CDMiner: Mining Cultural Differences between Equivalent Terms in English and Chinese

Yizhong Wang[1] and **Hanyuan Shi**[2] and **Kenny Q. Zhu**[3] and **Seung-won Hwang**[4]
[1,2,3] Shanghai Jiao Tong University, China
[4] Yonsei University, Korea
[1] umm@sjtu.edu.cn [2] shihanyuan@sjtu.edu.cn [3] kzhu@cs.sjtu.edu.cn
[4] seungwon.hwang@gmail.com

Abstract

Equivalent terms on the same object or entity in English and Chinese may exhibit subtle cultural differences. This paper proposes four word-embedding empowered methods to automatically discover and measure such differences. We developed a web demo that allows the user to search or browser for an English or Chinese word, and the system tells how much cultural difference is there for this word between English and Chinese and explain why by visualizing the neighboring words in the embedding space. Our method achieves a precision as high as 0.9 for the top 10 most culturally different term pairs.

1 Introduction

Entities or objects which have equivalent terms or expressions in different languages may have subtle differences in their use scenario or perceptions in the respective culture. For example, dragon and loong(龙) refer to the similar legendary reptile-like creature that can fly and exhale flames, in both English and Chinese. While the two concepts have influenced each other in the past, they have quite notable cultural differences. In the English or the larger European culture, this snake-like, winged being has a generally violent and fearsome image, which is often associated with negative contexts such as dungeon or hell. In the Chinese setting, however, loong is a symbol of royal power, majesty and auspiciousness. Sometimes, it even carries the notion of playfulness, as loong is often shown playing with a ball. In fact, these differences in perception can be observed by visual distinction in image search results. Figure 1 and Figure 2 show the top images returned from Bing image search of "dragon" and "龙"[1]. One can see that the popular images, which reflect the perception of these concepts in their respective cultures, are quite different.

Figure 1: Images of "dragon" on Bing.

Figure 2: Images of "龙" on Bing.

Understanding such cultural differences in the expressions between languages can be useful in many applications, especially when there is insufficient contextual information. In machine translation of short texts (such as movie subtitles), we want to avoid translated terms that are irrelevant or even offensive in another culture. In cross-lingual recommendation systems, we may only recommend products or services most relevant to the people of a particular culture. Finally, in foreign language education, it is useful to know that bowls are made to contain rice in China while they are typically much bigger in Europe to hold salad.

Previous efforts have focused on translating or paraphrasing concepts from one language to other languages (e.g., BabelNet (Navigli and Ponzetto,

[1] http://global.bing.com/images/.

2012) and YAGO3 (Mahdisoltani et al., 2014)). Other examples include mining synonymous word pairs across different languages(Mikolov et al., 2013a), or bilingual lexicon(Linard, 2015), for translation. In other words, they pay much attention on the similarity rather than differences between the cultures.

In this paper, we propose to develop metrics that quantatively measure the cultural differences between equivalent expressions of objects or entities in English and Chinese. Our proposed framework is language independent, such that, given suitable corpora, the same methodology developed here can be applied to other languages and cultures. Our goal is to compute a globally comparable score for any pair of English-Chinese translations, where the English term refers to an entity. We choose to restrict our study on entities, because these are traditionally thought to carry identical meaning across any culture. Furthermore, cultural difference and, for the same matter, culture similarity, for entities can be more easily verified by human judges using the image search described above. The main contributions of this paper can be summarized as follows.

- We propose four algorithms, all based on the skip-gram word embedding model trained from large literature corpora, to calculate the cultural difference between a pair of English-Chinese translated terms;
- We propose an innovative evaluation method that uses image search results to identify entity pairs which have cultural differences;
- All four algorithms above perform reasonably well, while the KNN-cosine method outperforms the rest and achieves 0.9 precision score for the top 10 culturally different term-pairs;
- We build a web-based demonstration system that allows users to search an English or a Chinese term, and returns the cultural difference score, a percentile ranking among all indexed terms, as well as a visual explaination why a pair of terms is cultural different (or indifferent).

Next we present our approaches (Section 2), show some key evaluation results (Section 4), propose a demonstration plan (Section 3), and discuss some previous related work (Section 5).

2 Approach

This section discusses the preprocessing of our text corpora, an overview of the Skip-grams model, which is the basis for capturing word semantics, as well as our four methods to compute the cultural difference score.

2.1 Data Preprocessing

We use non-parallel corpora to train word embeddings. We collect these corpora (both English and Chinese) from online electronic books including such categories as poetry, science fictions, politics, biography, fables, etc. The English corpus contains 2,988 books, totaling 1.8GB, while the Chinese corpus is made up of 25,000 books, with a combined size of 5.1GB. For English, we lemmatize the text and reduce all words to their original forms. Meanwhile, for Chinese, we apply a word segmentation tool called NLPIR (Huaping Zhang, 2016) to the text and remove all the non-Chinese terms. We also remove all punctuation charaters and those terms with frequency less than 10 for both English and Chinese to make sure every term in the corpora is meaningful.

Because our goal is to compute cultural difference score between a pair of equivalent English and Chinese terms, we need to prepare such term pairs in advance. First, we get the all English entities with a name that is a single word from WordNet as our candidate English terms. Then we filter out words that are highly ambigious from the above list. To do that, we search each word on English Wikipedia and keep those words that have only one dominant Wikipedia page, that is, when such a word is searched, Wikipedia directly takes you to its corresponding article, instead of a disambiguation page. After that we translate English words into Chinese by Youdao online dictionary [2], and select the first noun translation as the equivalent Chinese term.

2.2 The Skip-gram Model

We use Skip-gram model to train word embedding from English and Chinese corpus separately. At this step, two vector representations are generated for every word pair, one for English and the other for Chinese.

In the Skip-gram model, the objective is to max-

[2] http://dict.youdao.com.

imize the average log probability

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{j=-c}^{c}\log p(w_{t+j}|w_t)$$

where c is the size of training window, and T is the total size of training corpus.

$p(w_i|w_j)$ is defined by a softmax function

$$p(w_i|w_j) = \frac{\exp(u_{w_i}^{\top} v_{w_j})}{\sum_{l=1}^{V}\exp(u_l^{\top} v_{w_j})}$$

where V is the size of the vocabulary, u_w and v_w are the "input" and "output" vectors representing the word w. In our experiments, we set the window size c as 10 and the size of "input" and "output" vectors equals to 100.

After we train this model on each monolingual corpus, vector representations are generated for every word appearing in our vocabulary and we have two vector spaces: one for English corpus and the other for Chinese corpus.

2.3 Similarity Calculation

In this part, we will introduce four similarity calculation methods. The inverse of similarity between the vector representations for corresponding English and Chinese gives the amount of cultural difference for a pair.

2.3.1 Linear-transformation Algorithm

English and Chinese vector spaces trained from the Skip-gram model are not directly comparable due to unknown meaning of each dimension. However, experiments (Mikolov et al., 2013a) have shown that the relationship between these vector spaces can be possibly captured by rotation and scaling, represented by a linear transformation matrix W. This matrix can be learned using a number of words with *little* cultural difference and the following optimization problem:

$$\arg\min_{W}\sum_{i=1}^{n}\|Wx_i - t_i\|^2$$

where x_i is a word in Chinese while t_i is its corresponding translation in English and n is the size of training samples.

Thus we train a linear transformation matrix from Chinese to English spaces and map each Chinese word vector to the English space. Finally we calculate the cosine similarity between the two vectors in the English space.

2.3.2 KNN-set Algorithm

For each pair of English and Chinese terms, we find k nearest neighbors for both the Chinese and English word in their respective embedding space. We use two distance metrics, namely cosine similarity and Euclidean distance, to find k nearst neighbors. In our experiment, we tune k to be 100. Once we have these neighbors, we can use this set of terms to represent the orginal English or Chinese term. Since we can translate between Chinese and English terms, we can map the words in the Chinese set to English and calculate Jaccard similarity between this translated set and original English set as follows:

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

where A and B are two sets.

2.3.3 Distance-vector Algorithm

For English, we calculate the distance by cosine similarity from a term vector to all other terms in word embedding space, in order to get an n-dimensional vector, where n is the total number of terms in our vocabulary. We call this n-dimensional vector *distance vector*, and the i^{th} dimension for the distance vector represents the distance to word w_i in the vocabulary. We do the same for all terms in the Chinese space. Since each dimension in this new representation corresponds to a word in the vocabulary and each word has its mapping to another language, these distance vectors are comparable in two different spaces. Hence we can calculate the cosine similarity of all pairs using the distance vectors.

3 CDMiner Demonstration

We demonstrate CDMiner by building an online search engine[3] that can measure and visualize culture differences between equivalent terms in English and Chinese. We use cosine-knn method, which achieves the best performance in evaluation part, as our background technique to support CDMiner.

The front page shows a search box, a cultural difference scale, a series of sample pictures and two nearest neighbors graphs from English and Chinese cultures. User can enter either an English or a Chinese term in the search box. Fig-

[3] http://adapt.seiee.sjtu.edu.cn/cdminer

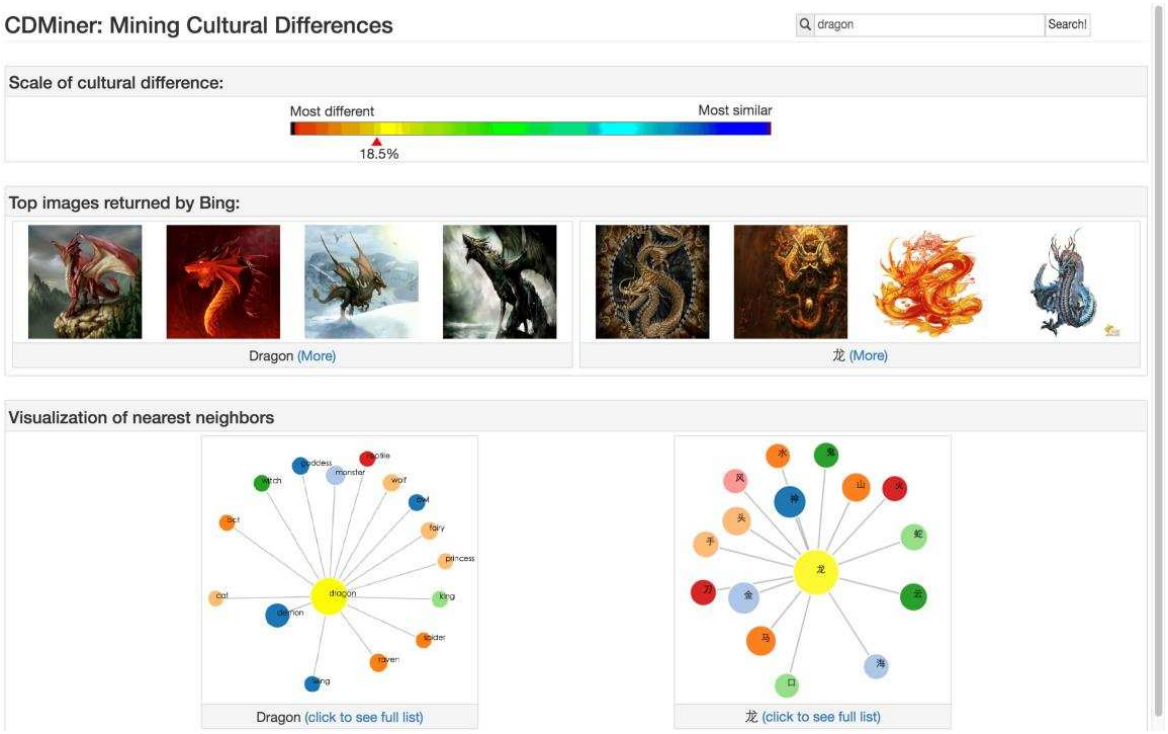


Figure 3: Result Page for "dragon" from CDminer

ure 3 shows the result page returned by searching "dragon". Below the search box, CDMiner shows a spectrum that indicates where the search term stands among all terms by the amount of cultural difference it has between English and Chinese. CDMiner also gives top 4 English and Chinese pictures according to Bing image search. In order to get a better view, "More" button links to the Bing image search to show you more pictures. The nearest neighbors lists top 15 related terms in both language using a 2-d visualization (Figure 4). Each circle links to the word's own result page. Meanwhile, the full list of related words is visible when you click on "click to see full list" button.

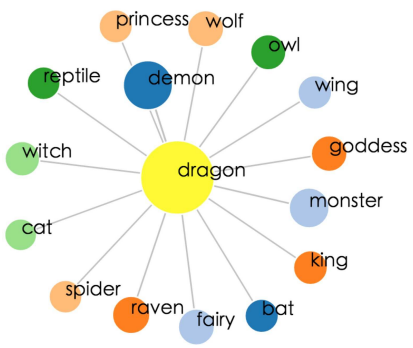


Figure 4: Nearest Neighbors Graph for "dragon"

4 Evaluation

Once we obtain the cultural difference score of each English-Chinese term pair, we are able to sort them by decreasing order. Our previous example of "dragon" and "loong" ranks 369 or top 18.5% among 2000 labeled pairs. Table 1 shows the top 10 culturally different pairs discovered by cosine-knn method. To understand why these terms have cultural differences, take "amusement" and "娱乐" for an example. Table 2 shows their top 5 nearest neighbors and we can find that "amusement" in English is more inclined to people's feeling, while "娱乐" in Chinese is usually related to the entertainment industry. These results do show substantial differences in the context of their use.

To determine the ground truth that a pair of terms is culturally different or not, we get help from Bing image search. We believe that if an entity or object has cultural differences, such differences should be reflected in people's general perception or images of the object, which are captured by large commercial search engines such as Bing. This is evident from the images of dragon and loong in Figure 1 and Figure 2. In this section, we will discuss how we create the evaluation data set with Bing search and present the results.

	English	Chinese		English	Chinese
1	**amusement**	**娱乐**	6	**blade**	**叶片**
2	**cross**	**十字架**	7	**raven**	**掠夺**
3	**relief**	**浮雕**	8	**regiment**	**团**
4	**pitcher**	**投手**	9	**content**	**内容**
5	citation	引用	10	**review**	**回顾**

Table 1: Top 10 culturally different words by cosine-knn (bold for culturally different pairs)

	amusement	**娱乐**
1	pleasure	投资 (investment)
2	sympathy	商业 (business)
3	humor	购物 (shopping)
4	pride	广告 (advertising)
5	enthusiasm	产业 (industry)

Table 2: 5 nearest neighbors of "amusement" and "娱乐" by cosine-knn

4.1 Evaluation data

We select the most common 2000 English terms (from our corpus) along with their Chinese equivalents to form an evaluation data set. Human annotators are asked to search the English and Chinese word in Bing image search engine respectively and judge whether the results are visually different. For example, if you search the term "dragon" in English and "loong" in Chinese, you will find quite different images from both queries. However, a search for "computer" and "电脑" yields very similar images.

4.2 Results and Discussion

As usual, we use precision, recall and F1 score to evaluate the top-k retrieval performance, as is shown in Figure 5 . We find that all of the four algorithms outperform the random baseline, which means our methodology captures cultural differences to varying degrees. Furthermore, cosine-knn algorithm performs better than other methods for term pairs up to top 500, while the highest precision of 0.9 is achieved for top 10 pairs.

To give an overall evaluation of our result, we also calculate the average precision of our four algorithms as follows:

$$\text{AvePrecision} = \frac{\sum_{k=1}^{n}(P(k) \times \text{rel}(k))}{|\text{culturally different pairs}|}$$

where $P(k)$ represents the precision score at top k and $\text{rel}(k)$ is an indicator function equaling 1 if the item at rank k is culturally different, zero otherwise. Table 3 shows the average precision scores of our four algorithms and we can see the cosine-knn method has a notable advantage over the other three. We think the reason why cosine-knn performs better is that it doesn't have any assumption for the relationship of different embedding spaces, which is the basics of linear transformation method. Meanwhile, cosine-knn is expressive enough as we tune the parameter k and a tuned model is able to filter out the noises that bring down distance-vector method.

	AvePrecision
linear-trans	0.606
cosine-knn	**0.617**
euclidean-knn	0.579
distance-vector	0.540

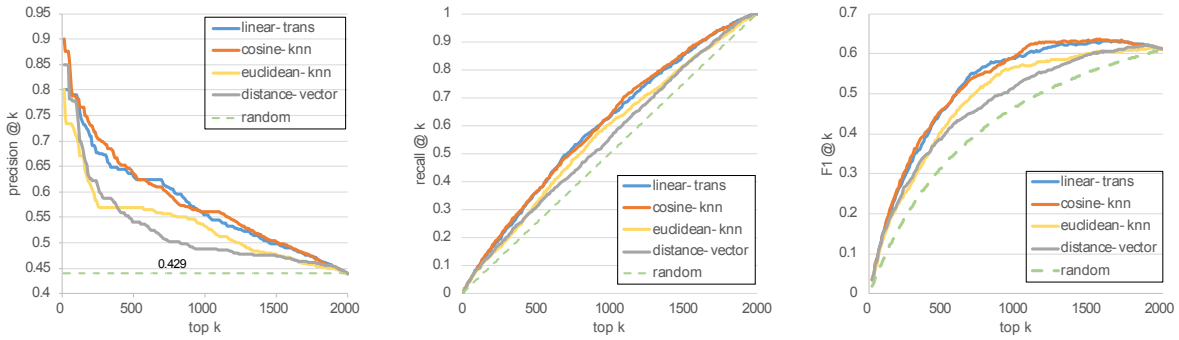
Table 3: Average precision of 4 similarity calculation method

5 Related Work

Cognitive linguistic studies (Kovecses, 2006) have shown that equivalent terms in different languages may have very different meanings. This phenomenon proves to hold between English and Chinese (Chen, 2007; Tavassoli, 1999; Krifka, 1995). In computational linguistics, discovery of relationships across languages is an emerging topic. There are generally two research directions: graph-based knowledge network or distribution-based vector representation.

BabelNet(Navigli and Ponzetto, 2012) and Yago3(Mahdisoltani et al., 2014) are representatives of graph-based knowledge network, with an ambition to construct a unified multilingual knowledge base (just like WordNet). They integrate resources such as WordNet and Wikipedia to achieve this goal. The knowledge base thus built can be used to calculate relatedness across languages. However, both of them rely on existing structured resources to create the networks, which limit their scale and extendability.

For distributional models, the predominant approach to represent the semantics of words is word embedding. The embeddings are usually trained using co-occurrence matrix, matrix factorization(Lebret and Collobert, 2013; Levy and Goldberg, 2014; Li et al., 2015) or neural network(Mikolov et al., 2013b). Traditionally, these vectors are trained on monolingual data and the vector spaces of different languages are not di-



(a) Precision curve (b) Recall curve (c) F1 curve

Figure 5: Precision, recall and F1-score for top k pairs

rectly comparable with each other. To solve this, some researchers try to train unified representations from multilingual corpus (Klementiev et al., 2012; Hermann and Blunsom, 2014; Vulic and Moens, 2015) or construct a mapping between the vector spaces of different languages (Mikolov et al., 2013a). These vectors are then evaluated in tasks such as bilingual lexicon induction or cross-lingual word sense disambiguation, and have shown to achieve state-of-art performance.

Our task is similar to bilingual lexicon induction, though we want to detect semantic difference instead of finding similar words. Tomas Mikolov(Mikolov et al., 2013a) shows the potential to detect errors in bilingual dictionary with Word2Vec and linear transformation among different vector spaces. In this paper, we implement their idea (linear-trans) and compare with several ideas of ours.

References

Jenn-Yeu Chen. 2007. Do chinese and english speakers think about time differently? failure of replicating boroditsky (2001). *Cognition*, 104(2):427–436.

Karl Moritz Hermann and Phil Blunsom. 2014. Multilingual models for compositional distributed semantics. *arXiv preprint arXiv:1404.4641*.

Huaping Zhang. 2016. NLPIR chinese segmentor system.

A Klementiev, I Titov, and B Bhattarai. 2012. Inducing crosslingual distributed representations of words.

Zoltan Kovecses. 2006. Language, mind, and culture.

Manfred Krifka. 1995. 11 common nouns: A contrastive analysis of chinese and english. *The generic book*, page 398.

Rémi Lebret and Ronan Collobert. 2013. Word emdeddings through hellinger pca. *arXiv preprint arXiv:1312.5542*.

Omer Levy and Yoav Goldberg. 2014. Neural word embedding as implicit matrix factorization. In *Advances in Neural Information Processing Systems*, pages 2177–2185.

Yitan Li, Linli Xu, Fei Tian, Liang Jiang, Xiaowei Zhong, and Enhong Chen. 2015. Word embedding revisited: A new representation learning and explicit matrix factorization perspective. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence*, pages 3650–3656.

Alexis Linard. 2015. Bilingual lexicon extraction.

Farzaneh Mahdisoltani, Joanna Biega, and Fabian Suchanek. 2014. Yago3: A knowledge base from multilingual wikipedias. In *7th Biennial Conference on Innovative Data Systems Research*.

Tomas Mikolov, Quoc V Le, and Ilya Sutskever. 2013a. Exploiting Similarities among Languages for Machine Translation. *arXiv.org*, September.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Roberto Navigli and Simone Paolo Ponzetto. 2012. BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artif. Intell.*, 193:217–250.

Nader T Tavassoli. 1999. Temporal and associative memory in chinese and english. *Journal of Consumer Research*, 26(2):170–181.

Ivan Vulic and Marie-Francine Moens. 2015. Bilingual word embeddings from non-parallel document-aligned data applied to bilingual lexicon induction. pages 719–725, July.