponding English translation, *"ambulance exchange point"*,

Table 1: The alignment matrix, $A_{example}$

		English		
		ambulance	**exchange**	**point**
French	**point**	(0, 0)	(0, 0)	(1, 1)
	transfert	(0, 0)	(1, 1)	(0, 0)
	ambulancier	(1, 1)	(0, 0)	(0, 0)

Using $A_{example}$, the following is the output of the Smart Shuffling with $SI = 3$. The English terms are underlined. 1-3 are based on the ordering of French sentence (forward direction) and 4-6 are based on English sentence (backward direction).

(1) point point transfert exchange ambulancier ambulance
(2) point point transfert exchange ambulancier ambulance
(3) point point exchange transfert ambulance ambulancier
(4) ambulance ambulancier transfert exchange point point
(5) ambulance ambulancier exchange transfert point point
(6) ambulancier ambulance exchange transfert point point

4 EXPERIMENTAL EVALUATION

We investigate four language pairs in our experiments: French to English as *Fre-Eng*, Italian to English as *Ita-Eng*, Finnish to English as *Fin-Eng*, and German to English as *Deu-Eng*. The first language is the language of the query and the second is the language of the collection. We selected these languages based on the number of available query sets and language families.[2]

Text Pre-processing. In order to have consistent pieces of text across different resources, characters are normalized by mapping diacritic characters to the corresponding unmarked characters and lower-casing. We remove non-alphabetic, non-printable, and punctuation characters from each word. The NLTK library [2] is used for tokenization and stop-word removal. No stemming is performed. For XLM, we use each model's BPE codes and vocabulary.

Translation Resources. We use two resources: (i) *Word-level.* We use the Panlex lexicon. Its data acquisition strategy emphasizes high-quality lexical and broad language coverage [25]. (ii) *Sentence-level.* We use the Europarl v7 sentence-aligned corpus [26]. The number of parallel text along with vocabulary size for each language pair are given in Table 2.

Table 2: Statistics of Translation Resources

Lang. Pair	Resource	#Pairs	$\lvert V_F \rvert$	$\lvert V_E \rvert$
Fre-Eng	Panlex	656,297	234,623	247,309
	Europarl	1,995,528	141,338	105,182
Ita-Eng	Panlex	373,655	177,362	165,467
	Europarl	1,894,217	146,036	77,441
Fin-Eng	Panlex	427,506	208,077	134,488
	Europarl	1,905,683	637,902	75,851
Deu-Eng	Panlex	646,925	338,188	240,880
	Europarl	1,901,027	318,715	76,309

Translation Models Training. For SMT, we use the GIZA++ toolkit [41]. Two versions of parallel data are fed to the model: *first*, only Europarl data, *second*, a concatenation of Panlex and Europarl data. For NMT, we use a popular convolutional architecture [17] from Fairseq [43] sequence to sequence learning library. The architecture consists of 512 hidden units for both encoders and decoders. Nesterov's accelerated gradient method with a momentum value of 0.99 and gradient clipping value of 0.1 was used along with a learning rate of 0.25. All models were trained using single-gpu approach with mini-batch size of 64 and a dropout value of 0.2. For CLE models, we use the authors' implementations. For projection-based CLE approaches, we first train monolingual embeddings using FastText library's skip-gram model. All the parameters are their default values in the provided toolkit unless otherwise stated. The CLE embedding size is set to 300 for all of our experiments. The context window size is set using 2-fold cross-validation, with values $\{1, 2, 4, 10, 25, 50\}$.

Query Set and Text Collection. We performed experiments on the Cross-Language Evaluation Forum (CLEF) 2000-2003 campaign for bilingual ad-hoc retrieval tracks [4–7]. We aggregate all four years' track topics and query relevance judgments in order to have a higher number of queries. The text collection for our four language pairs is the Los Angeles Times (LAT94) comprising over 113k news articles.[3] We only use the *text* field of the LAT94 corpus for indexing. Queries are selected from $C001 - C200$ topic set for each language. Queries without any relevant document are excluded, resulting in 151 queries for each language. We use a concatenation of *title* and *description* fields of the topic sets as our queries[4].

Evaluation. For evaluating retrieval effectiveness, we report Mean Average Precision (MAP) of the top 1000 ranked documents, and precision of the top 10 retrieved documents (P@10). Statistically significant differences of MAP and P@10 values are determined using the two-tailed paired t-test with $p\_value < 0.05$ (i.e. 95% confidence level).

4.1 Term-Matching Retrieval

Let the query, $q_F \in Q_F$, be in source language F, with constituent terms $\{q_1^F, \cdots, q_{|q_F|}^F\}$, and the document, $d_E \in D_E$, in target language E, with constituent terms $\{d_1^E, \cdots, d_{|d_E|}^E\}$. The query is translated term-by-term and then the retrieval is conducted. For SMT, GIZA++ provides a translation table with the probability of translation. For CLE, we use cosine similarity to find translations of each query term with their translation score. Using either of these translation methods, for a given query term we obtain a sorted list of top-T translations with the corresponding score – $T = 1$ is set by default. Formally, for the i^{th} query term, $q_i^F =< \cdots, (q_{(i,j)}^E,\ t(q_{(i,j)}^E | q_i^F)), \cdots >$, where $(1 \leq j \leq T)$ and $t(\cdot|\cdot)$ is the translation probability from F to E. For out-of-vocabulary (OOV) terms in queries, we set $q_i^F =< (q_{(i)}^F,\ 1.0) >$ to allow exact matching. From this point on, any

[2] English and German are in the Germanic language family, Italian and French are in the Romance language family, and Finnish is in the Uralic language family.

[3] In the 2003 track an additional text collection was added to the evaluations, the Glasgow Herald corpus. For consistency in the evaluations, we filter out this collection with its relevance judgments.

[4] Preliminary experiments show that in terms of the translation quality comparisons, short queries are equally disadvantaged across different languages and retrieval models.

Table 3: Query Translation Performance using Term-Matching Ranking. Note that the first three rows are reported as references. For each column the highest value is marked with bold text. Significance tests with respect to SMT (marked with ▹) are marked with $^{\diamond}$, $^{\blacktriangle}$, and $^{\triangledown}$ for no difference, improvement, and degradation, respectively.

		Fre-Eng		Ita-Eng		Fin-Eng		Deu-Eng	
Trans. Model	Trans. Resource	MAP	P@10	MAP	P@10	MAP	P@10	MAP	P@10
Human Translation	CLEF Topics	0.3703	0.3503	0.3703	0.3503	0.3703	0.3503	0.3703	0.3503
Google Translation	Commercial MT	0.3237	0.3291	0.3244	0.3205	0.3195	0.3119	0.3271	0.3238
No Translation	—	0.1346	0.1464	0.1054	0.1007	0.0577	0.0603	0.1309	0.1199
NMT	Europarl	0.2932	0.3007	0.3091	0.3060	0.2571	0.2748	0.2944	0.2840
SMT	Europarl	0.3208	0.3152	0.3142	0.3126	0.2694	0.2689	0.3044	0.2874
Dictionary	Panlex	0.2069	0.2040	0.2282	0.2351	0.1254	0.1371	0.2075	0.2139
▹ SMT	Europarl + Panlex	0.3312	0.3232	0.3140	0.3146	0.2852	0.2834	0.2908	**0.2894**
PRJ-UNSUP	Europarl	0.2866$^{\triangledown}$	0.2669$^{\triangledown}$	0.2905$^{\triangledown}$	0.2934$^{\triangledown}$	0.2073$^{\triangledown}$	0.2146$^{\triangledown}$	0.2673$^{\triangledown}$	0.2616$^{\triangledown}$
PRJ-SUP	Europarl + Panlex	0.2825$^{\triangledown}$	0.2709$^{\triangledown}$	0.2919$^{\triangledown}$	0.2987$^{\triangledown}$	0.1995$^{\triangledown}$	0.2007$^{\triangledown}$	0.2531$^{\triangledown}$	0.2563$^{\triangledown}$
PSD	Europarl	0.2434$^{\triangledown}$	0.2709$^{\triangledown}$	0.2802$^{\triangledown}$	0.2940$^{\triangledown}$	0.1668$^{\triangledown}$	0.1609$^{\triangledown}$	0.1947$^{\triangledown}$	0.2146$^{\triangledown}$
JNT	Europarl + Panlex	0.2803$^{\triangledown}$	0.2748$^{\triangledown}$	0.2917$^{\triangledown}$	0.2987$^{\triangledown}$	0.2404$^{\triangledown}$	0.2371$^{\triangledown}$	0.2608$^{\triangledown}$	0.2457$^{\triangledown}$
Smart Shuf.	Europarl + Panlex	0.3276$^{\diamond}$	0.3162$^{\diamond}$	0.3109$^{\diamond}$	0.3033$^{\triangledown}$	0.2799$^{\diamond}$	0.2728$^{\triangledown}$	0.2914$^{\diamond}$	0.2795$^{\diamond}$
Smart Shuf. with SMT	Europarl + Panlex	**0.3482**$^{\blacktriangle}$	**0.3238**$^{\diamond}$	**0.3253**$^{\blacktriangle}$	**0.3159**$^{\diamond}$	**0.2975**$^{\blacktriangle}$	**0.2901**$^{\diamond}$	**0.3072**$^{\blacktriangle}$	**0.2894**$^{\diamond}$

monolingual ranking method (e.g., probabilistic or language modeling) can be applied [39, 51] to calculate score(q_F, d_E). We use the Galago's implementation[5] of Okapi BM25 [44] with default parameters. For the translated query we exploit the Galago's weighted *#combine* operator. We call this ranking as term-matching retrieval for distinguishing with two other neural (re-)ranking models we use in our study, named as semantic-matching and deep-matching.

4.2 Experimental Results

Table 3 presents our experimental results for term-matching retrieval across different language pairs and translation models. In order to provide indicative reference points, human translation (English queries), Google Translation (a commercial MT system), and no translation (treating the non-English language as if it were English) experiments are also reported. It comes as a surprise to us to see that NMT and commercial translation systems fall short in performance. While these state-of-the-art machine translation systems have been trained for translating sentences, they might struggle to translate bag-of-keywords queries, which do not usually possess a grammatical structure.

As we can see for the SMT model, providing Panlex data improves the retrieval performance for French and Finnish queries in terms of MAP and P@10. For Italian and German queries, even though the MAP values are slightly lower, P@10 values are improved. We interpret these as generally better translation and compare CLE methods with this SMT baseline (marked with ▹). We also included Panlex only query translation performance, in which we take all the translations with equal probabilities for query construction—resulted in a very low retrieval performance.

Considering existing CLE methods trained on the same translation resources (in the third section of the table) we can clearly see a *translation gap* when compared to the SMT model—a statistically significant drop in the retrieval performance for existing state-of-the-art CLE methods. This translation gap is more severe for Finnish and German. This supports our hypothesis on the *translation gap* and answers *RQ1*. We can also partially answer *RQ2*. As we can see, in terms of MAP there is no difference between SMT and Smart Shuffling CLE. However, we see a slight degradation in MAP for all pairs and a significant degradation in P@10 for Italian and Finnish queries.

In order to compare the translated queries between Smart Shuffling CLE and MT systems, and for answering the last part of *RQ2*, we combined the translated query from both models with equal weights. We report results with this combined query in the last row of Table 3. Comparing this combination with both SMT and Smart Shuffling CLE individual retrieval performance shows a statistically significant improvement in MAP values with the combined query. Given the difference in the nature of these two translation models and considering that there is no significant difference in P@10 values, we hypothesize this might be because CLE provides more related and relevant terms whereas SMT provides more synonyms.

5 DEEP-MATCHING RE-RANKING MODEL

In this section, in order to show the intertwined impact of translation and relevance for CLIR, and provide an answer for *RQ3*, we present a deep neural re-ranker based on the Deep Relevance Matching Model (DRMM) [22]. DRMM is a relevance-matching method employing a joint deep architecture at the query term level, proposed for monolingual ad-hoc retrieval task. We specifically selected DRMM for our experiments since it is highly sensitive to the quality of the pre-trained embeddings, given its static use of the provided representation for re-ranking without updating it. In addition, it has a considerably simpler network architecture compared to the other existing methods and is well-suited for data-scarcity situations. However, nothing prevents the use of any other monolingual neural model with the provided extension on the loss function.

[5] https://www.lemurproject.org/galago.php

Table 4: Deep-Matching Re-ranking Results. Significance tests for Smart Shuffling with respect to other CLEs are conducted with Bonferroni correction ($^{\diamond}$ Insignificant, $^{\blacktriangle}$ Improvement, and $^{\triangledown}$ Degradation).

	Fre-Eng		Ita-Eng		Fin-Eng		Deu-Eng	
CLE Model	MAP	P@10	MAP	P@10	MAP	P@10	MAP	P@10
PRJ-UNSUP	0.3148	0.2227	0.3069	0.2245	0.2150	0.1801	0.2453	0.2099
PRJ-SUP	0.3017	0.2203	0.3009	0.2258	0.2066	0.1755	0.2500	0.2106
PSD	0.2395	0.1971	0.2183	0.1748	0.1236	0.1252	0.1533	0.1437
JNT	0.3212	0.2334	0.3175	0.2477	0.1939	0.1616	0.2494	0.1828
Smart Shuffling	**0.3635**$^{\blacktriangle}$	**0.2612**$^{\blacktriangle}$	**0.3707**$^{\blacktriangle}$	**0.2735**$^{\blacktriangle}$	**0.2531**$^{\blacktriangle}$	**0.2106**$^{\blacktriangle}$	**0.3020**$^{\blacktriangle}$	**0.2358**$^{\blacktriangle}$

Table 5: Qualitative Analysis of CLEs. Queries are C071, C106, and C148, respectively—only the title field is presented. For each query translation method, the bold term(s) are the translations that are not synonyms of the original query term.

French Query	English Trans. (Human)	Query Translation with $T = 1$		
		Smart Shuffling	JNT	SMT
Légumes, fruits et cancer	Vegetables, Fruit and Cancer	vegetables fruit cancer	**fruit** fruit cancer	vegetables fruit cancer
L'industrie automobile en Europe	European car industry	industry car europe	industry **industry** europe	industry car europe
Dommages à la couche d'ozone	Damages in Ozone Layer	damage layer ozone	**ravages ozone** ozone	damage layer ozone

Deep-Matching. Using CLE as the pre-trained representation, for each possible pair of terms from the query, q_F, and document, d_E, local interactions are constructed. Then, a fixed-length matching histogram is used for transforming the variable-length local interactions of query terms. A feed-forward matching network is used to learn hierarchical matching patterns. Note that for each query term a matching score is calculated and the overall matching score is aggregated using a term gating network for weighted aggregation. Given (q_F, d_E^+, d_E^-), with d_E^+ denoting a relevant document and d_E^- denoting a non-relevant document, the loss function is defined as below (hinge loss as a pairwise ranking loss).

$$\mathcal{L}(q_F, d_E^+, d_E^-; \Theta) = \max(0, 1 - s(q_F, d_E^+) + s(q_F, d_E^-))$$

$s(q_F, d_E)$ denotes the aggregated score, and Θ denotes the parameters that our deep network learns using the Adadelta [56] optimizer. Guo et al. [22] provide further details on each of these steps. For OOV terms in queries, the model allows exact matching in the document terms.

Hyper-parameters. For the network configuration we tune the hyper-parameters using the parameters both in the original paper and the ones tuned for the MatchZoo implementation[16]. We find that the default parameters in the MatchZoo library perform better. Therefore, in all of our experiments, we use a four layer architecture; (i) A histogram input layer with 60 nodes, (ii) Two hidden layers in the feed forward matching network with 20 and 1 nodes, respectively, (iii) A term gating output layer with 1 node. We also set the batch-size to 100, and use early stopping strategy on 400 epochs for training the model.

Experimental Details. Given the limited number of queries in each language, we use the supervised learning paradigm similar to DRMM experiments [22] through a 5-fold cross-validation. For each fold, the training, validation, and test data are 60%, 20%, and 20% of the query set, respectively. The reported evaluation values are averaged across 5 folds. For the initial retrieval, we use our Term-Matching ranking model with SMT translation to obtain the top 1000 documents for each language pair. We select this initial retrieval system to provide the same set of relevant and non-relevant documents when the model is training using different CLEs.

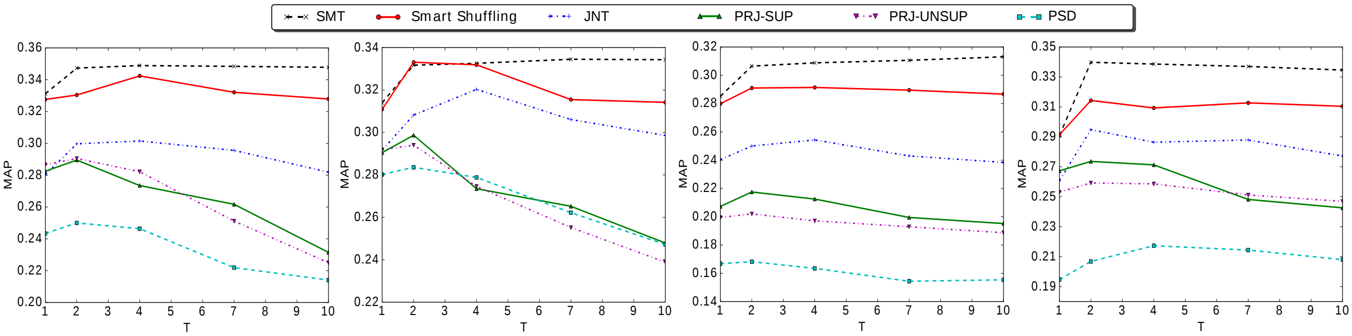
Experimental Results. Table 4 presents our experimental results for Deep-Matching re-ranking model across different language pairs and translation models. Comparing the Smart Shuffling results with the existing CLE methods, we see 13%, 17%, 18%, and 21% relative improvements in terms of MAP for French, Italian, Finnish, and German queries, respectively, compared to the highest MAP value from the existing CLE models. Further, comparing deep-matching CLIR results with Table 3 shows that for French and Italian queries human translation retrieval performance are approached, in terms of MAP. However, we still see a low precision when compared to term-matching results. We hypothesize that this might be due to the fundamental differences in the ranking models. Another interesting observation is the comparison of retrieval performance across languages. As we can see for Finnish and German query languages the retrieval performance is still lower compared to Table 3, i.e. no improvement. We think this might be due to our heuristic alignment method and the fundamental differences in those languages compared to English. However, this conclusion needs further investigations and we leave it for future work.

6 DISCUSSION AND FURTHER ANALYSIS

Here we further analyze Smart Shuffling to provide insight on the proposed method and discuss its limitations.

6.1 A Qualitative Analysis

Table 5 presents a qualitative comparison of three example French query translations using Smart Shuffling, JNT, and SMT. For these three queries our CLE provides the same translation as the SMT method. Moreover, both the translation approaches provide very close approximation to the human translations. However, JNT does



(a) Fre-Eng (b) Ita-Eng (c) Fin-Eng (d) Deu-Eng

Figure 1: Effect of query expansion with more than one translation.

Table 6: Semantic-Matching Ranking Results. Significance tests for Smart Shuffling with respect to other CLEs are conducted with Bonferroni correction (◇ Insignificant, ▲ Improvement, and ▽ Degradation).

	Fre-Eng		Ita-Eng		Fin-Eng		Deu-Eng	
CLE Model	MAP	P@10	MAP	P@10	MAP	P@10	MAP	P@10
PRJ-UNSUP	0.2156	0.2124	0.1781	0.1888	0.1366	0.1464	0.1485	0.1623
PRJ-SUP	0.2208	0.2178	0.1754	0.1881	0.1390	0.1431	0.1558	0.1623
PSD	0.1059	0.1067	0.1388	0.1656	0.0799	0.0980	0.0857	0.1113
JNT	0.2307	0.2211	0.2024	0.2080	0.1416	0.1404	0.1696	0.1775
XLM (MLM_100)	0.1648	0.1795	0.1292	0.1550	0.1205	0.1384	0.1043	0.1172
XLM (MLM_17)	0.0673	0.0636	0.0902	0.0993	—	—	0.0887	0.1026
XLM (MLM_TLM_15)	0.0004	0.0000	—	—	—	—	0.0006	0.0000
Smart Shuffling	**0.2559▲**	**0.2424▲**	**0.2575▲**	**0.2517▲**	**0.1970▲**	**0.2073▲**	**0.2307▲**	**0.2332▲**

not provide accurate translations as it is focused toward related words. For the last query in Table 5 JNT provide “ravages” as the translation of the first french query term and our approach provides “damages” which is the same as the human provided translation. We note that “ravages” is not an unreasonable translation - it is just not the exact one. The same scenario is happening for the other two example queries. We have marked the possibly inaccurate translations generated using JNT with bold text. This analysis shows that the translation errors are propagated in the neural re-ranking model for the other CLE methods preventing the neural IR approach from providing an improved ranked list.

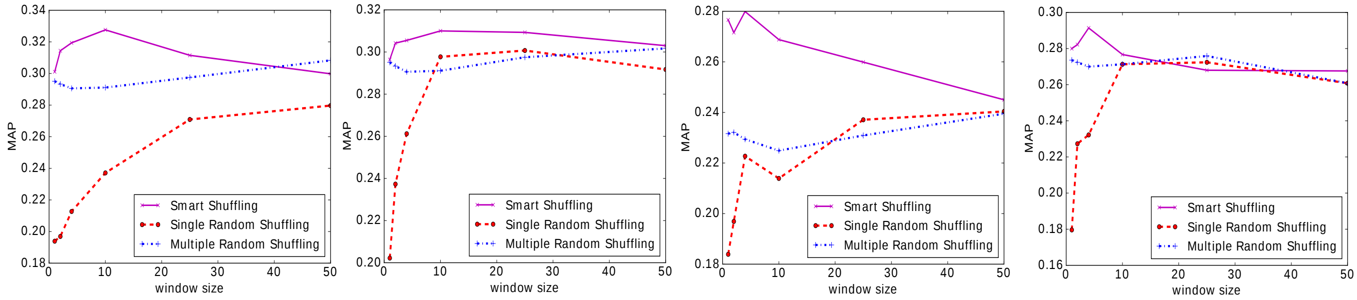
6.2 Incorporating More Translation Terms

Given the special characteristics of the translation task for CLIR, providing more translation terms for each query term is usually helpful for the retrieval performance [39]. This is a mixture of translation and expansion for a query with $T > 1$ terms. To analyse the effect, we use the top T translations with $T \in \{1, 2, 4, 7, 10\}$ as the translated query. Figure 1 shows that in almost all language pairs, when 2 or more translations are available, CLEs do not yet outperform SMT query translation. For French and Italian queries, we see that the gap is narrower compared to other two languages. This might be due to the language similarities with the English language as the target language. Among CLE methods, we can see that Smart Shuffling outperforms every baseline. Interestingly, projection-based CLE methods perform poorly when more translations are extracted.

6.3 Semantic-Matching Ranking Model

In order to further show the importance of the CLE quality, we conduct a semantic ranking similar to Litschko et al. [31]'s BWE-AGG method. Using CLE's dense vector representations, the vector of queries and documents are derived using simple aggregation of their constituent terms. The query representation is simply the summation of query term vectors, i.e. $\vec{q}_F = \sum_{i=1}^{|q_F|} \vec{q}_i^{\,F}$. For the document representation, it has been shown that inverse document frequency (IDF) weighted summation operator outperforms unweighted summation [31]. The document representation is $\vec{d}_E = \sum_{j=1}^{|d_E|} idf(d_j^E) \cdot \vec{d}_j^{\,E}$. For OOV terms, we ignore the term. For XLM model, after some investigation of different query and document vector aggregation methods (for example using *'[CLS]'* or individual token-wise representations) we decided to use individual tokens representations, as it provided higher ranking results. We employ dot-product similarity scoring for ranking documents, i.e. score(q_F, d_E). We call this ranking Semantic-Matching.

Table 6 presents our experimental results for Semantic-Matching across different language pairs and CLE models. We can see a statistically significant improvement in terms of MAP and P@10 values for all four language pairs. Comparing with JNT method, we can see 11%, 27%, 39%, and 36% relative improvements, in terms of MAP, for Fre-Eng, Ita-Eng, Fin-Eng, and Deu-Eng language pairs, respectively. The increase in retrieval performance is comparably higher for Finnish and German languages. This might be due to language structure differences between each of these languages with English.



(a) Fre-Eng (b) Ita-Eng (c) Fin-Eng (d) Deu-Eng

Figure 2: Sensitivity analysis on embedding construction context window size for different shuffling techniques.

Table 7: BLI Performance Results (MAP). Significance tests for Smart Shuffling with respect to other CLE methods are conducted with Bonferroni correction.

CLE Model	Fre-Eng	Ita-Eng	Fin-Eng	Deu-Eng
PRJ-UNSUP	0.0051	0.5189	0.1882	0.4018
PRJ-SUP	0.4268	0.5538	0.3446	0.4549
PSD	0.6126	0.5958	0.3122	0.5356
JNT	0.5111	0.6774	0.4718	0.6463
Smart Shuffling	**0.7197**▲	**0.7363**▲	**0.5121**▲	**0.6852**▲

Moreover, XLM results show that such pre-trained models with many languages are not providing high gain for CLIR and needs further investigations for fine-tuning or training. Our interesting observation is that, with TLM fine-tuning, which requires parallel sentence-level data, the ranking performance dropped significantly.

6.4 Smart vs. Random Shuffling

As discussed in Section 3, we analyze the impact of shuffling on the quality of embeddings. Using our method, we analyze three different shuffling methods for all four language pairs. Figure 2 shows the retrieval performance, in terms of MAP. For each of these, embeddings are constructed with different window sizes, $\{1, 2, 4, 10, 25, 50\}$. Note that multiple random shuffling matches with Vulić and Moens [51]'s method. We also randomly shuffle only once for demonstration purposes. For larger window sizes, all the shuffling methods are almost indistinguishable, but comparing the multiple random shuffling with single random shuffling shows that shuffling multiple times brings every possible pair of terms close to each other—i.e., *context flattening*. However, with Smart Shuffling, we find a peak point for window size ranging from 4 to 10. When the window size is large any shuffling can provide the context needed.

6.5 BLI Evaluation

We also evaluate Smart Shuffling using the Bilingual Lexicon Induction (BLI) task that is mainly designed for the CLE quality assessments. The task is to rank target language translations for a given set of source language terms. We use the test sets provided by Glavas et al. [19] for our language pairs, each containing 2K words, and report our results in terms of MAP in Table 7. Smart Shuffling provides significant improvements for the BLI task. Comparing our results with the performance reported by Glavas et al. [19], Smart Shuffling is achieving state-of-the-art results on the provided data. For instance, the best reported BLI performance for Deu-Eng on their test data is 0.58, which Smart Shuffling improves on by 18%. We also note that some part of this high BLI performance, when compared to the reported results in Glavas et al. [19], might be due to sentence-level data quality that we use – our JNT baseline provides 0.64 for the same language pair trained on Europarl data.

7 CONCLUSION

In this study, we paved the way toward an end-to-end neural CLIR model. First, we showed a translation gap between existing state-of-the-art CLE methods when compared to an effective MT model, trained on the same data. We proposed Smart Shuffling, a special embedding construction method that improved the retrieval performance significantly when compared to other CLE methods used with unsupervised ranking methods. Finally, employing a deep relevance-based re-ranking method and training in a supervised paradigm, we showed convincingly that the type of distributed representations of the query and document terms impacts neural CLIR performance and that better CLE approaches – e.g., Smart Shuffling as shown in Table 3 and Table 6 – result in substantially stronger results (Table 4). We also showed that Smart Shuffling's improvements generalize well to a second task (BLI).

Two major and fundamental steps are needed as future work to gain a better understanding toward neural CLIR. The first is to re-design the monolingual neural IR systems to model a joint loss in terms of translation and relevance. The second is to extract more translation resources that can be used as synthetic queries with translations for training the neural models in a weak supervision paradigm [12]. We are particularly interested in extending our findings with Smart Shuffling into transformer-based pre-trained models and providing CLIR extension of recent success in fine-tuning BERT for monolingual IR [13, 53].

Acknowledgements. This work was supported in part by the Center for Intelligent Information Retrieval, in part under USC (University of Southern California) subcontract no. 124338456 under IARPA prime contract no. 2019-19051600007, and in part by the Air Force Research Laboratory (AFRL) and IARPA under contract #FA8650-17-C-9118 under subcontract #14775 from Raytheon BBN Technologies Corporation. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor.

REFERENCES

[1] Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. In *ACL'18*, Vol. 1. 789–798.

[2] Steven Bird and Edward Loper. 2004. NLTK: The natural language toolkit. In *ACL'04*. 31.

[3] Hamed Bonab, James Allan, and Ramesh Sitaraman. 2019. Simulating CLIR translation resource scarcity using high-resource languages. In *ICTIR'19*. 129–136.

[4] Martin Braschler. 2000. CLEF 2000 - overview of results. In *Workshop of Cross-Language Evaluation Forum, CLEF 2000. Lisbon, Portugal.* 89–101.

[5] Martin Braschler. 2001. CLEF 2001 - overview of results. In *Workshop of the Cross-Language Evaluation Forum, CLEF 2001. Darmstadt, Germany.* 9–26.

[6] Martin Braschler. 2002. CLEF 2002 - overview of results. In *Workshop of the Cross-Language Evaluation Forum, CLEF 2002. Rome, Italy.* 9–27.

[7] Martin Braschler. 2003. CLEF 2003 - overview of results. In *Workshop of the Cross-Language Evaluation Forum, CLEF 2003. Trondheim, Norway.* 44–63.

[8] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116* (2020).

[9] Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. Word translation without parallel data. *ICLR'18* (2018).

[10] Javid Dadashkarimi, Mahsa S Shahshahani, Amirhossein Tebbifakhr, Heshaam Faili, and Azadeh Shakery. 2017. Dimension projection among languages based on pseudo-relevant documents for query translation. In *ECIR'17*. 493–499.

[11] Javid Dadashkarimi, Azadeh Shakery, Heshaam Faili, and Hamed Zamani. 2017. An expectation-maximization algorithm for query translation based on pseudo-relevant documents. *Information Processing & Management* 53, 2 (2017), 371–387.

[12] Mostafa Dehghani, Hamed Zamani, Aliaksei Severyn, Jaap Kamps, and W Bruce Croft. 2017. Neural ranking models with weak supervision. In *SIGIR'17*. 65–74.

[13] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[14] Fernando Diaz, Bhaskar Mitra, and Nick Craswell. 2016. Query expansion with locally-trained word embeddings. In *ACL'16*, Vol. 1. 367–377.

[15] Long Duong, Hiroshi Kanayama, Tengfei Ma, Steven Bird, and Trevor Cohn. 2016. Learning crosslingual word embeddings without bilingual corpora. In *EMNLP'16*. 1285–1295.

[16] Yixing Fan, Liang Pang, JianPeng Hou, Jiafeng Guo, Yanyan Lan, and Xueqi Cheng. 2017. Matchzoo: A toolkit for deep text matching. *arXiv preprint arXiv:1707.07270* (2017).

[17] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. 2017. Convolutional sequence to sequence learning. In *ICLML'17*. 1243–1252.

[18] Daniela Gerz, Ivan Vulić, Edoardo Ponti, Jason Naradowsky, Roi Reichart, and Anna Korhonen. 2018. Language modeling for morphologically rich languages: Character-aware modeling for word-level prediction. *TACL* 6 (2018), 451–465.

[19] Goran Glavas, Robert Litschko, Sebastian Ruder, and Ivan Vulic. 2019. How to (properly) evaluate cross-lingual word embeddings: On strong baselines, comparative analyses, and some misconceptions. In *ACL'19*.

[20] Stephan Gouws, Yoshua Bengio, and Greg Corrado. 2015. BILBOWA: Fast bilingual distributed representations without word alignments. In *ICML'15*. 748–756.

[21] Stephan Gouws and Anders Søgaard. 2015. Simple task-specific bilingual word embeddings. In *NAACL'15*. 1386–1390.

[22] Jiafeng Guo, Yixing Fan, Qingyao Ai, and W Bruce Croft. 2016. A deep relevance matching model for ad-hoc retrieval. In *CIKM'16*. ACM, 55–64.

[23] Wenpeng Hu, Jiajun Zhang, and Nan Zheng. 2016. Different contexts lead to different word embeddings. In *COLING'16 (Technical Papers)*. 762–771.

[24] Martin Josifoski, Ivan S. Paskov, Hristo S. Paskov, Martin Jaggi, and Robert West. 2019. Crosslingual document embedding as reduced-rank ridge regression. In *WSDM '19*. 744–752.

[25] David Kamholz, Jonathan Pool, and Susan M Colowick. 2014. PanLex: Building a resource for panlingual lexical translation. In *LREC'14*. 3145–3150.

[26] Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT summit*, Vol. 5. 79–86.

[27] Guillaume Lample and Alexis Conneau. 2019. Cross-lingual Language Model Pretraining. *arXiv preprint arXiv:1901.07291* (2019).

[28] Omer Levy, Anders Søgaard, and Yoav Goldberg. 2017. A strong baseline for learning cross-lingual word embeddings from sentence alignments. In *EACL'17*, Vol. 1. 765–774.

[29] Bo Li and Ping Cheng. 2018. Learning neural representation for CLIR with adversarial framework. In *EMNLP'18*. 1861–1870.

[30] Pierre Lison and Andrey Kutuzov. 2017. Redefining context windows for word embedding models: An experimental study. *arXiv preprint arXiv:1704.05781* (2017).

[31] Robert Litschko, Goran Glavaš, Simone Paolo Ponzetto, and Ivan Vulić. 2018. Unsupervised cross-lingual information retrieval using monolingual data only. In *SIGIR'18*. 1253–1256.

[32] Robert Litschko, Goran Glavaš, Ivan Vulic, and Laura Dietz. 2019. Evaluating resource-lean cross-lingual embedding models in unsupervised retrieval. In *SIGIR'19*. 1109–1112.

[33] Sean MacAvaney, Andrew Yates, Arman Cohan, and Nazli Goharian. 2019. CEDR: Contextualized embeddings for document ranking. In *SIGIR'19*. 1101–1104.

[34] J Scott McCarley. 1999. Should we translate the documents or the queries in cross-language information retrieval?. In *ACL'99*. 208–214.

[35] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS'13*. 3111–3119.

[36] Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013. Linguistic regularities in continuous space word representations. In *NAACL'13*. 746–751.

[37] Bhaskar Mitra and Nick Craswell. 2018. An introduction to neural information retrieval. *Foundations and Trends in Information Retrieval* (2018).

[38] Bhaskar Mitra, Eric Nalisnick, Nick Craswell, and Rich Caruana. 2016. A dual embedding space model for document ranking. *arXiv preprint arXiv:1602.01137* (2016).

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

[40] Rodrigo Nogueira, Wei Yang, Kyunghyun Cho, and Jimmy Lin. 2019. Multi-stage document ranking with BERT. *arXiv preprint arXiv:1910.14424* (2019).

[41] Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics* 29, 1 (2003), 19–51.

[42] Kezban Dilek Onal, Ye Zhang, Ismail Sengor Altingovde, Md Mustafizur Rahman, Pinar Karagoz, Alex Braylan, Brandon Dang, Heng-Lu Chang, Henna Kim, Quinten McNamara, et al. 2018. Neural information retrieval: At the end of the early years. *Information Retrieval Journal* 21, 2-3 (2018), 111–182.

[43] Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. Fairseq: A fast, extensible toolkit for sequence modeling. In *NAACL-HLT 2019: Demonstrations*.

[44] Stephen E Robertson, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, Mike Gatford, et al. 1995. Okapi at TREC-3. *NIST Special Publication* 109 (1995), 109.

[45] Sebastian Ruder, Ivan Vulić, and Anders Søgaard. 2017. A survey of cross-lingual word embedding models. *arXiv preprint arXiv:1706.04902* (2017).

[46] Shadi Saleh. 2017. *Cross-lingual information retrieval systems.* Ph.D. Dissertation. Institute of Formal and Applied Linguistics, Charles University in Prague.

[47] Sheikh Muhammad Sarwar, Hamed Bonab, and James Allan. 2019. A multi-task architecture on relevance-based neural query translation. In *ACL'19*. 6339–6344.

[48] Shota Sasaki, Shuo Sun, Shigehiko Schamoni, Kevin Duh, and Kentaro Inui. 2018. Cross-lingual learning-to-rank with shared representations. In *NAACL'18*, Vol. 2. 458–463.

[49] Peter H Schönemann. 1966. A generalized solution of the orthogonal procrustes problem. *Psychometrika* 31, 1 (1966), 1–10.

[50] Jesús Vilares, Michael P Oakes, and Manuel Vilares. 2007. Character N-grams translation in cross-language information retrieval. In *International Conference on Application of Natural Language to Information Systems.* Springer, 217–228.

[51] Ivan Vulić and Marie-Francine Moens. 2015. Monolingual and cross-lingual information retrieval models based on (bilingual) word embeddings. In *SIGIR'15*. 363–372.

[52] Shijie Wu and Mark Dredze. 2019. Beto, Bentz, Becas: The surprising cross-lingual effectiveness of BERT. *arXiv preprint arXiv:1904.09077* (2019).

[53] Wei Yang, Haotian Zhang, and Jimmy Lin. 2019. Simple applications of BERT for ad hoc document retrieval. *arXiv preprint arXiv:1903.10972* (2019).

[54] Hamed Zamani and W Bruce Croft. 2016. Embedding-based query language models. In *ICTIR'16*. 147–156.

[55] Rabih Zbib, Lingjun Zhao, Damianos Karakos, William Hartmann, Jay DeYoung, Zhongqiang Huang, Zhuolin Jiang, Noah Rivkin, Le Zhang, Richard Schwartz, and John Makhoul. 2019. Neural-network lexical translation for cross-lingual IR from text and speech. In *SIGIR'19*. 645–654.

[56] Matthew D Zeiler. 2012. ADADELTA: An adaptive learning rate method. *arXiv preprint arXiv:1212.5701* (2012).

[57] Lingjun Zhao, Rabih Zbib, Zhuolin Jiang, Damianos Karakos, and Zhongqiang Huang. 2019. Weakly supervised attentional model for low resource ad-hoc cross-lingual information retrieval. In *Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019)*. 259–264.

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