Query Translation for Cross-Language Information Retrieval using Multilingual Word Clusters

Paheli Bhattacharya, Pawan Goyal and Sudeshna Sarkar
Department of Computer Science and Engineering
Indian Institute of Technology Kharagpur
West Bengal, India
paheli@iitkgp.ac.in, {pawang,sudeshna}@cse.iitkgp.ernet.in

Abstract

In Cross-Language Information Retrieval, finding the appropriate translation of the source language query has always been a difficult problem to solve. We propose a technique towards solving this problem with the help of multilingual word clusters obtained from multilingual word embeddings. We use word embeddings of the languages projected to a common vector space on which a community-detection algorithm is applied to find clusters such that words that represent the same concept from different languages fall in the same group. We utilize these multilingual word clusters to perform query translation for Cross-Language Information Retrieval for three languages - English, Hindi and Bengali. We have experimented with the FIRE 2012 and Wikipedia datasets and have shown improvements over several standard methods like dictionary-based method, a transliteration-based model and Google Translate.

1 Introduction

With the advancement of the Web and availability of multilingual contents, searching over the Web is not limited only to one's native language but is extended to other languages as well. Relevant and adequate information may not always be available in only one particular language but may be spread across other languages. This gives rise to the necessity of Cross-Language Information Retrieval (CLIR, where only two languages are involved) and Multilingual Information Retrieval (MLIR, where more than two languages are involved), where the query and the documents do not belong to a single language only. Specifically, in CLIR, the user query is in a language different than the collection.

Since the language of the query is different from the language of the documents in CLIR and MLIR, a translation phase is necessary. Translating documents is a tedious task. So the general standard is to translate the query and we follow the query translation approach for CLIR. Common or popular approaches for query translation include, but are not limited to, leveraging bilingual or multilingual dictionaries, Statistical Machine Translation (SMT) systems, transliteration based models, graph-based models and online translation systems like Bing and Google Translate.

Each of the approaches have their own advantages and disadvantages. For instance, SMTs require parallel corpus and for languages such as Indian languages where such resources are scarce, SMTs are not very suitable. The dictionary based approaches require substantial word pair translations and suffer from coverage issues and data sparsity problems. We study the effectiveness of word embeddings in such a scenario where we want to have good quality translations that can improve CLIR performance in spite of having a scarcity in data-aligned resources.

Representing words using low dimensional vectors, called word embeddings, are now being widely used in many Natural Language Processing tasks. Each dimension of the vector represents a latent feature capturing useful properties. It has been seen that in the distributional space defined by the vector dimensions, syntactically and semantically similar words are close to each other. In the multilingual space, the objective is to have similar representations of similar words across different languages.

Proceedings of the 6th Workshop on South and Southeast Asian Natural Language Processing,
pages 152–162, Osaka, Japan, December 11-17 2016.

However, using the translations obtained from multilingual word embeddings directly has some drawbacks – words that are not much relevant to the source language word, may also come up as a translation. For instance, for the word "*desha*" (meaning, country) in Hindi, although correct translations like "country" and "democracy" were provided, irrelevant words like "aspiration" and "kind" also showed up as potential translations. Inclusion of such non-related words in a query greatly harms the IR performance. To address this problem, we propose to use multilingual clustering. In multilingual clustering, words from the same as well as across language, that more likely to represent similar concepts, fall in the same group. We use the multilingual embeddings to build these clusters. When multilingual clusters were used, candidate English translations besides "country" and "democracy" for our running example "*desha*" were "nation" and "cities". Our proposed method has shown significant improvements over dictionary-based method, a transliteration-based model and Google Translate.

The rest of the paper is organized as follows: Section 2 discusses recent work in the fields of Cross-Language Information Retrieval and Word Embeddinggs. In Section 3, we describe our proposed approach. The experimental settings and results have been covered in Section 4. Finally, we conclude in Section 5.

2 Related Work

2.1 Word Embeddings

Mikolov et. al (2013a) proposed a neural architecture that learns word representations by predicting neighbouring words. There are two main methods by which the distributed word representations can be learnt. One is the Continuous Bag-of-Words (CBOW) model that combines the representations of the surrounding words to predict the word in the middle. The second is the Skip-gram model that predicts the context of the target word in the same sentence. GloVe or Global Vectors (Pennington et al., 2014) is another unsupervised learning algorithm for obtaining word vectors.

2.2 Cross-lingual Vector Representations

The two major ways to learn word representations in the cross-lingual domain are to either first train the embeddings of the words separately for the languages and then project them to a common space (Faruqui and Dyer, 2014; Mikolov et al., 2013b) or co-learn the embeddings jointly for both monolingual and cross-lingual domains (Gouws et al., 2015; Luong et al., 2015).

Faruqui and Dyer (2014) uses Canonical Correlation Analysis (CCA) that maps words from two different languages in to a common, shared space. (Mikolov et al., 2013a) builds a translation matrix using linear regression that transforms the source language word vectors to the target language space. Huang et. al (2015) constructs translation invariant word embeddings by building on (Faruqui and Dyer, 2014). It performs matrix factorization where the matrices include a multilingual co-occurence matrix and other matrices based on the dictionary. Gouws and Søgaard (2015) uses a task-specific dictionary, i.e., a list of word pairs that are equivalent in some respect, depending on the task. Using a non-parallel corpora, given a sentence in one language, for each word in the sentence, equivalent words are substituted in its place. Then the CBOW model of the word2vec tool is employed.

Bilingual Bag-of-Words without Alignment (BilBOWA) (Gouws et al., 2015) uses monolingual datasets coupled with sentence aligned parallel data to learn word embeddings. They utilize the Skip-Gram model of word2vec to learn the monolingual features and a sampled bag-of-words technique for each parallel sentence as the cross-lingual objective. Chandar et al. (2014) shows that by learning to reconstruct the bag-of-words representations of aligned sentences, within and between languages, high-quality word representations can be learnt. They use an auto-encoder for this purpose.

Given an alignment link between a word w_1 in a language l_1 and a word w_2 in another language l_2, Luong et al. (2015) uses the word w_1 to predict the neighbours of the word w_2 and vice-versa. Klementiev et. al. (2012) induces distributed representations for a pair of languages jointly. They treat it as a multitask learning problem where each task corresponds to a single word and task relatedness is

derived from co-occurence statistics in bilingual parallel data, with word alignments available.

2.3 Cross-Language Information Retrieval

Hull and Grefenstette (1996), Pirkola (1998), Ballesteros and Croft (1996) perform Cross-Language Information Retrieval through dictionary-based approaches. Littman et al. (1998) performs Latent Semantic Indexing on the term-document matrix. Statistical Machine Translations have also been tried out in (Schamoni et al., 2014; Türe et al., 2012b; Türe et al., 2012a; Sokolov et al., 2014). (Padariya et al., 2008; Chinnakotla et al., 2008) use transliteration for Out-of-Vocabulary words. In this method the dictionary-based technique is combined with a transliteration scheme in to a pageRank algorithm. We report their work as one of the baselines. Herbert et al. (2011) uses Wikipedia concepts along with Google Translate to translate the queries. By mining the cross-lingual links from the Wikipedia articles, a translation table is built. This is now coupled with translations from Google. Franco-Salvador et. al. (2014) leverages BabelNet, a multilingual semantic network for CLIR. Hosseinzadeh Vahid et al. (2015) uses Google and Bing to translate the queries and shows how the performances vary with translations from two different online systems.

Bhattacharya et. al (2016) uses word embeddings for Cross-Language Information Retrieval, learning word vectors from the document set. They also propose methods such that the query can be represented by a vector. We present their work as a baseline. Discriminative projection approaches for documents have also been applied to CLIR using Oriented Principal Component Analysis (OPCA), Coupled Probabilistic Latent Semantic Analysis (CPLSA) (Platt et al., 2010) and learning by Siamese Neural Network (S2Net) (Yih et al., 2011). Vulić and Moens (2015) uses word embeddings for CLIR. They collect document-aligned corpora and randomly merge and shuffle the pairs and feed them to the Skip-Gram architecture of word2vec. This way, they obtain cross-lingual word vectors, which they combine to obtain query vectors and document vectors. They perform IR by computing the cosine similarity between the query and the document vectors and ranking the documents according to the similarity.

3 Proposed Framework

We follow the query translation based approach towards Cross-Language Information Retrieval from Hindi to English and Bengali to English. We propose an approach for query translation using multilingual word clusters obtained from word embeddings.

Word embeddings serve as a potential tool for translation by bridging the gap between good quality translations and scarcity of data-aligned resources, like sentence-aligned parallel corpora and bilingual or multilingual dictionaries. Given a training corpus, word embeddings are able to generalize well over words that occur less frequently as well. Many words in Indian languages have been borrowed from English and have been added to the vocabulary, without any English translations like "*kaiMsara*" (meaning, Cancer, a disease). If a dictionary-based query translation is used for translating such terms, there is a high probability that the translations of such words shall be missing. Word embeddings on the other hand provide relevant translations like "cancer","disease","leukemia" for "*kaiMsara*".

To obtain multilingual word embeddings for the languages such that words that are similar across these languages have similar word vector representations, we use two state-of-the-art techniques to obtain these embeddings. The first approach is based on (Mikolov et al., 2013a) and (Mikolov et al., 2013b). The second approach is based on the idea of (Vulić and Moens, 2015). We use these methods since they use comparable and document-aligned corpora respectively, which are not very difficult to obtain. As described earlier, embedding methods requiring parallel corpora are difficult to get in resource-scarce languages. We describe the methods in Section 3.3.

In spite of multilingual embeddings being a powerful tool, translations obtained directly (by picking the top k target language words that have the highest cosine similarity with the source word) are sometimes irrelevant to the source language word. For instance, for the Hindi word "*pheMkanaa*" (meaning, throw) besides giving the correct translation "throw", the method also came up with not-so-relevant translations like "wash" and "splashing". In such situations, the performance of the CLIR system is

greatly harmed. To deal with such scenarios, we propose the use of clustering. Multilingual clustering groups together similar words across languages that share the same concept. After the multilingual word embeddings have been obtained, we construct a graph $G = (V, E)$ from the word embeddings. V, set of vertices, represents words from the languages and E, set of edges, is formed if the cosine similarity between any two words (or vertices) is above a particular threshold and if so, then the weight of the edge is the cosine similarity value.

After such a graph has been constructed, we employ Louvain (Blondel et al., 2008), an efficient community-detection algorithm that runs in $O(nlogn)$ time. Applying Louvain on the above graph outputs clusters that contain words from all the languages that represent the same concept. More details on graph and cluster formation are provided in Section 3.4.

On clustering, words across languages that represent a certain concept will form dense clusters and edges representing a high cosine similarity value but an irrelevant translation will get overshadowed. Hence, the cluster containing "*pheMkanaa*" has similar and more related words like "hurl" and "dart" instead of "wash" and "splashing", which are now in a different cluster. These clusters are now used for the purpose of query translation for CLIR as described in Section 3.5.

3.1 Dataset

For obtaining multilingual word embeddings, we use two different approaches requiring two kinds of corpora: one approach requires comparable monolingual corpora for each of the three languages (English, Bengali and Hindi) and dictionaries containing Hindi-English and Bengali-English translations. The other approach requires document-aligned corpora for the three languages. The dataset details are as follows :

- Comparable Corpora : We have used FIRE (Forum for Information Retrieval Evaluation, developed as a South-Asian counterpart of CLEF, TREC, NTCIR) 2012 dataset [1]. The documents were obtained from the newspapers, 'The Telegraph' and 'BDNews24' for English; 'Amar Ujala' and 'Navbharat Times' for Hindi; 'Anandabazar Patrika' and 'BDNews24' for Bengali. There were 1,427,986 English; 1,164,526 Hindi and 500,122 Bengali documents.

- Document-Aligned Corpora: We have used the Wikipedia dumps[2] available for download for each of the three languages, English, Bengali and Hindi. In order to get the cross-lingual articles, we made use of the inter-wiki links that exist in the corresponding Wikipedia pages. There were 55,949 English-Hindi pages; 34,234 English-Bengali pages and 12,324 English-Bengali-Hindi pages.

- Cross-Language Information Retrieval: We used the FIRE 2012 queries for Hindi and Bengali for Hindi to English and Bengali to English CLIR. There were 50 queries with topics numbered from 176-225. We used the title fields for querying.

- Other resources: We used a Hindi-English dictionary[3] that had 26,485 translation-pairs, Bengali-English dictionary[4] containing 29,890 translation-pairs, Stopword lists[5] and an English Named-Entity Recognizer[6]. Louvain Method for community detection algorithm (Blondel et al., 2008) was used for clustering.

3.2 Pre-processing the Dataset

We perform the basic pre-processing tasks on the documents, like removing the html tags, sentence boundaries and reducing all the letters to lowercase (for English). We obtain word vectors from this document set. We count the term frequencies of the words and remove stopwords, top 50 most frequently occurring words and words below frequency of 20 (for Wikipedia dataset), 50 (for English and Bengali

[1] http://fire.irsi.res.in/fire/data
[2] https://dumps.wikimedia.org/backup-index.html
[3] http://ltrc.iiit.ac.in/onlineServices/Dictionaries/Dict_Frame.html
[4] http://www.cfilt.iitb.ac.in/Downloads.html
[5] http://www.ranks.nl/stopwords
[6] http://nlp.stanford.edu/software/CRF-NER.shtml

Table 1: Statistics of the number of words (vertices) used to create word clusters, separately for FIRE and Wikipedia Datasets

	Pair				**Multi**		
	English-Hindi		**English-Bengali**		**English-Hindi-Bengali**		
	English	Hindi	English	Bengali	English	Hindi	Bengali
FIRE	129,688	84,773	129,688	93,057	129,688	84,773	93,057
Wikipedia	106,746	35,361	77,302	24,794	50,620	16,534	13,490

Table 2: Statistics of the Bilingual and Multilingual Clusters

		# Levels		**# Clusters**	
		English-Hindi	**English-Bengali**	**English-Hindi**	**English-Bengali**
Pair	**FIRE**	5	5	403	384
	Wiki	4	4	19611	20627
Multi Wiki		5		406	

FIRE dataset) and 20 (for Hindi FIRE dataset). We choose these numbers so that we have balanced number of words for the three languages. We then obtain the embeddings of the remaining words.

3.3 Obtaining Multilingual Word Embeddings

For creating multilingual word clusters using an embedding based approach, we first need to obtain multilingual word vectors. Multilingual word vectors can be obtained from parallel corpora (Gouws et al., 2015), document-aligned corpora (Vulić and Moens, 2015) and comparable corpora using a dictionary (Mikolov et al., 2013b). Since, Hindi and Bengali are resource-scarce languages, parallel, sentence-aligned data are scarce and insufficient to train word vector models. Hence, we use the methods involving document-aligned corpora and comparable corpora using a dictionary for obtaining word embeddings and test their performance. We describe these two methods next.

3.3.1 Dictionary Projection based Approach using Comparable Corpora

We obtain monolingual word embeddings separately for English, Hindi and Bengali using the *word2vec* (Mikolov et al., 2013a) tool available for download [7]. We use the Continuous Bag-of-Words (CBOW) variant to learn the monolingual word embeddings, as it has been shown to work faster than Skip-Gram for large datasets.

For learning the projection function from the source languages (Hindi and Bengali) to the target language (English), we use the linear regression method similar to (Mikolov et al., 2013b). The idea is as follows: given a dictionary of translation word-pairs $\{x_i, y_i\}$ whose monolingual word vectors $x_i \in \mathbb{R}^{\mathrm{d}_1}$ – a d_1- dimensional embedding, $\mathrm{y}_i \in \mathbb{R}^{\mathrm{d}_2}$ – a d_2- dimensional embedding, are known, the objective is to learn a translation matrix W such that the root mean square error between Wx_i and y_i is minimized. Once W has been learnt, it can now be used to project the entire vocabulary of the source language to the English space. The vectors of the words from all the three languages are now in a common vector space and can be used for translation.

3.3.2 Learning Embeddings together in a Joint Space using Document-Aligned Corpora

Vulić and Moens (2015) uses document-aligned corpora to learn bilingual embeddings. We use this approach and extend it for obtaining multilingual embeddings together in a joint space.

Let $D = \{(d_{s_1}, d_{t_1}), (d_{s_2}, d_{t_2}), \ldots, (d_{s_n}, d_{t_n})\}$ be the set of document-aligned, comparable corpora where (d_{s_i}, d_{t_i}) denotes a pair of aligned documents in source language s and target language t and n is the number of such aligned-document pairs constituting the corpus. In order to learn bilingual word embeddings, the first step is to merge the two document pairs (d_{s_i}, d_{t_i}) in to a "pseudo-bilingual"

[7] https://code.google.com/p/word2vec

document and remove sentence boundaries. Next, this bilingual document is randomly shuffled and is used as training for monolingual skip-gram model of *word2vec* (Vulić and Moens, 2015).

The idea of document-aligned "pairs" can be readily extended to document-aligned "triplets", where now there are three documents $(d_{e_i}, d_{h_i}, d_{b_i})$ in three languages that are document-aligned. In this case, we merge and shuffle the i^{th} document-triplet and obtain embeddings for words from all the three languages.

3.4 Creating Graph and obtaining Clusters

After obtaining the multilingual embeddings separately by the two methods described above, we compute the cosine similarities between the word vectors. Now a graph $G = (V, E)$ is constructed, where the vertex set V represents words from both the languages and E defines the set of edges - an edge exists between two vertices if the cosine similarity value of the word embeddings of the two vertices is greater than or equal to a threshold of 0.5. The edge weights are the cosine similarity of the embeddings of the connecting vertices (words).

After the graphs have been obtained, we apply the Louvain algorithm for community detection (Blondel et al., 2008) separately for the graphs. Given a graph, Louvain looks for small clusters, optimizing the modularity in a local way. In the first pass, small communities are formed. In the subsequent passes, it combines communities from the lower level to create larger sized clusters. The iteration stops once maximum modularity is achieved. It performs hard clustering, that is, a word belongs to only one cluster. The algorithm runs pretty fast in $O(nlogn)$ time.

Table 1 shows word-count statistics that have been used as vertices to create clusters. "Pair" indicates that the words (or vertices) are from two languages while "Multi" indicates that the words (or vertices) are from three languages.

Table 2 shows the number of levels and number of clusters for each language pair on different corpora. Since, multilingual clusters using the dictionary-based approach were not used in our experiments due to poor performance, we do not report its statistics.

In lower levels, the number of clusters were more and words that should belong to the same cluster were scattered in other clusters. In the topmost level of clustering, although there were some clusters that had a large number of words and were unrelated, most of them had related words in the same cluster. On observing the bilingual and multilingual clusters closely, we find that the bilingual clusters were mostly small and contained words that were translations and/or transliterations of each other. For clusters that were large, the communities were well representative of the words. Our main focus was on multilingual clusters since the bigger objective of our work is to have an unified representation of words for Indian Languages. Following are some examples of clusters [8] :

- FIRE Hindi-English : (inflation, *mudraasphiiti*, money, *paise*, *rakama*, *dhanaraashi*, prices, cost)
- Wikipedia Multi : (*aarthika* (hi), currency, economics, *mudraasphiiti* (bn), inflation, *arthaniiti* (bn), *munaaphaa* (hi))

3.5 Query Translation from Word Clusters

After forming multilingual word clusters, we use them for the purpose of query translation in CLIR. Given a query Q = $q_1 q_2 \cdots q_n$ in Hindi or Bengali, we first find the cluster c_k to which the query word q_i belongs. We then extract all the English words from c_k and pick the top t most similar English words from the cluster c_k for the query word q_i. We repeat this step for all the query words and append them consecutively. Note that while the stopwords in the query are already filtered, the named-entities do not have the embeddings because of filtering of words below the threshold frequency. These named-entities are dealt separately, as described in the next section.

[8] All non-English words have been written in ITrans using `http://sanskritlibrary.org/transcodeText.html`
Hindi words have been abbreviated as 'hi' and Bengali words as 'bn'.

Table 3: Performance of the Baseline Approaches for Hindi to English and Bengali to English CLIR on FIRE 2012 Dataset

		Hindi to English CLIR			Bengali to English CLIR		
		MAP	**P5**	**P10**	**MAP**	**P5**	**P10**
English Monolingual		0.3218	0.56	0.522	0.3218	0.56	0.522
Bhattacharya et al. (2016)	**FIRE**	0.2802	0.436	0.392	0.2368	0.334	0.318
	Wikipedia	0.1524	0.232	0.22	0.3027	0.448	0.402
Dictionary		0.1691	0.2048	0.2048	0.134	0.165	0.132
Chinnakotla et al. (2008)		0.2236	0.3347	0.3388	0.18	0.275	0.232
Google Translate		0.3566	0.576	0.522	0.294	0.524	0.48

3.6 Transliteration of Named Entities

Although most of the named-entities are filtered out in the pre-processing stage, some words like the names of political parties, e.g., *BJP, Congress* in Hindi and Bengali have embeddings and so we obtain similar words like the names of other political parties and also words like 'government' and 'parliament' in English. During our experiments, we observed that inclusion of such terms can harm the retrieval process and so we prefer to transliterate these. Since we did not have access to any Named-Entity Recognition (NER) tool for Hindi and Bengali, we resort to a transliteration based process similar to (Chinnakotla et al., 2008; Padariya et al., 2008). For each Hindi/Bengali character, we construct a table of its possible transliterations and also apply some language specific rules. Given a Hindi/Bengali query term h, we first transliterate it using the method described above and for each word e in the list of words returned as named entities by the NER tool for English, we apply the Minimum Edit Distance algorithm to h and e. If we find an e within a range of 0 to 1.5, we treat h as a named-entity and use the transliteration with the least distance. If no such e is returned, we consider it as a non-named entity and use the cluster based approach to obtain translation.

4 Experiments

We used Apache Solr version 4.1 as the monolingual retrieval engine. The similarity score between the query and the documents is the default TF-IDF Similarity [9]. The human relevance judgments were available from FIRE. Each query had about 500 documents that were manually judged as relevant (1) or non-relevant (0). We then used the trec-eval tool [10] for finding the Mean Average Precision (MAP), Precision at 5 (P5) and Precision at 10 (P10).

4.1 Baselines

In this section we describe the baseline methods we have used to compare our proposed approach.

- **English Monolingual**: FIRE provides corresponding queries for most Indian languages and also English. This baseline uses the English queries for retrieval.
- **Bhattacharya et al. (2016)**: In this approach, once the word vector of each query term projected in the target language (v) is obtained, cosine similarity between the vector embedding of each English word and v is computed, and the 3 best translations are picked. Although they obtained best results when the query as a whole was represented as a vector but our method involves translation at the cluster level and so we do not find such a comparison suitable. Hence, we report their result on query word vectors.
- **Dictionary**: This is the dictionary-based method where the query word translations have been obtained from the dictionary. For words that contain multiple translations, we include all of them.

[9] https://lucene.apache.org/core/3_5_0/api/core/org/apache/lucene/search/Similarity.html

[10] http://trec.nist.gov/trec_eval/

Table 4: Performance of the Proposed Cluster-based Approach for Hindi to English and Bengali to English CLIR on FIRE 2012 Dataset

Datasets		Methods	Hindi to English CLIR			Bengali to English CLIR		
			MAP	P5	P10	MAP	P5	P10
Pair En-Hi / En-Ben	FIRE	Cluster	0.352	0.4503	0.427	0.3038	0.478	0.418
		Cluster+DT	0.362	0.537	0.52	0.326	0.495	0.464
		Cluster+DT +GT	**0.452**	**0.627**	**0.578**	0.342	0.534	0.49
	Wikipedia	Cluster	0.2832	0.3760	0.35	0.3233	0.468	0.43
		Cluster+DT	0.324	0.408	0.386	0.361	0.482	0.458
		Cluster+DT +GT	0.42	0.526	0.501	0.389	0.517	0.487
Multi En-Ben-Hi	Wikipedia	Cluster	0.3014	0.446	0.37	0.3557	0.476	0.418
		Cluster+DT	0.356	0.541	0.510	0.396	0.538	0.501
		Cluster+DT+GT	0.432	0.575	0.538	**0.42**	**0.56**	**0.545**

Named entities are handled as in Section 3.6. If the translation of a query word is not present in the dictionary, it is ignored.

- **(Chinnakotla et al., 2008)** : The method proposed by (Chinnakotla et al., 2008) is used as a baseline.[11].
- **Google Translate** : Translations of the Hindi query to English have been obtained by using Google Translate.

4.2 Proposed Cluster-based Approach

We have experimented with various similarity thresholds and various levels of clustering and report the best results. We experimented with the following variants of our approach :

- **Cluster**: In this method, we simply pick the top 3 (experimentally chosen) most similar English words for each query term within the cluster and append them. We proportionally assign weights to each translation of a query term according to its similarity to the query word such that the weight of all the translations of a query term add up to 1. The named-entities were assigned a weight of 1.
- **Cluster + DT**: We combine translations from the dictionary as well as from the clusters. We first take translations from the dictionary, if a translation exists. If not, we take it only from clusters. In case translations exist in both, we assign 80% weightage to the cluster translations and 20% weightage to the dictionary translations. [12]
- **Cluster + DT + GT**: In this scheme, we combine translations from Google Translate as well as with the dictionary. We assign equal weightage to Cluster words and translations from Google, 40% each, and the rest to dictionary translations.

4.3 Results

Table 3 shows the baseline results for the CLIR task for Hindi to English and Bengali to English. The results of our proposed approached are in Table 4. For Hindi to English CLIR, dictionary-projection method performs the best and the performance improves when it is combined with dictionary translations and translations from Google. This is because the dictionary for Bengali-English was not as rich as Hindi-English. For Hindi-English the number of word pair translations trained on were 8714 and for

[11](Chinnakotla et al., 2008) is an improved version of (Padariya et al., 2008)

[12]We experimented with other weightages like 70%-30%, 90%-10%, but the 80%-20% division gives the best results.

Table 5: Some example queries and their performances

Query	Gloss	Translation Method	Translation	MAP	P5	P10
poliyo unmuulana abhiyaana	Polio eradication mission	No Cluster	vaccine polio campaign campaigns	0.4	0.55	0.48
		Wiki Pair Cluster	polio vaccine eradication mission	0.6	0.7	0.6
		Wiki Multi Cluster	polio infection prevention campaign	**0.85**	**1**	**0.9**
griisa iuro kaapa 2004 jaya	2004 Greece Euro Cup victory	No Cluster	Greece 2004 euro banknotes tournament champions victory win defeat	0.5	0.7	0.6
		Wiki Pair Cluster	Greece 2004 Euro euro trophy Football teams victory win winning	0.6	0.75	0.7
		Wiki Multi Cluster	Greece 2004 Euro trophy cup champions winner	**0.9**	**1**	**0.8**

Bengali-English the number was 6012. Multilingual word clusters perform better than bilingual word clusters when the multilingual embeddings have been learnt jointy using the Wikipedia document-aligned corpora suggesting that when another language is incorporated, cluster information improves and words in the clusters are more related with each other and aligned to the semantic information exhibited by the cluster.

Multilingual and bilingual word clusters formed using Wikipedia document-aligned data perform better for Bengali to English CLIR comapared to the dictionary-based approach using FIRE data. Multilingual word clusters alone performs well when compared in terms of MAP with Google Translate and shows improvements when combined with dictionary and Google Translate. The number of documents in Bengali from the FIRE dataset were less and this may be a probable cause for its poor performance.

Table 5 shows two example queries. The first query is for Hindi to English CLIR and the second query is for Bengali to English CLIR. For the first two translation methods, no translation is available for "*unmuulana*" (meaning, eradication) but multilingual clustering suggests the word "prevention". Also, for "*poliyo*", multilingual clustering comes up with more related word "infection" rather than "vaccine" since "polio" is primarily a disease/infection and vaccination is a medication and is secondary. For the second query, the word "*Euro*" is related to sports and not economics. No Cluster method wrongly predicts the context and suggests words like 'banknotes". On the other hand, pairwise clustering understands that "cup" is related to some sports, "football" to be more specific. Multilingual clustering restricts to a shorter query and hence translates to only "trophy" and "cup".

5 Conclusion and Future Extensions

In this paper, we proposed a method to cluster semantically similar words across languages, and evaluated it for query translation in the CLIR task. Experimental results confirm that it performs better than the dictionary method, English monolingual and transliteration based approaches. When combined with the dictionary and Google Translate in a hybrid model, it achieves the best performance. In future, we plan to extend the work for other Indian languages and obtain communities containing similar concept in multiple languages.

6 Acknowledgments

The authors would like to thank the anonymous reviewers for their valuable comments. This work is supported by the project "To Develop a Scientific Rationale of IELS (Indo-European Language Systems) Applying A) Computational Linguistics & B) Cognitive Geo-Spatial Mapping Approaches" funded by the Ministry of Human Resource Development (MHRD), India and conducted in Artificial Intelligence Laboratory, Indian Institute of Technology Kharagpur.

References

Lisa Ballesteros and W. Bruce Croft. 1996. Dictionary Methods for Cross-Lingual Information Retrieval. In *Proceedings of the 7th International Conference on Database and Expert Systems Applications*, DEXA '96, pages 791–801.

Paheli Bhattacharya, Pawan Goyal, and Sudeshna Sarkar. 2016. Using word embeddings for query translation for hindi to english cross language information retrieval. *Computación y Sistemas*, 20(3):435–447.

Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast Unfolding of Communities in Large Networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008.

Sarath Chandar, Stanislas Lauly, Hugo Larochelle, Mitesh Khapra, Balaraman Ravindran, Vikas C Raykar, and Amrita Saha. 2014. An autoencoder Approach to Learning Bilingual Word Representations. In *Advances in Neural Information Processing Systems*, pages 1853–1861.

Manoj Kumar Chinnakotla, Sagar Ranadive, Om P. Damani, and Pushpak Bhattacharyya. 2008. Hindi to English and Marathi to English Cross Language Information Retrieval Evaluation. In *8th Workshop of the Cross-Language Evaluation Forum, CLEF 2007, Budapest, Hungary, September 19-21, 2007, Revised Selected Papers*, pages 111–118.

Manaal Faruqui and Chris Dyer. 2014. Improving Vector Space Word Representations Using Multilingual Correlation. In *Proceedings of EACL*.

Marc Franco-Salvador, Paolo Rosso, and Roberto Navigli. 2014. A Knowledge-based Representation for Cross-Language Document Retrieval and Categorization. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2014)*.

Stephan Gouws and Anders Søgaard. 2015. Simple task-specific bilingual word embeddings. *Proceedings of NAACL-HLT*, pages 1386–1390.

Stephan Gouws, Yoshua Bengio, and Greg Corrado. 2015. Bilbowa: Fast Bilingual Distributed Representations without Word Alignments. In *International Conference on Machine Learning (ICML)*.

Benjamin Herbert, György Szarvas, and Iryna Gurevych. 2011. Combining Query Translation Techniques to Improve Cross-language Information Retrieval. In *Proceedings of the 33rd European Conference on Advances in Information Retrieval*, ECIR'11, pages 712–715.

Ali Hosseinzadeh Vahid, Piyush Arora, Qun Liu, and Gareth J.F. Jones. 2015. A Comparative Study of Online Translation Services for Cross Language Information Retrieval. In *Proceedings of the 24th International Conference on World Wide Web*, pages 859–864.

Kejun Huang, Matt Gardner, Evangelos E. Papalexakis, Christos Faloutsos, Nikos D. Sidiropoulos, Tom M. Mitchell, Partha Pratim Talukdar, and Xiao Fu. 2015. Translation Invariant Word Embeddings. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 1084–1088. The Association for Computational Linguistics.

David A. Hull and Gregory Grefenstette. 1996. Querying across languages: A dictionary-based approach to multilingual information retrieval. In *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '96, pages 49–57.

Alexandre Klementiev, Ivan Titov, and Binod Bhattarai. 2012. Inducing Crosslingual Distributed Representations of Words. In *COLING*.

Michael L. Littman, Susan T. Dumais, and Thomas K. Landauer, 1998. *Automatic Cross-Language Information Retrieval Using Latent Semantic Indexing*, pages 51–62.

Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Bilingual Word Representations with Monolingual Quality in Mind. In *NAACL Workshop on Vector Space Modeling for NLP*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Tomas Mikolov, Quoc V. Le, and Ilya Sutskever. 2013b. Exploiting Similarities among Languages for Machine Translation. *CoRR*, abs/1309.4168.

Nilesh Padariya, Manoj Chinnakotla, Ajay Nagesh, and Om P Damani. 2008. Evaluation of Hindi to English, Marathi to English and English to Hindi CLIR at FIRE 2008. In *Working Notes of Forum for Information Retrieval and Evaluation (FIRE), 2008*.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation. In *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 '98, 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 '10, pages 251–261.

Shigehiko Schamoni, Felix Hieber, Artem Sokolov, and Stefan Riezler. 2014. Learning Translational and Knowledge-based Similarities from Relevance Rankings for Cross-Language Retrieval. In *ACL*.

Artem Sokolov, Felix Hieber, and Stefan Riezler. 2014. Learning to Translate Queries for CLIR. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '14, pages 1179–1182.

Ferhan Türe, Jimmy Lin, and Douglas W Oard. 2012a. Looking inside the box: Context-sensitive translation for cross-language information retrieval. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, pages 1105–1106. ACM.

Ferhan Türe, Jimmy J Lin, and Douglas W Oard. 2012b. Combining Statistical Translation Techniques for Cross-Language Information Retrieval. In *COLING*, pages 2685–2702.

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*, pages 363–372. ACM.

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 '11, pages 247–256.