Supplementary Material for "Understanding Slang Terms across Languages": Computing Cross-cultural Similarities and Differences

Bill Y. Lin*, **Frank F. Xu and Kenny Q. Zhu**

{yuchenlin,frankxu}@sjtu.edu.cn,kzhu@cs.sjtu.edu.cn

Department of Computer Science and Engineering

Shanghai Jiao Tong University

800 Dongchuan Road, Shanghai, China 200240

Abstract

In this supplementary material, we generalize our work in the student abstract "Understanding Slang Terms across Languages" in a broader context of capturing cross-cultural differences. We propose one more task as the application of our proposed *SocVec*. We also present our models with more detail as well as more experiments.

Capturing cross-cultural differences between terms is an important challenge in bilingual text understanding and machine translation. This paper presents a novel framework for obtaining bilingual word representations from social media by leveraging "social vocabularies". Such representations can act as a building block for cross-lingual and cross-cultural studies in computational social science. We evaluate our framework on two tasks: 1) detection of cross-cultural differences of named entities in social media and 2) to explain slang terms in one language with terms in another language. We also release two new datasets for these tasks. Experimental results show that simple methods based on our proposed word representations outperform a number of strong baseline methods by substantial margins.

Introduction

Computing similarities between terms is one of the most fundamental computational tasks in natural language understanding. Much work has been done in this area, most notably using the distributional properties drawn from large monolingual textual corpora to train vectorial representations of words or other linguistic units (Bengio et al. 2000; Pennington, Socher, and Manning 2014; Le and Mikolov 2014). Recently there is growing interest in cross-lingual and cross-cultural similarity computation (Luong, Pham, and Manning 2015; Garimella, Mihalcea, and Pennebaker 2016; Camacho-Collados et al. 2017). In this paper, we consider interesting questions like these:

1. *Is there any cross-cultural difference between Nagoya (a city in Japan) for native English speakers and 名古屋 (Nagoya in Chinese) for Chinese people?*
2. *What are the English terms that can explain the meaning of "浮云" (a Chinese Internet slang term)?*

Such questions are important in cross-cultural social studies and machine translation systems for producing culturally sensitive results. We can generalize these two questions into two cross-cultural and cross-lingual tasks.

The first task is mining cross-cultural differences in the perception of named entities (e.g., persons, places and organizations). Opinions about named entities can be very different from culture to culture. Back in 2012, in the case of "Nagoya", while most native English speakers considered the city to be a nice travel destination, Chinese people overwhelmingly greeted the city with anger and condemnation because the city mayor denied the truthfulness of the Nanjing Massacre in 1937. Enabling machines to understand such cross-cultural differences toward named entities can be useful in various cross-lingual language processing tasks and human-computer interactions.

The second task is to explain slang terms across languages. Social media is a rich soil to produce novel slang terms in all cultures. For example, "浮云" literally means "floating clouds", but now it almost always means "nothingness" on the Internet; "天朝" literally means "Heavenly Kingdom", now becomes a common nickname for the Chinese government. Experiments show that state-of-the-art machine translators often translate such slang terms to their literal meanings, even under a clear context where slang meanings are much more appropriate. Simply put, given a slang term in one language, this task is to find some terms in another language which can help explain its meaning.

Both of the two tasks share a core problem, which is *how to compute cross-cultural differences (or similarities) between two terms from different languages*. [1]

Existing bilingual word representation models require bilingual supervisions: aligned parallel corpora with alignments, bilingual lexicons or comparable documents (upa 2016). However, aligned parallel corpora from social media are usually expensive to obtain and do not scale. Plus, most models do not purposely preserve features reflecting social and cultural contexts in different cultures.

In this paper, we propose a novel approach to project two heterogeneous monolingual word vector spaces into one bilingual word vector space, known as social vector space

*Phone: +86 13120889217; URL: https://yuchenlin.github.io

[1] A term here refers to either an ordinary word, an entity name or a slang term.

(*SocVec*), by constructing a bilingual social lexicon (BSL), which contains a set of bilingual mappings of selected words that reflect opinion, sentiment, cognition and many other psychological processes. These words, we believe, are central to capturing cultural differences (Garimella, Mihalcea, and Pennebaker 2016). Each dimension of our *SocVec* then represents a feature derived from mono-lingual semantic similarities between an input term and each entry of the BSL. Consequently, a term W in language L_1 and a term U in language L_2 can each be respectively projected into a bilingual word vector in the *SocVec* while maintaining their social features. Thus the cross-cultural similarity between W and U can be computed directly with the two vectors from the *SocVec*. Note that *ScoVec* is an encoding for words across languages, where each dimension has clear meaning. Such encoding methods have been shown to be beneficial in cross-lingual transformation and multilingual tasks (Berend 2017; Smith et al. 2017).

In summary, this paper makes the following contributions:

1. We propose a direct and uncomplicated bilingual word representation model as a building block for cross-cultural social studies with low-cost bilingual resources.
2. We propose two novel and important tasks in computational social science and evaluate our model on both of them. Experimental results show that our model outperforms strong baseline methods by significant margins.
3. We open-source a prototype tool for building *SocVec* and release two valuable datasets on the above tasks as well as a bilingual socio-linguistic lexicon, which will benefit future research in this area.

Proposed *SocVec* Model

We first discuss the intuition behind our model informally, then give the overall workflow of our approach, and finally present the details of the *SocVec* framework.

Problem Statement and the Intuition

Assuming the language L_1 is English and L_2 is Chinese,[2] the problem we address is: given an English term W and a Chinese term U, compute a cross-lingual similarity score, $clsim(W, U)$, representing the cross-cultural similarity between W and U. Although we can easily train English and Chinese word embeddings respectively, we cannot directly calculate the similarity between the mono-lingual word vectors of W and U since they are trained separately and thus their dimensions are inconsistent. Consequently, we have to devise a reasonable way to calculate the similarity across two different vector spaces while retaining their respective socio-linguistic features at the same time. That is the main challenge of the the problem.

A very intuitive solution to the problem is to translate U to its English counterpart U' through a Chinese-English bilingual lexicon and then simply consider the cosine similarity between W and U' by their English word embeddings. However, this solution is infeasible for the two tasks for three reasons: i) if U is an OOV (Out of Vocabulary) term, e.g., a slang term, then there is no U' in the bilingual lexicon; ii) if W and U refer to the same named entity, $U' = W$, then $clsim(W, U)$ is just the similarity between W and itself, therefore we cannot capture any cross-lingual differences. iii) Besides, this approach does not purposely preserve the cultural and social context of the terms. Therefore, this kind of solutions are not suitable for the two aforementioned tasks, which require the cross-cultural similarities between slang terms and differences between entity names.

To overcome the above problems, our intuition is thus to project English and Chinese word vectors to a common third space, known as *SocVec* and this projection is supposed to carry social and cultural context such as opinions, sentiments and cognition associated with the terms in respective languages. Such information is suppose to be encoded as values on each dimension of *SocVec*.

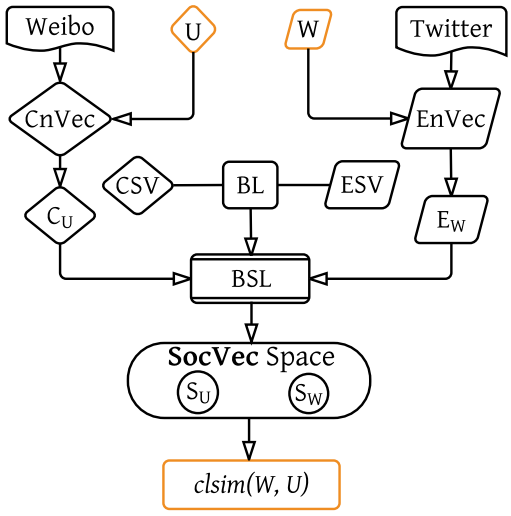

Figure 1: Workflow for computing the cross-cultural similarity between an English word *W* and a Chinese word *U*.

Overall Workflow of Building *SocVec*

Our proposed *SocVec* model attacks the problem with the help of three low-cost external resources: i) English and Chinese social media corpora; ii) an English-Chinese bilingual lexicon (*BL*); iii) English and Chinese social word vocabularies (*ESV* and *CSV*). Examples of social words in English include *fawn, inept, tremendous, gratitude, terror, terrific, kiss, loving, traumatic*, etc.

Figure 1 shows the overall workflow of our framework to construct the *SocVec* and compute $clsim(W, U)$. Notations: *CnVec* = Chinese word vector space, *EnVec* = English word vector space, *CSV* = Chinese social word vocab, *ESV* = English social word vocab, *BL* = Bilingual Lexicon, *BSL* = Bilingual Social Lexicon. Finally, E_x, C_x and S_x denote the word vectors of word x (either U or W) in *EnVec* space, *CnVec* space and *SocVec* space respectively.

First, we train English and Chinese word embeddings (*EnVec* and *CnVec*) on the English and Chinese social media

[2]Due to the salient cross-cultural differences between the east and the west, this paper chooses English and Chinese as the example language pair. Nevertheless, the techniques developed here are language independent and thus can be used for any two natural languages so long as we have the necessary resources.

corpora respectively. Then, we build a *BSL* from the CSV and ESV as well as *BL*, . The *BSL* helps us map previously incompatible *EnVec* and *CnVec* into the common higher-dimensional *SocVec* space, where two new vector representations, S_W for W and S_U for U, are now comparable.

SocVec Modeling

In this section, we present the details of building the *BSL* and constructing the *SocVec* space.

Building the *BSL* The process of building the *BSL* is illustrated in Figure 2. We first utilize the bilingual lexicon to translate each social word in the *ESV* into Chinese words and then filter out all the words that are not in the *CSV*. After that, we have a set of Chinese social words for each English social word. We call this set of Chinese words the "translation set". The final step is to generate a Chinese "pseudo-word" for each English social word.[3]

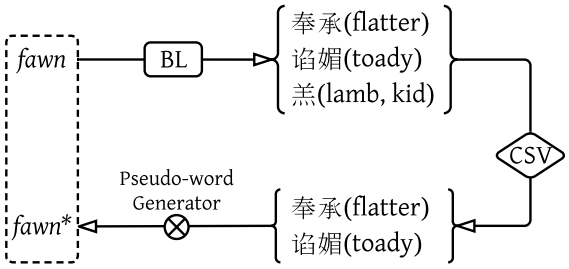

Figure 2: Generating an entry in BSL for "*fawn*" and its pseudo-word "*fawn**".

For example, in Figure 2, the English social word "*fawn*" has three Chinese translations in the bilingual lexicon, but only two of them are social words. The pseudo-word generator takes the word vectors of the two words, namely 奉承(flatter) and 谄媚(toady), as input, and generates the pseudo-word vector of "*fawn*", denoted as "*fawn**".

Given an English social word s, we denote $\mathbf{C_i}$ as word vector of the i^{th} word of the translation set (consisting of N words). We design four intuitive types of pseudo-word generator as follows:

Max.: Maximum of the values in each dimension, assuming K-dimensional word vectors.

$$Pseudo(\mathbf{C_1}, ..., \mathbf{C_N}) = \begin{bmatrix} max(C_1^{(1)}, ..., C_N^{(1)}) \\ \vdots \\ max(C_1^{(K)}, ..., C_N^{(K)}) \end{bmatrix}^{\mathrm{T}}$$

Avg.: Average of the values in each dimension.

$$Pseudo(\mathbf{C_1}, ..., \mathbf{C_N}) = \frac{1}{N}\sum_i^N \mathbf{C_i}$$

[3]A pseudo-word can be either an existing word that is the most representative word of the translation set or a fabricated word whose word vector is the combination of word vectors of the translation set.

WAvg.: Weighted average value of each dimension with respect to the translation confidence.

$$Pseudo(\mathbf{C_1}, ..., \mathbf{C_N}) = \frac{1}{N}\sum_i^N w_i\mathbf{C_i}$$

Top: Choose the most confident translation, $\mathbf{C_{top}}$.

$$Pseudo(\mathbf{C_1}, ..., \mathbf{C_N}) = \mathbf{C_{top}}$$

The direction of building *BSL* can also be from Chinese to English, in the same manner. One can also remove this asymetry by averaging the results from both directions. However, in practice, we find that the current English to Chinese direction gives better results due to the better quality of English to Chinese translation in our BL.

At the end of this step, the *BSL* contains a set of English-Chinese word vector pairs, each entry representing an English social word and its Chinese pseudo-word.

Constructing the *SocVec* Space We denote $\mathbf{E_x}$, $\mathbf{C_x}$ and $\mathbf{S_x}$ as the word vectors of word x in *EnVec*, *CnVec* and *SocVec* spaces respectively. Let B_i be an English word and B_i^* be the corresponding Chinese pseudo-word for the ith entry of *BSL*. We can project an English word vector $\mathbf{E_W}$ into *SocVec* by computing the cosine similarity between $\mathbf{E_W}$ and each English word vector in *BSL*, effectively constructing a new vector $\mathbf{S_W}$ of size L. We also map a Chinese word vector $\mathbf{C_U}$ to be a new vector $\mathbf{S_U}$. Now $\mathbf{S_W}$ and $\mathbf{S_U}$ belong to the same vector space *SocVec* and are comparable.

For example, if W is "Nagoya" and U is "名古屋", we compute the cosine similarities between "Nagoya" and each English social word in *BSL*. Such similarities compose $\mathbf{S}_{\text{nagoya}}$. Similarly, we compute the cosine similarities between "名古屋" and each Chinese pseudo-words and form the social word vector $\mathbf{S}_{名古屋}$.

Formally, $clsim(W, U)$ is computed as:

$$\begin{aligned} clsim(W, U) &:= f(\mathbf{E}_{\mathrm{W}}, \mathbf{C}_{\mathrm{U}}) \\ &= sim\left(\begin{bmatrix} cos(\mathbf{E}_{\mathrm{W}}, \mathbf{E_{B_1}}) \\ \vdots \\ cos(\mathbf{E}_{\mathrm{W}}, \mathbf{E_{B_L}}) \end{bmatrix}^{\mathrm{T}}, \begin{bmatrix} cos(\mathbf{C}_{\mathrm{U}}, \mathbf{C_{B_1^*}}) \\ \vdots \\ cos(\mathbf{C}_{\mathrm{U}}, \mathbf{C_{B_L^*}}) \end{bmatrix}^{\mathrm{T}}\right) \\ &= sim(\mathbf{S}_{\mathrm{W}}, \mathbf{S}_{\mathrm{U}}), \end{aligned}$$

where cos denotes the cosine similarity. The function sim is a generic similarity function, for which a number of metrics will be considered later in experiments.

Proposed Tasks and Evaluation

In this section, we evaluate our framework on two tasks requiring cross-cultural differences/similarities computation: i) mining cross-cultural differences in named entities, ii) finding the most similar words for slang terms across languages. Following subsections first discuss some preliminary setup, then present our experiments for the two tasks.

Experiment Setup

Prior to evaluating *SocVec* with our two proposed tasks, we need to first pre-process the social media copora, perform

entity linking, then train mono-lingual word embeddings, and finally collect the bilingual lexicon for common words and social word vocabularies, which contain opinion and sentiment related words.

Social Media Corpora The English Twitter corpus is from Archive Team's Twitter stream grab[4]. The Sina Weibo corpus comes from Open Weiboscope Data Access[5] (Fu, Chan, and Chau 2013). Both corpora cover the whole year of 2012. We then downsample each corpus to 100 million messages, each containing at least 10 characters, normalize the text (Han, Cook, and Baldwin 2012), lemmatize the text (Manning et al. 2014) and use LTP (Che, Li, and Liu 2010) to do Chinese word segmentation.

Entity Linking and Monolingual Word Vectors After preprocessing the corpora, we first do entity linking. For the Twitter corpus, we use Wikifier (Cheng and Roth 2013; Ratinov et al. 2011), a widely used entity linker. Because no suitable Chinese entity linking tool is available, we implement our own tool that is optimized for high precision. This tool prefers to link an entity with a surface form that appears more frequently in our corpus. Next we use Word2Vec (Mikolov et al. 2013) to train English and Chinese word embedding respectively.

Bilingual Lexicon Our bilingual lexicon is collected from Microsoft Translator[6], which translates English words to multiple Chinese words with confidence scores. [7]

Social Word Vocabulary Our social word vocabularies come from Empath (Fast, Chen, and Bernstein 2016) and OpinionFinder (Choi et al. 2005) for English, and TextMind (Gao et al. 2013) for Chinese. Empath is similar to LIWC (Pennebaker, Francis, and Booth 2001), but with more words and more categories and publicly available. We manually select 91 categories of words that are relevant to human perception and psychological processes. OpinionFinder consists of words relevant to opinion and sentiment. TextMind is a Chinese counterpart for Empath. In summary, we obtain 3343 words from Empath, 3861 words from OpinionFinder, and 5574 unique words in total.

Task 1: Mining cross-cultural differences of named entities in social media

This task is to discover and quantify cross-cultural differences of concerns towards name entities. We first explain how we obtain the ground truth from human annotators, then present several baseline methods to this problem and finally show and discuss our experiment results in detail.

Ground Truth Harris (1954) states that the meaning of words is evidenced by the contexts they occur with. Likewise, in this work, we assume that the cultural properties of an entity can be captured by the terms they always co-occur with in large text corpus. Thus, for each named entity, we present human annotators with two lists of 20 most co-occurred words with the named entity, from Twitter and Weibo respectively. We select 700 named entities for annotators to label, which are the most frequently mentioned both in Twitter and Sina Weibo. Annotators are instructed to rate the relatedness between the two word lists with one of following labels: "very different", "different", "hard to say", "similar" and "very similar".[8] We chose to annotate the data in this way mainly to avoid subjectivity and be efficient. We argue that since the words presented to the annotators come from social media messages, the social elements are already embedded in these words.

We then map the labels to numerical scores from 1 to 5 and use the average scores from the annotators as the ground truth for score ranking and binary classification evaluation. For the binary classification problem, an entity is considered culturally similar if the score is larger than 3.0, and culturally different otherwise. The inter-annotator agreement is 0.672 by Cohen's kappa coefficient, suggesting substantial correlation, according to the Wikipedia entry of Cohen's kappa.

Baseline and Our Methods We propose eight benchmark methods. The first three are *distribution*-based, while the next two are *transformation*-based. The last three, namely MultiCCA, MultiCluster and Duong are three popular bilingual word representation models for general use. Distribution-based methods compare lists of surrounding English and Chinese terms, denoted as L_E and L_C, by computing cross-lingual relatedness between two lists, though different baselines differ in the selection of words and the way of computing similarities. Transformation-based methods compute the vector representation in English and Chinese corpus respectively, and then train a transformation. Bilingual word representations based methods use the existing state-of-the-art models and then compute the similarities between two bilingual word vectors as $clsim$.

- *Bilingual Lexicon Jaccard Similarity (BL-JS)* BL-JS uses the bilingual lexicon to translate L_E to a Chinese word list L_E^* as a medium and then calculates the Jaccard Similarity between L_E^* and L_C as J_{EC}. Similarly, we can compute J_{CE}. Finally, we compute $\frac{J_{EC}+J_{CE}}{2}$ as the cross-cultural similarity of this given name entity.
- *WordNet Wu-Palmer Similarity (WN-WUP)* Instead of using the bilingual lexicon and Jaccard Similarity, WN-WUP uses Open Multilingual Wordnet (Wang and Bond 2013; Bond and Foster 2013) to calculate the average similarity of two lists of words from different languages.
- *Word Embedding based Jaccard Similarity (EM-JS)* EM-JS is very similar to BL-JS, except that its L_E and L_C are generated by ranking the similarities between the name of

[4] https://archive.org/details/twitterstream

[5] http://weiboscope.jmsc.hku.hk/datazip/

[6] http://www.bing.com/translator/api/Dictionary/Lookup?from=en&to=zh-CHS&text=<input_word>

[7] All named entities and slang terms used in the following experiments are excluded from this bilingual lexicon.

[8] All four annotators are native Chinese speakers but have excellent command of English and lived in the US extensively. Annotators are educated to have shared understanding of the five-level labels with selected examples. We do not choose ranking based annotation method as it demands annotators to look at 40+40 words for the two terms in two languages before a decision can be made, which is more expensive and harder to administer in our opinion.

Entity	Twitter topics	Weibo topics
Maldives	coup, president Nasheed quit, political crisis	holiday, travel, honeymoon, paradise, beach
Nagoya	tour, concert, travel, attractive, Osaka	Mayor Takashi Kawamura, Nanjing Massacre, denial of history
Yao Ming	NBA, Chinese, good player, Asian	patriotism, collective values, Jeremy Lin, Liu Xiang, Chinese Law maker, gold medal superstar
University of Southern California	college football, baseball, Stanford, Alabama, win, lose	top destination for overseas education, Chinese student murdered, scholars, economics, Sino American politics

Table 1: Selected culturally different named entities, with Twitter and Weibo's trending topics manually summarized

entities and all English words and Chinese words respectively.

- *Linear Transformation (LTrans)* We follow the steps in Mikolov et al. (2013) to train a transformation matrix between *EnVec* and *CnVec*, using 3000 translation pairs with confidence of 1.0 in the bilingual lexicon. Given a named entity, this solution simply calculates cosine similarity between the vector of its English name and the *transformed* vector of its Chinese name.
- *Bilingual Lexicon Space (BLex)* This baseline is similar to *SocVec* but it does not utilize social word vocabularies and solely uses the bilingual lexicon.
- *MultiCCA* (Ammar et al. 2016) This method takes two mono-lingual word embeddings and a bilingual lexicon as input and develop a bilingual word representations. We use both the Microsoft bilingual lexicon (BL) and the bilingual social lexicon (BSL) we constructed as the bilingual lexicon to compare their effectiveness. Dimensionality is tuned from $\{50, 100, 150, 200\}$ in all methods.
- *MultiCluster* (Ammar et al. 2016) This method requires re-training the bilingual word embeddings from the two mono-lingual corpora with a bilingual lexicon. We also use our BSL as an additional test (MultiCluster-BSL).
- *Duong* (Duong et al. 2016) Similar to MutltiCluster, this method retrains the embeddings from mono-lingual corpora with an EM style training algorithm.
- *Our SocVec-based method* With the help of our constructed *SocVec*, given a named entity with its English and Chinese name, we simply compute the similarity between their *SocVec*s as its cross-cultural difference score.

Experimental Results For qualitative evaluation, Table 1 shows some of the most culturally different entities mined by our method. The hot and trending topics on Twitter and Weibo are manually summarized to help explain the cultural difference. The perception of these entities diverges widely between English and Chinese social networks, thus suggesting significant cross-cultural differences.

In Table 2, we evaluate the benchmark methods and our approach with three metrics: Spearman and Pearson correlation on the ranking problem, and Mean Average Precision (MAP) on the classification problem. The *BSL* of *SocVec:opn* uses only OpinionFinder as English sociolinguistic vocabulary, while *SocVec:all* uses the union of

Method	Spearman	Pearson	MAP
BL-JS	0.276	0.265	0.644
WN-WUP	0.335	0.349	0.677
EM-JS	0.221	0.210	0.571
LTrans	0.366	0.385	0.644
BLex	0.596	0.595	0.765
MultiCCA-BL(dim=100)	0.325	0.343	0.651
MultiCCA-BSL(dim=150)	0.357	0.376	0.671
MultiCluster-BL(dim=100)	0.365	0.388	0.693
MultiCluster-BSL(dim=100)	0.391	0.425	0.713
Duong-BL(dim=100)	0.618	0.627	0.785
Duong-BSL(dim=100)	0.632	0.651	0.813
SocVec:opn	0.668	0.662	**0.834**
SocVec:all	**0.676**	**0.671**	**0.834**
SocVec:noun	0.564	0.562	0.756
SocVec:verb	0.615	0.618	0.779
SocVec:adj.	0.636	0.639	0.800

Table 2: Comparison of Different Methods

Similarity	Spearman	Pearson	MAP
PCorr.	0.631	0.625	0.806
L1 + M	0.666	0.656	0.824
Cos	**0.676**	0.669	**0.834**
L2 + E	**0.676**	**0.671**	**0.834**

Table 3: Evaluation of Different Similarity Functions

Emapth and OpinionFinder vocabularies.[9] To show the effectiveness of social-linguistic vocabulary versus other type of words as the bridge between the two cultures, we also compare the results using sets of nouns, verbs and adjectives within the same *SocVec* framework. All vocabularies under comparison are of similar sizes (around 5000), which also indicates that the improvement of our method is not just the result of sparsity. Results show that *SocVec* models, and in particular, the *SocVec* model using the social words as cross-lingual media, performs the best.

We also evaluate the effectiveness of four different similarity options in *SocVec*, namely, Pearson Correlation Coefficient (*PCorr.*), L1-normalized Manhattan distance (*L1+M*), Cosine Similarity (*Cos*) and L2-normalized Euclidean dis-

[9]Having tuned the parameters, we use the best parameters for the *SocVec* methods: 5-word context window and 150 dimensions used in training monolingual word vectors, cosine similarity as the *sim* function within the *SocVec* space, and "*Top*" as the pseudo-word generator.

Generator	Spearman	Pearson	MAP
Max.	0.413	0.401	0.726
Avg.	0.667	0.625	0.831
W.Avg.	0.671	0.660	0.832
Top	**0.676**	**0.671**	**0.834**

Table 4: Evaluation of Different Pseudo-word Generators

tance (*L2+E*). From Table 3, we conclude that among these four options, *Cos* and *L2+E* perform the best. Table 4 shows effect of using four different pseudo-word generator functions, from which we can infer that "*Top*" generator function performs best for it reduces the noise brought by the less probable translation pairs.

Task 2: Finding words for explaining slang terms across languages

In this section, we first introduce the ground truth and baseline methods for comparison. Then, we analyze the experimental results quantitatively and qualitatively.

Ground Truth We use an online Chinese slang glossary[10] consisting of 200 popular slang terms with English explanations. For English, we resort to a slang word list from OnlineSlangDictionary[11] with explanations and then downsample the list to 200 terms. For each Chinese slang term, the target English terms are hand picked from the English explanation. For each English slang term, the target Chinese terms are the word-to-word translation from the words hand picked from the English explanation. Different methods should produce a list of translation terms as similar as possible to the ground truth target terms. For example, we construct the ground truth target terms for the Chinese slang term "二百五" by manually labeling words related to its meaning in its explanation from the glossary:

二百五 A ***foolish*** person who is lacking in sense but still ***stubborn***, ***rude***, and ***impetuous***.

Baseline and Our Methods We propose two types of baseline methods for this task. The first type is based on well-known *on-line translators*, namely Google (Gg), Bing (Bi) and Baidu (Bd), as of August, 2017. Another baseline method specific to Chinese slang is from CC-CEDICT[12] (CC), an on-line public-domain Chinese-English dictionary, which is well updated with popular slang terms. Considering situations that many slang terms have literal meaning, it is unfair to retrieve target terms from such on-line translators by simply inputing slang terms without slang contexts. Thus, we use example slang sentences from some websites (mainly from Urban Dictionary[13]) as input, so that translators have a great chance of knowing this is a slang use, rather than an ordinary term. [14] The following example shows how we obtain the target translation terms for the slang word "fruitcake" (an insane person) from Google Translator:
Input Sentence: Oh man, you don't want to date that girl. She's always drunk and yelling. She is a total fruitcake.[15]
Google Translation: 哦, 男人, 你不想约会那个女孩。她总是喝醉了, 大喊大叫。她是一个水果蛋糕。

Since all possible target terms must come from the bilingual lexicon, we can score each of them and consider the top five as the target terms. Given a source term to be translated, several such *scoring-based baseline methods* are as follows. Linear Transform (LT), MultiCCA, MultiCluster and Duong method score the candidate target terms by computing cosine similarities in their constructed bilingual vector space with the tuned best settings in previous evaluation. A more sophisticated baseline (TransBL) leverages the bilingual lexicon: for each candidate target term w, we first obtain its translations T_w back into the source language and then calculate the average word similarities between the source term and T_w as the score of w.

Our *SocVec-based method* (SV) simply calculates the cosine similarities between the source term and each candidate target term within *SocVec* space as scores.

Experimental Results To quantitatively evaluate our methods, we need to measure similarities between the produced target term set and the ground truth word set. Exact-matching Jaccard similarity is too strict to capture valuable relatedness between two word sets. We argue that average cosine similarity (ACS) between two sets of word vectors is a better metric to evaluate the similarity between two word sets. The following equation illustrates such computation, where A and B are the two word sets, $\mathbf{A_i}$ and $\mathbf{B_j}$ denotes the word vector of the i^{th} word in A and j^{th} word in B respectively. The word vectors used in ACS computation is a third-party pre-trained embedding[16] and thus the ACS computation is fair over different methods. Table 6 shows the sums of ACS over 200 slang translations.

$$ACS(A,B) = \frac{1}{|A||B|}\sum_{i=1}^{|A|}\sum_{j=1}^{|B|}\frac{\mathbf{A_i}\cdot\mathbf{B_j}}{\|\mathbf{A_i}\|\|\mathbf{B_j}\|}$$

Experimental results of Chinese and English slang translation in terms of the sum of *ACS* over 200 terms are shown in Table 6. The performance of on-line translators for slang typically depends on human-set rules and supervised learning on well-annotated parallel corpora, which are rare and costly, especially for social media where slang emerges the most. This could be a possible reason why they do not perform well. Linear transformation model is trained on translation pairs with high confidence in the bilingual lexicon, which contains little information about the OOV slang terms and social context on them, which is why LT method performs badly. *BL* method is competitive because its similarity computations are within monolingual semantic spaces and it uses a bilingual lexicon to transform, while it loses

[10] https://www.chinasmack.com/glossary

[11] http://onlineslangdictionary.com/word-list/

[12] https://cc-cedict.org/wiki/

[13] http://www.urbandictionary.com/

[14] Nevertheless, we noticed that the on-line translators often ignore the slang contexts and still produce literal translations.

[15] http://www.englishbaby.com/lessons/4349/slang/fruitcake

[16] https://nlp.stanford.edu/projects/glove/

Slang	Explanation	Google	Bing	Baidu	Ours
浮云	something as ephemeral and unimportant as "passing clouds"	clouds	nothing	floating clouds	nothingness, illusion
水军	"water army", people paid to slander competitors on the Internet and to help shape public opinion	Water army	Navy	Navy	propaganda, complicit, fraudulent
floozy	a woman with a reputation for promiscuity	N/A	劣根性 (depravity)	荡妇(slut)	骚货(slut),妖精(promiscuous)
fruitcake	a crazy person, someone who is completely insane	水果蛋糕 (fruit cake)	水果蛋糕 (fruit cake)	水果蛋糕 (fruit cake)	怪诞(bizarre),令人厌烦(annoying)

Table 5: Slang Translation Examples

the information from the related words which are not in the bilingual lexicon. Our method (SV) outperforms baselines by directly using the distances in our proposed bilingual embeddings *SocVec*, which proves that *SocVec* can capture the cross-cultural similarities between terms. To qualitatively evaluate our model, in Table 5, we present several examples of our translations for Chinese and English slang terms as well as their explanations from glossaries. Our results are highly correlated with these explanations and capture their core semantics, whereas most online translators just offer literal translatation of such slang terms, even with the ample slang contexts. Additionally, we take a step forward to directly translate between English slang terms and Chinese slang terms by simply filtering out ordinary (non-slang) words in the original target term lists. Examples are shown in Table 7.

Gg	Bi	Bd	CC	LT
18.24	16.38	17.11	17.38	9.14
TransBL	MultiCCA	MultiCluster	Duong	SV
18.13	17.29	17.47	20.92	**23.01**

(a) Chinese Slang to English

Gg	Bi	Bd	LT	TransBL
6.40	15.96	15.44	7.32	11.43
MultiCCA	MultiCluster	Duong	SV	
15.29	14.97	15.13	**17.31**	

(b) English Slang to Chinese

Table 6: ACS Sum Results of Slang Translation

Chinese Slang	English Slang	Explanation
萌	adorbz, adorb, adorbs, tweeny, attractiveee	cute, adorable
二百五	shithead, stupidit, douchbag	A foolish person
鸭梨	antsy, stressy, fidgety, grouchy, badmood	stress, pressure, burden

Table 7: Slang-to-Slang Translation Examples

Related Work

Cross-cultural studies have been conducted in sociology and anthropology for many years. Recently, some researchers propose studying cross-cultural analysis through text mining and NLP techniques. Nakasaki et al. (2009) and Elahi et al. (2012) show that User Generated Contents (UGC), like microblog and user comment, are valuable and essential resources. The most relevant work to the Task 1 is Pennebaker et al. (2016), which studies the cross-cultural differences in word usage between Australian and American English through their proposed "socio-linguistic features" (similar to our social words), with a supervised model which is dependent on large volume of training data. To the best of our knowledge, we are among the first to focus on cross-cultural differences in named entities and to propose a straightforward but effective unsupervised approach.

Previous computational work on slang mainly focuses on automatic discovering of slang terms (Elsahar and Elbeltagy 2014) and normalization of noisy texts (Han, Cook, and Baldwin 2012). Research on automatic translation or explanation for slang terms in another language is missing from the literature. Our work on Task 2 fills the gap by directly computing cross-cultural similarities to find the most similar words in another language.

Most existing cross-lingual word representations rely on expensive parallel corpora with word or sentence alignments (Klementiev, Titov, and Bhattarai 2012; Kočiskỳ, Hermann, and Blunsom 2014) or a supervised model to learn a transformation matrix between two monolingual vector spaces (Mikolov, Le, and Sutskever 2013). Such work often aims to improve monolingual tasks and cross-lingual document classification, which does not require cross-cultural signals. We put our work in a broader context of building bilingual word representations by positioning it in the survey of Ruder (2017): our work is "monolingual mapping" based, uses only lexicon resource and maps monolingual vector spaces into a common high-dimensional third space by incorporating social words as pivot, where orthogonality is approximated by setting clear meaning to each dimension of *SocVec* space.

Conclusion

In this paper, we conclude that the cultural properties and social elements of a term (including both entity names and slang terms) can be effectively embedded by its similarities to social words with the help of our proposed *SocVec*, which

enables the comparison between two incompatible monolingual semantic spaces. Our proposed framework can be valuable assistance to cross-cultural social studies, acting as a building block for computing such cross-cultural differences and similarities. The two novel tasks with datasets as well as benchmark results also benefit further study in related topics of computational social science.

References

Ammar, W.; Mulcaire, G.; Tsvetkov, Y.; Lample, G.; Dyer, C.; and Smith, N. A. 2016. Massively multilingual word embeddings. *arXiv preprint arXiv:1602.01925*.

Bengio, Y.; Ducharme, R.; Vincent, P.; and Janvin, C. 2000. A neural probabilistic language model. *Journal of Machine Learning Research* 3:1137–1155.

Berend, G. 2017. Sparse coding of neural word embeddings for multilingual sequence labeling. *TACL* 5:247–261.

Bond, F., and Foster, R. 2013. Linking and extending an open multilingual wordnet. In *ACL (1)*, 1352–1362.

Camacho-Collados, J.; Pilehvar, M. T.; Collier, N.; and Navigli, R. 2017. Semeval-2017 task 2: Multilingual and cross-lingual semantic word similarity. In *SemEval*.

Che, W.; Li, Z.; and Liu, T. 2010. Ltp: A chinese language technology platform. In *ACL*, 13–16. Association for Computational Linguistics.

Cheng, X., and Roth, D. 2013. Relational inference for wikification. In *EMNLP*.

Choi, Y.; Cardie, C.; Riloff, E.; and Patwardhan, S. 2005. Identifying sources of opinions with conditional random fields and extraction patterns. In *HLT-EMNLP*, 355–362. Association for Computational Linguistics.

Duong, L.; Kanayama, H.; Ma, T.; Bird, S.; and Cohn, T. 2016. Learning crosslingual word embeddings without bilingual corpora. *arXiv preprint arXiv:1606.09403*.

Elahi, M. F., and Monachesi, P. 2012. An examination of cross-cultural similarities and differences from social media data with respect to language use. In *LREC*, 4080–4086.

Elsahar, H., and Elbeltagy, S. R. 2014. A fully automated approach for arabic slang lexicon extraction from microblogs. 79–91.

Fast, E.; Chen, B.; and Bernstein, M. S. 2016. Empath: Understanding topic signals in large-scale text. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, 4647–4657. ACM.

Fu, K.-w.; Chan, C.-h.; and Chau, M. 2013. Assessing censorship on microblogs in china: Discriminatory keyword analysis and the real-name registration policy. *IEEE Internet Computing* 17(3):42–50.

Gao, R.; Hao, B.; Li, H.; Gao, Y.; and Zhu, T. 2013. Developing simplified chinese psychological linguistic analysis dictionary for microblog. In *International Conference on Brain and Health Informatics*, 359–368. Springer.

Garimella, A.; Mihalcea, R.; and Pennebaker, J. W. 2016. Identifying cross-cultural differences in word usage. In *COLING*.

Han, B.; Cook, P.; and Baldwin, T. 2012. Automatically constructing a normalisation dictionary for microblogs. In *EMNLP-CoNLL*, 421–432.

Harris, Z. S. 1954. Distributional structure. *Word* 10(2-3):146–162.

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

Kočiskỳ, T.; Hermann, K. M.; and Blunsom, P. 2014. Learning bilingual word representations by marginalizing alignments. *arXiv preprint arXiv:1405.0947*.

Le, Q. V., and Mikolov, T. 2014. Distributed representations of sentences and documents. In *ICML*.

Luong, T.; Pham, H.; and Manning, C. D. 2015. Bilingual word representations with monolingual quality in mind. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, 151–159.

Manning, C. D.; Surdeanu, M.; Bauer, J.; Finkel, J. R.; Bethard, S.; and McClosky, D. 2014. The stanford corenlp natural language processing toolkit. In *ACL (System Demonstrations)*, 55–60.

Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G. S.; and Dean, J. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, 3111–3119.

Mikolov, T.; Le, Q. V.; and Sutskever, I. 2013. Exploiting Similarities among Languages for Machine Translation. *arXiv.org*.

Nakasaki, H.; Kawaba, M.; Yamazaki, S.; Utsuro, T.; and Fukuhara, T. 2009. Visualizing cross-lingual/cross-cultural differences in concerns in multilingual blogs. In *ICWSM*.

Pennebaker, J. W.; Francis, M. E.; and Booth, R. J. 2001. Linguistic inquiry and word count: Liwc 2001. *Mahway: Lawrence Erlbaum Associates* 71(2001):2001.

Pennington, J.; Socher, R.; and Manning, C. D. 2014. Glove: Global vectors for word representation. In *EMNLP*, volume 14, 1532–1543.

Ratinov, L.; Roth, D.; Downey, D.; and Anderson, M. 2011. Local and global algorithms for disambiguation to wikipedia. In *ACL*, 1375–1384. Association for Computational Linguistics.

Ruder, S. 2017. A survey of cross-lingual embedding models. *arXiv preprint arXiv:1706.04902*.

Smith, S. L.; Turban, D. H. P.; Hamblin, S.; and Hammerla, N. Y. 2017. Offline bilingual word vectors, orthogonal transformations and the inverted softmax. *ICLR* abs/1702.03859.

2016. Proceedings of the 54th annual meeting of the association for computational linguistics, ACL 2016, august 7-12, 2016, berlin, germany, volume 1: Long papers.

Wang, S., and Bond, F. 2013. Building the chinese open wordnet (cow): Starting from core synsets. In *Proceedings of the 11th Workshop on Asian Language Resources, a Workshop at IJCNLP*, 10–18.