Ermes: Emoji-Powered Representation Learning for Cross-Lingual Sentiment Classification*

Zhenpeng Chen, Sheng Shen, Ziniu Hu, Xuan Lu, Qiaozhu Mei, Xuanzhe Liu

Abstract

Most existing sentiment analysis approaches heavily rely on a large amount of labeled data that usually involve time-consuming and error-prone manual annotations. The distribution of this labeled data is significantly imbalanced among languages, e.g., more English texts are labeled than texts in other languages, which presents a major challenge to cross-lingual sentiment analysis. There have been several cross-lingual representation learning techniques that transfer the knowledge learned from a language with abundant labeled examples to another language with much fewer labels. Their performance, however, is usually limited due to the imperfect quality of machine translation and the scarce signal that bridges two languages. In this paper, we employ emojis, a ubiquitous and emotional language, as a new bridge for sentiment analysis across languages. Specifically, we propose a semi-supervised representation learning approach through the task of emoji prediction to learn cross-lingual representations of text that can capture both semantic and sentiment information. The learned representations are then utilized to facilitate cross-lingual sentiment classification. We demonstrate the effectiveness and efficiency of our approach on a representative Amazon review data set that covers three languages and three domains.

1 Introduction

In the past decades, sentiment analysis has been a research focus in various communities such as natural language processing [1, 2], Web mining [3, 4], information retrieval [5, 6], ubiquitous computing [7, 8], and human-computer interaction [9, 10]. As an important way to understand attitudes and opinions of users, sentiment analysis is especially valuable in applications such as customer feedback tracking [11], sales prediction [12], product ranking [13], stock market prediction [14], opinion polling [15], and even election outcome prediction [16]. The rapid growth and availability of sentimental corpora on the Web such as blogs, tweets, user reviews, and forum discussions have accelerated the sentiment analysis research.

Existing work on sentiment analysis mainly focuses on English texts [1, 2, 5, 6]. Although a few efforts have been spent on other languages such as Japanese, these efforts are far from sufficient, given that 74.7% of Internet users are non-English speakers.[1] The sparsity of labeled examples interacts with insufficient attention, making the research progress of sentiment analysis across language more and more imbalanced.

An intuitive idea to address this problem is to leverage the labeled resources in English (i.e., the source language) to learn the sentiment classifier for other languages (i.e., the target language), which is the so called ***cross-lingual sentiment classification*** in the research community [17]. In practice, the biggest challenge of the cross-lingual sentiment classification is the ***linguistic gap between English and the target language***. To increase the labeled examples in the target language, most of the recent studies use existing machine translation tools to generate *pseudo parallel texts* and then learn bilingual representations for the downstream sentiment classification tasks [17, 18, 19, 20, 21]. More specifically, these methods usually force the naturally aligned bilingual texts to share the same representation [20]

*Corresponding: liuxuanzhe@pku.edu.cn. Z. Chen and S. Shen made equal contribution to this work.

[1] https://www.internetworldstats.com/stats7.htm, retrieved on May 12, 2018.

(e.g., through a unified embedding) from different languages [18]. In these approaches, the quality of representations heavily relies on the quality of machine translation from English to the target language, which is currently far from perfect [22]. Additionally, most of the existing methods capture only the semantic correlations through the pseudo parallel texts, but neglect the subtlety of sentiment signals [21]. The pitfall either appears in the machine translation or the representation learning phase. For example, in Japanese, the common expression "湯水のように使う" has a negative sentiment, describing the behavior of excess or waste. However, when directly translated to English, "use it like hot water," it not only loses the negative sentiment but also becomes an odd expression in English. Hence, simply leveraging pseudo parallel text translated from a single language may miss the language-specific semantics and sentiments.

A fundamental problem leading to these pitfalls is the lack of sentimental signals bridging two languages. In a way, all machine translation systems are leveraging lexical or semantic relations (e.g., co-occurrence in parallel corpora or other resources) as the bridge between two languages. But sentiments are more subtle and sparse than semantics; they either don't exist in the parallel texts the machine translation algorithms were trained on, or they are missed when optimizing the objective of machine translation. We envision that a sound solution should either explore stronger sentimental signals that bridge different languages, or make sentiments a more direct objective in cross-lingual text analysis, or both.

In this paper, we employ such a new signal, i.e., emojis, which are reported to be a ubiquitous language adopted by worldwide users from various backgrounds [23] to express their sentiment in different languages [24]. More specifically, we use emojis as a "proxy" of sentiment labels that resides in cross-lingual texts and try to learn a text representation for sentiments that are generalizable across different languages. We propose Ermes, a novel emoji-powered text representation learning framework for the cross-lingual sentiment classification. In Ermes, the embedding of every single language is first derived based on how emojis are used among the texts. Instead of relying on parallel corpora, we use a large-scale collection of Tweets in both the source (English) language and the target language to learn the original representations for both languages. Without explicit sentiment labels, emojis used in Tweets serve as distant supervision for learning the text representations.

Detailedly, the workflow of Ermes consists of the following phases. First, we use large-scale Tweets to learn word embeddings (through Word2Vec [25]) of both the source and the target languages in an unsupervised way. Second, in a distant-supervised way [26, 27, 2, 28], we use emojis as complementary sentiment labels to transform the word embeddings into a higher-level sentence representation that encodes rich sentiment information via an emoji-prediction task (through an attention-based stacked bi-directional LSTM model). This step is also conducted separately for both the source and the target languages. Third, we translate the labeled English data into the target language (through an off-the-shelf machine translation system), represent the pseudo parallel texts with the pre-trained language-specific models, and use an attention model to train the final sentiment classifier.

The performance of Ermes is evaluated on a representative Amazon review data set that has been used in various cross-lingual sentiment analysis studies [17, 18, 29], covering nine tasks combined from three target languages (i.e., Japanese, French, and German) and three domains (i.e., book, DVD, and music). Our Ermes outperforms existing approaches on all these tasks in terms of classification accuracy. Experiments show that the emoji-powered model still works well even when the unlabeled and labeled data are limited. To evaluate the generalizability of Ermes, we also apply our approach on Tweets and it can also achieve the state-of-the-art performance.

To sum up, this paper makes the following major contributions:

- To the best of our knowledge, we are the first to leverage emojis as new complementary sentiment labels to address the cross-lingual sentiment classification problem.
- We propose a representation learning approach to incorporate the emoji power into the cross-lingual sentiment classification task. More specifically, we introduce a hierarchical attention-based LSTM network to capture the sentiment information implied in the ubiquitous emoji usage.
- We apply our Ermes approach on reviews and Tweets, and demonstrate that Ermes can significantly

outperform the state-of-the-art results on benchmark data sets.

- We systematically evaluate the power of emojis in the sentiment classification problem and discuss insightful implications for further research[2].

The rest of this paper is organized as follows. Section 2 presents a sample to illustrate our motivation. Section 3 formulates the problem and presents our cross-lingual representation learning approach. Section 4 evaluates our approach on nine tasks and explores the power of emojis in the learning process. Section 5 discusses the data size sensitivity and generalizability of our approach. Section 6 presents the implications and limitations of this study, followed by related work in Section 7 and concluding remarks in Section 8.

2 An illustrating Example

We start with a simple example to illustrate the idea of adopting emojis for sentiment classification through representation learning.

As is mentioned in Section 1, "湯水のように使う" is a common negative expression in Japanese. However, when it is translated to English, the odd expression "use it like hot water" loses the negative sentiment. The loss of this sentiment may further reveal the limitation of existing cross-lingual sentiment classification approaches. Since they mainly learn sentiment knowledge from the labeled data in English and transfer the knowledge to the target language through translation, the language-specific sentiment in the target language will be missed.

To capture the language-specific sentiment knowledge in the target language, we need to find a sentiment "proxy" that is widely used in the target language to supply its absence of labeled data. Considering the ubiquitous characteristic [23, 30] and the sentiment expression ability [24, 31, 32] of emojis, we introduce them into our cross-lingual sentiment classification.

As "湯水のように使う" is used to describe the waste behavior, it could be usually expressed together with negative emojis. For example, one can say "彼はお湯水のような紙を使う 😡" ("*He uses paper like hot water* 😡"). In practice, "😡" is also used frequently in some obvious negative Japanese contexts such as " 私は本当に怒っている 😡" ("*I am really angry* 😡"). As the current English sentiment classification approaches can easily detect the negative sentiment in this expression "I am really angry". Through translation, its Japanese version can also be correctly classified. Now, to correctly identify the sentiment in the expression "湯水のように使う", we just need to use the emoji 😡 as a bridge to transfer the negative signal from "私は本当に怒っている" to it. To this end, we propose the Ermes approach to represent the sentences co-occurred by the same emojis similarly. Then the similar representations facilitate the sentiment classifier to classify them as the same sentiment. In the next section, we will introduce the framework of our approach in details.

3 The Ermes Approach

In this section, we propose our approach to learning the representations for cross-lingual sentiment classification by leveraging the power of emojis.

First, we formulate our problem. The cross-lingual sentiment analysis aims to use the labeled data in the source language (with relatively sufficient annotated resources) to learn a model that can classify the sentiment of test data in target language (with poor annotated resources). In this paper, we use *English* as the source language. We have labeled English documents $L_S = \{x_i, y_i\}_{i=1}^{N_1}$ (abbreviated as "source-language texts"), where x_i is the text and y_i is the sentiment label. In the training process, we also use the unlabeled data both in the source language $U_S = \{x_i\}_{i=1}^{N_2}$ and the target language $U_T = \{x_i\}_{i=1}^{N_3}$. In practice, there exist a large volume of unlabeled data that can be easily accessed in public platforms

[2]We plan to release the code and pre-trained models for all languages when the work is published.

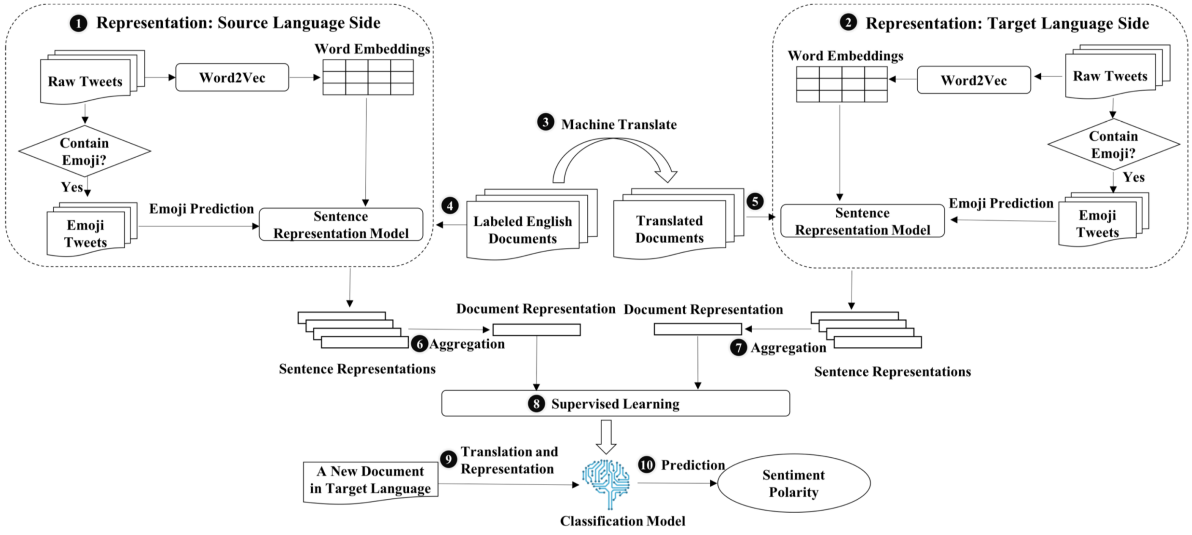


Figure 1: The overview of our Ermes approach.

such as Twitter, Facebook, and so on. In the unlabeled data, there are many sentences containing emojis that can be used to capture the emotional information into our model, and we call such texts as *"emoji-texts"*. We denote the emoji-texts in the source language as $ET_S = \{x_i, e_i\}_{i=1}^{N_4}$ and emoji-texts presented in the target language as $ET_T = \{x_i, e_i\}_{i=1}^{N_5}$, where x_i represents the text by removing emojis and e_i represents the emojis contained in the raw text. Our task is to build the model that can classify the sentiment polarity of the texts presented in the target language (abbreviated as "target-language texts"), solely based on the source-language texts (i.e., L_S) and the unlabeled data (i.e., U_S, U_T, ET_S and ET_T). Finally, we use the labeled target-language texts $L_T = \{x_i, y_i\}_{i=1}^{N_6}$ to evaluate the model.

The framework of our approach is illustrated in Figure 1, with the following phases in its workflow. (1) We use U_S and ET_S to learn representations for the source language. (2) We use U_T and ET_T to learn representations for the target language. (3) We translate the documents in L_S into the target language. (4) We input each document in L_S into the source-language representation model and get sentence representation for every single sentence. (5) For each translated document, we input it into the target-language representation model and get sentence representations for every single sentence in it. (6) We aggregate the sentence representations to form a compact representation for the each source-language document. (7) We generate the representation for each translated document. (8) We merge the document representations of the source-language document and the translated document as features and the sentiment labels in L_S as the ground-truth to learn the final sentiment classification. (9) For a new document presented in the target language, we first translate it into English and then follow the previous steps to represent and merge the document representations as features. (10) The classifier predicts the sentiment polarity of the target-language sentiment based on its final representation. We next introduce the details.

3.1 Representation Learning

In this phase, we define representation rules for the source language and the target language, respectively. In practice, we can use the existing word embedding techniques to create representations at the word level and concatenate or average the word vectors in each document to represent it. However, as the goal of our study is the cross-lingual sentiment classification, we call for a better representation approach that can capture both the sequential relationships of words and the sentiment singal. Existing research

efforts in ubiquitous computing [23, 33] and human-computer interaction [31] have demonstrated that emojis can express the sentiment across languages. Thus, we choose emojis as weak but complementary sentiment labels and use the unlabeled texts to learn representations in a distant-supervised way. First, we create basic word embeddings based on large-scale unlabeled texts in an unsupervised way. Then, we devise a distant emoji-prediction task for learning the sentence representations, which can incorporate both the semantic and the sentiment information.

3.1.1 Unsupervised Word Embedding

To better represent words, we leverage large-scale public Tweets, which can be easily acquired using Twitter API [3]. The word embedding technique can encode every single word into a continuous vector space. Word embedding is usually performed in an unsupervised fashion as it leverages only the unlabeled raw texts to capture the semantic information of words. In this paper, we apply the *skip-gram* algorithm [25] to implement the word embeddings. The core idea of the skip-gram algorithm is to use the current word to predict the context words. The context range is determined by the parameter called "window size" in the algorithm. By predicting the nearby words, the words that usually occur in the similar contexts are embedded closely in the vector space, which indicates the semantic information of the embeddings. Noted that the skip-gram algorithm only adopts word co-occurrence signal, the learned word representation can mainly capture semantic information rather than sentimental information. As the word embedding approach is standard as the well-known *Word2Vec* [4], we do not describe the details of the whole process. Since the large scale of this vocabulary results in huge amount of parameters, we initialize the word embeddings by pre-trained Word2Vec model, the parameters of which can be further fine-tuned through the following sentence representation phase.

3.1.2 Distant-supervised Sentence Representation Learning

In this phase, we learn the sentence representations through an emoji-prediction task in a distant-supervised way. The model architecture of this process is illustrated in Figure 2, including the word representation, the sentence representation, and the emoji prediction. First, based on the word embeddings created above (refers to the Word Embedding Layer in Figure 2), we can represent every single word as a unique vector. Second, we use the stacked bi-directional LSTM layers and one attention layer to encode these word vectors into sentence representations. The attention layer takes the outputs of both the embedding layer and the two LTSM layers as input by the skip-connection algorithm [34] and enables the unimpeded information flow in the whole training process. Finally, the model parameters can be learned accordingly through the emoji prediction task.

We introduce the details of bi-directional LSTM layer, attention layer, and softmax layer, respectively.

Bi-directional LSTM layer. As a kind of recurrent neural network (RNN), the Long Short-Term memory Network (LSTM) [35] is suitable for processing the sequential data such as texts due to its recurrency nature. At each time step, LSTM combines the current input and knowledge from the previous time steps to update the state of the hidden layer. In addition, to tackle the gradient vanishing problem [36] of traditional RNNs, LSTM incorporates a gating mechanism to determine when and how the states of hidden layers can be updated. Each LSTM unit contains a memory cell and three gates (i.e., an input gate, a forget gate, and an output gate) [37]. The input and output gate control the input activations into the memory cell and the output flow of cell activations into the rest of the network, respectively. The memory cells in LSTM store the temporal states of the network. Each memory cell has a self-loop whose weight is controlled by the forget gate.

In this study, each training sample can be denoted as (x, e), where $x = [d_1, d_2, ..., d_L]$ is a sequence of word vectors that represent the plain text (by removing emoji) and e is the emoji contained in the text.

[3] https://developer.twitter.com/, retrieved on May 12, 2018.
[4] https://code.google.com/archive/p/word2vec/, retrieved on May 12, 2018.

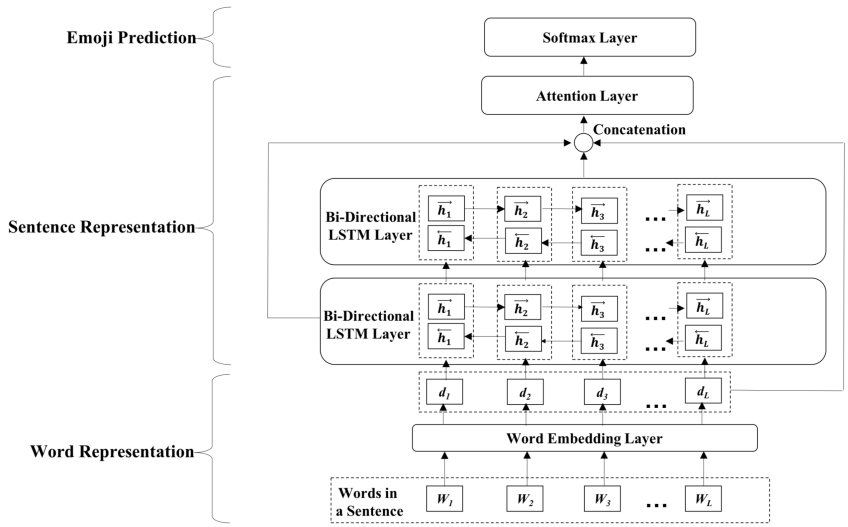


Figure 2: The architecture of sentence representation learning network.

At time step t, the LSTM computes unit states of the network as follows:

$$
\begin{aligned}
i^{(t)} &= \sigma(U_i x^{(t)} + W_i h^{(t-1)} + b_i), \\
f^{(t)} &= \sigma(U_f x^{(t)} + W_f h^{(t-1)} + b_f), \\
o^{(t)} &= \sigma(U_o x^{(t)} + W_o h^{(t-1)} + b_o), \\
c^{(t)} &= f_t \odot c^{(t-1)} + i^{(t)} \odot tanh(U_c x^{(t)} + W_c h^{(t-1)} + b_c), \\
h^{(t)} &= o^{(t)} \odot tanh(c^{(t)}),
\end{aligned}
$$

where $x^{(t)}$, $i^{(t)}$, $f^{(t)}$, $o^{(t)}$, $c^{(t)}$, and $h^{(t)}$ denote the input vector, the state of the input gate, forget gate, output gate, memory cell and hidden layer at time step t. W, U, b, respectively denote the recurrent weights, input weights, and biases. $\odot$ is the element-wise product. We can extract the latent vector for each time step t from LSTM. In order to capture the information from the past and future of a word in its current context, we use the bi-directional LSTM. We concatenate the latent vectors from both directions to construct a bi-directional encoded vector h_i for every single word vector d_i as:

$$
\begin{aligned}
\overrightarrow{h_i} &= \overrightarrow{LSTM}(d_i), i \in [1, L] \\
\overleftarrow{h_i} &= \overleftarrow{LSTM}(d_i), i \in [L, 1] \\
h_i &= [\overrightarrow{h_i}, \overleftarrow{h_i}],
\end{aligned}
$$

Attention Layer. As we employ the skip-connection that concatenates the outputs of the embedding layer and the two bi-directional LSTM layers as the input of the attention layer, the i-th word of the input sentence can be represented as u_i:

$$
u_i = [d_i, h_{i1}, h_{i2}],
$$

where d_i, h_{i1}, and h_{i2} denote the encoded vector of word extracted in the word embedding layer, the first bi-directional LSTM, and the second bi-directional LSTM, respectively. Since not all words contribute equally to predicting emojis and classifying sentiment, we employ the attention mechanism [38] to determine the importance of every single word. The attention score of the i-th word is calculated by

$$a_i = \frac{exp(W_a u_i)}{\sum_{j=1}^{L} exp(W_a u_j)},$$

where W_a is the weight matrix for the attention layer. Then each sentence can be represented by the weighted sum of all words in it, using the attention scores as weights. It means that the sentence representation is calculated as:

$$v = \sum_{i=1}^{L} a_i u_i,$$

Softmax Layer. The sentence representations are then transferred into the softmax layer, which returns a probability vector Y. Each element of this vector indicates the probability that this sentence contains a specific emoji. The i-th element of the probability vector is calculated as:

$$y_i = \frac{exp(v^T w_i + b_i)}{\sum_{j=1}^{K} exp(v^T w_j + b_j)},$$

where w_i and b_i are respectively the weight and bias of the i-th element. Then we use the cross entropy between the probability vector and the one-hot representation of the emoji contained in the sentence, and thus learn the models' parameters by minimizing the prediction error. After learning the parameters, we can extract the output of the attention layer to represent the input sentence. Through this emoji-prediction phase, the words with the distinctive emotion can be identified, and the texts co-used with the same emoji will be represented similarly. Recall the examples in Section 2, i.e., "彼はお湯水のような紙を使う 😡" and " 私は本当に怒っている 😡". As the two Japanese expressions have the same label (i.e., "😡"), they can be represented similarly. Given the fact that the scale of labeled data is quite limited, we should avoid the possible over-fitting problem by freezing the sentence representation model.

3.2 Supervised Training

After the distant-supervised emoji prediction phase, the sentences in each language with the similar semantic and sentiment information can be represented quite closely in the vector space. We can leverage the pre-trained representation model to further conduct the cross-lingual sentiment classification.

First, we use *Google Translate*[5] to translate the English text $x \in L_S$ to the target language. Second, we use the pre-trained representation model to generate representation for every single sentence in the original documents and the translated documents. Third, we aggregate these sentence representations to derive compact document representations. Because different parts of a document can always have quite different importances for the overall sentiment, we still adopt attention mechanism here. We denote the document vector as r and the sentence vector extracted from the pre-trained model as v. We then calculate r as:

$$\beta_i = \frac{exp(W_b v_i)}{\sum_{j=1}^{T} exp(W_b v_j)},$$

$$r = \sum_{i=1}^{T} \beta_i v_i,$$

where W_b is the weight matrix of the attention layer and β_i is the attention score of the i-th sentence in the document. For each labeled English document x_s and its corresponding translation x_t, supposing that the text representations of them are obtained in previous steps as r_s and r_t , we concatenate them as $r_c = [r_s, r_t]$. Then we take the concatenated representations as the input into a softmax layer and minimize the cross entropy with the real sentiment labels to learn the network parameters in a supervised way.

[5]https://translate.google.com/, retrieved on May 12, 2018.

3.3 Classification of Target-Language Document

So far, we have used the unlabeled data to learn representations for both the source and target languages. Based on the pre-trained representation models, we use the English labeled data to train a cross-lingual classification. Here, we introduce how to use this model to predict the sentiment polarity of the target-language document. To this end, we first translate the document into source language. Then based on the representation models mentioned above, the original document and translated document can be represented as r_t and r_s. Finally, we represent this document as $[r_s, r_t]$ and input it into the classifier. The classifier will output the predicted sentiment polarity.

4 Evaluation

As we have introduced the framework, we then use widely adopted benchmark data sets in cross-lingual sentiment analysis and a large-scale corpus of Tweets from Twitter data to validate the classification power of Ermes,.

4.1 The Data Set

The labeled data (L_S for training and L_T for testing) used in our work are from the Amazon review data set[6] created by Prettenhofer and Stein [17]. This data set is representative and widely used in various cross-lingual sentiment analysis work [17, 18, 39, 29]. It covers four different languages, i.e., English, Japanese, French, and German. For each language, it contains reviews from three different domains, i.e., book, DVD, and music. For each combination of language and domain, the data set contains 1,000 positive reviews and 1,000 negative reviews that have already been labeled. We select English as the source language and use the labeled English reviews to train the model. Then we test the sentiment classifier on the the reviews for the target language (i.e., Japanese, French, and German) of three domains. Therefore, we have a total of nine tasks to evaluate our approach. The data set provides the translations of the test data (target language reviews), so we need to translate only the training English reviews to target languages. For simplicity, all the reviews are tokenized and converted into lowercase. It should be noted that, in Japanese, words are not separated by whitespace. Hence, we use a tokenization tool called *MeCab*[7] to segment Japanese reviews.

As is described in Section 3, we also need the unlabeled data in both the source language and the target language to train the model. Here, we crawl English, Japanese, French, and German data from Twitter, where emojis are widely used. We preprocess the raw Tweets in the following ways:

- We remove the reTweets to ensure all the words appear in their original contexts.
- We remove the Tweets containing URLs in case the words' interpretations depend on external contents rather than just the surrounding texts.
- We tokenize all the Tweets into words and convert them into lowercase.
- We use a special token to replace all the mentions (e.g., @IMWUT and @UbiComp) so that they can be treated the same. For numbers in Tweets, we also perform such a process.
- We shorten the words with redundant characters into their canonical forms (e.g., "cooooool" and "cooool" are both converted to "cool").

After these steps, we get the final unlabeled source-language data (U_S) and unlabeled target-language data (U_T). As emojis are widely adopted in Twitter [40], we construct the emoji-texts (E_S and E_T) directly from the U_S and U_T. To fully exploit the characteristic properties of different languages, we

[6] https://www.uni-weimar.de/en/media/chairs/computer-science-department/webis/data/corpus-webis-cls-10/, retrieved on May 12, 2018.

[7] http://taku910.github.io/mecab, retrieved on May 12, 2018.

Table 1: The amount of the raw Tweets and emoji-texts.

Language	English	Japanese	French	German
Unlabeled Tweets	39.4M	19.5M	29.2M	12.4M
Emoji Tweets	6.6M	2.9M	4.4M	2.7M

extract the most frequently used 64 emojis , which cover about 70% of the total emoji usage in each language, with their corresponding Tweets for distant supervision. The Tweets that do not contain any of these emojis will be filtered out accordingly. As many Tweets contain repetitions of the same emoji or multiple different emojis, for each Tweet, we create separate samples for each unique emoji in it. For example, "I love you ❤️❤️💋" can be separated into two samples, i.e., ("I love you", "❤️") and ("I love you", "💋"). This processing makes our emoji prediction task a single-label classification task instead of a complicated multi-label one. The created Tweet samples are distributed into the E_S or E_T according to the language of their preceding Tweets. We present an overview of our U_S, U_T, E_S, and E_T in Table 1.

4.2 Implementation

In the word embedding process, word vectors for each language are initialized by Word2Vec [25]. We train the word embedding vectors on the unlabeled corpus using the skip-gram model with the window-size of 5. The word vectors are then fine-tuned during the sentence representation learning phase. To regularize our model, the L2 regularization of 10^{-6} is applied for embedding weights. Following previous work [41], the dropout with 0.5 rate is introduced before the softmax layer and after the word embedding layer. The hidden units of bi-directional LSTM layers are set as 1,024 (512 in each direction). We randomly split the emoji-texts into the training, validation, and test sets in the proportion of 7:2:1. Accordingly, we use early stopping [42, 43] to tune parameters based on the validation performance through 50 epochs, with mini-batch size of 250. We used the Adam [44] for optimization, with the two momentum parameters set to 0.9 and 0.999, respectively. The initial learning rate was set to 10^{-3}. In the supervised training phase, for exhaustive parameter tuning, we randomly select 90% of the labeled data as the training set and the rest 10% as the validation set. The whole framework is implemented with TensorFlow and run on a Nvidia M40 graphic card.

4.3 Baselines and Results

To evaluate the performance of Ermes, we employ some existing methods for comparison:

MT-BOW uses only the bag-of-words features to learn a linear classifier on the training data in the source language. Then it uses *Google Translate* to translate the test data to the source language and applies the pre-trained linear classifier to predict the sentiment polarity of the translated documents. It is a common naive baseline for measuring the performance, and we simply use the result reported in [17].

CL-SCL is the cross-lingual structural correspondence learning method proposed by Prettenhofer and Stein [17]. This approach defines the pivot words in the source language for the sentiment classification task, and then uses *Google Translate* to obtain the translation oracle for mapping these pivot words to the target language. The pivot feature vectors, which model the correlations between pivot and non-pivot words in the bag-of-words representations of target language, can be learned through the structural correspondence learning algorithm [45]. The feature vectors for each document in target language will finally be applied for a linear sentiment classifier.

CL-RL is the cross-lingual representation learning method proposed by Xiao and Guo [18]. It constructs a unified word representation that consists of language-specific components and shared components, for both the source and target languages. To establish connections between the two languages, this approach selects a set of critical bilingual word pairs based on the translation and the word frequency in the labeled data of each language, then forces each parallel word pair to share the same word representation. The document representation is computed by taking average over all words in the document.

Table 2: The accuracy (%) of the cross-lingual sentiment classifications for the nine benchmark tasks.

Target language	Domain	MT-BOW [17]	CL-SCL [17]	CL-RL [18]	DRL [20]	BiDRL [29]	Ermes
Japanese	Book	70.22	73.09	71.11	71.75	73.15	**78.60**
	DVD	71.30	71.07	73.12	75.40	76.78	**80.30**
	Music	72.02	75.11	74.38	75.45	78.77	**81.60**
French	Book	80.76	78.49	78.25	84.25	84.39	**86.90**
	DVD	78.83	78.80	74.83	79.60	83.60	**85.90**
	Music	75.78	77.92	78.71	80.09	82.52	**86.70**
German	Book	79.68	79.50	79.89	79.51	84.14	**86.70**
	DVD	77.92	76.92	77.14	78.60	84.05	**85.70**
	Music	77.22	77.79	77.27	82.45	84.67	**87.40**

Given the representation scheme, it trains a linear Support Vector Machine (SVM) model using labeled English data.

DRL is a direct document representation learning method proposed by Pham *et al.* [20], which extends the paragraph vector approach proposed by Le and Mikolov [46] into the bilingual setting. This approach predicts each word in the document by its context tokens and the document. During this process, it can learn word representations and document representations simultaneously. To adapt to the bilingual setting, DRL applies the translation method as *Google Translate* to construct parallel documents and each pair of parallel documents share the same document representation.

BiDRL is the state-of-the-art document representation learning method for the cross-lingual sentiment classification task proposed by Zhou *et al.* [29]. This approach uses *Google Translate* to create the labeled parallel documents. Similar to DRL, it also uses the paragraph vector approach to create document representations. Besides minimizing the differences of parallel document vectors, it also adapts the document representations in a sentiment level. It uses constraints to make the document vectors associated with different sentiments fall into different positions in the embedding space. Furthermore, it forces the document with large textual differences but the same sentiment to have similar representations. Based on the representation approach, it concatenates the vectors of one document in both languages to represent the document and trains a logistic regression sentiment classifier.

As the benchmark data sets have quite balanced positive and negative reviews as samples, we follow these aforementioned studies to use *accuracy* as the evaluation metric. We summarize the performance of the baseline methods and our Ermes in Table 2. All the baseline methods have been evaluated on the **same** benchmark data sets in the previous literature [17, 18, 29]. Just like our approach, they only leverage the 2,000 English labeled reviews in training process without any labeled target language resources. Hence, we take their reported results directly in this paper.

As illustrated in Table 2, the performance of Japanese tasks are worse than French and German tasks. It is reasonable. On one hand, Japanese words are not separated by whitespace and more difficult to tokenize, which could affect the classification performance. On the other hand, according to the language systems defined by ISO 639 [8], English, French, and German belong to the same language family (i.e., Indo-European), while Japanese belongs to Japonic. In fact, French and German are more similar to English than Japanese. It is easier to translate English to French and German and transfer the sentiment classification knowledge from English to them. Therefore, Japanese tasks are the most difficult in these tasks and all the previous methods can not achieve an accuracy above 80%. However, it is encouraging to find that our approach achieves significant improvement in Japanese tasks. We achieve a 80.30% accuracy in Japanese DVD task and a 81.60% accuracy in Japanese music task. The 78.60% accuracy in book task is also non-negligible as it outperforms the previous best performance (73.15%) with 5.45%. By contrast, the corresponding improvement from the the early representation learning approach CL-SCL (proposed in 2010) for this problem to the recent BiDRL (proposed in 2016) is only 0.06%. With respect to French and German tasks, although they are a little easier than Japanese, no existing approaches can achieve

[8] https://www.iso.org/iso-639-language-codes.html, retrieved on May 12, 2018.

Table 3: The performance of our original Ermes and the simplified S-Ermes without distant-supervised phase.

	Japanese			French			German		
	Book	DVD	Music	Book	DVD	Music	Book	DVD	Music
Ermes (Cross-Lingual)	78.60	80.30	81.60	86.90	85.90	86.70	86.70	85.70	87.40
S-Ermes (Cross-Lingual)	52.65	50.70	51.25	50.56	50.70	50.30	51.25	52.05	51.25
S-Ermes (Mixed Data)	75.35	73.25	74.60	79.25	80.20	81.95	79.85	78.65	75.85

85% accuracy on any of the six tasks. However, our approach can achieve accuracy higher than 85% on all of the six tasks, which indicates the power of leveraging emojis.

Then we compare the results more thoroughly and further demonstrate the advantages of our approach. As is shown, the representation learning approaches (CL-SCL, CL-RL, DRL, BiDRL, and Ermes) all outperform the naive MT-BOW on most tasks. It is reasonable because representation learning approaches represent words as high-dimensional vectors in a continuous space and thus overcome the feature sparsity of the traditional bag-of-words approaches. For more careful exploration, we observe the document representation approach (DRL, BiDRL, and Ermes) perform better than the representation learning approaches centered on word-level information (CL-SCL and CL-RL). It indicates that incorporating document-level information is more effective than focusing on independent words as it can capture the composability of the words. Then we compare the DRL with BiDRL and Ermes. Despite the three approaches all learn document representations and capture semantic relationships between words, DRL does not take sentiment information into the representations. As BiDRL and our Ermes outperform DRL in all the nine tasks, we find that the specific sentiment effort pays off. Finally, we find that our approach outperforms the BiDRL in all tasks. It can be attributed to two factors. First, our approach uses the attention mechanism that can treat every single word and sentence with different importances and thus create more effective document representations. Second, we use the emoji prediction task to learn the sentiment information while BiDRL directly forces the documents with the same sentiment to be close in the vector space. The labeled data that BiDRL needs to perform this adaption is very valuable but quite limited, which narrows its performance to a large extent. However, our approach leverages the strengths of distant-supervised learning and uses the texts containing emojis, which can be easily collected from the public platform and bring informative insights. Then we want explore the power of emojis in our learning process.

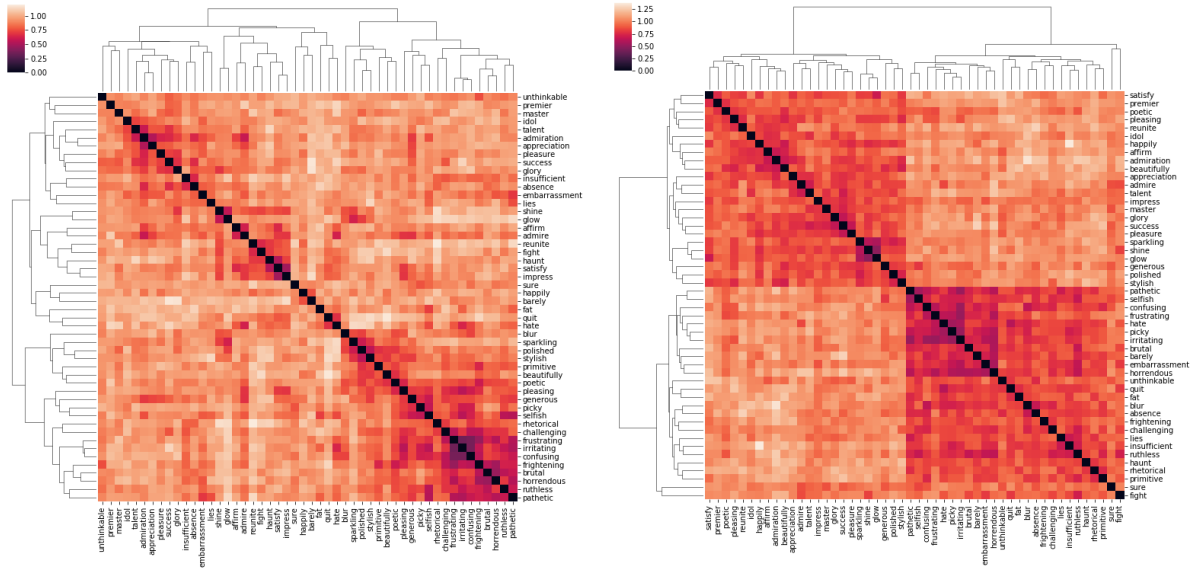
4.4 Power of Emojis

The core insight of our approach is using emojis to capture the sentiment information for representations in a distant-supervised way. To further evaluate the "emoji power", we conduct three subsequent experiments to investigate the effects of emojis on overall performance, representation learning, and document comprehension.

4.4.1 Overall Performance

To understand how emojis affect our cross-lingual sentiment model, we first remove our distant-supervised emoji-prediction phase and implement a simplified model for comparison. The simplified version (abbreviated as "S-Ermes") directly uses two attention layers to realize the representation transformation from word to sentence, and eventually to document. Compared to the original approach, the S-Ermes learns sentiment information only in the supervised way through the limited 2,000 English labeled reviews.

As is illustrated in Table 3 (Ermes vs. S-Ermes in cross-lingual setting), the performance of S-Ermes decreases dramatically compared to our original Ermes. The prediction accuracy among different tasks surpasses only quite a little compared to the uniform guess (50%). Apart from the impact of removing emoji-prediction, one alternative conjecture is that the 2,000 training reviews are insufficient to train such a complex network model, which leads to the well-known "over-fitting" problem. To alleviate such



(a) Cluster map before the distant-supervised phase.

(b) Cluster map after the distant-supervised phase.

Figure 3: Comparison of the word representations *before* and *after* the emoji-prediction phase.

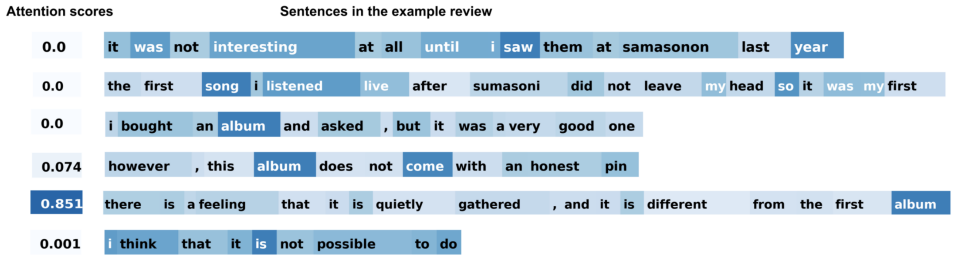
potential concern, we mix up the English labeled reviews and the labeled data in target language that are originally for testing. We randomly select also 2,000 samples from the mixed data as a training set and the remaining samples as a test set. We find the testing results (i.e., the performance of S-Ermes with mixed data in Table 3) are acceptable compared with the previous. It indicates that the"over-fitting" is not the major reason and the architecture of this simplified model is still suitable for the sentiment classification task. The main drawback can then be ascribed to that in cross-lingual setting, the sentiment information learned from the 2,000 labeled English data can not be effectively transferred to the target language purely based on the translated parallel texts. In other words, the sentiment information learned from emoji-prediction phase, which can be reflected in the representation model in both word level and sentence level, is the key to our performance improvement. We conduct two following empirical experiments to further explore such insights, and find how emojis benefit our model to learn the sentiment information.

4.4.2 Representation Learning Efficiency

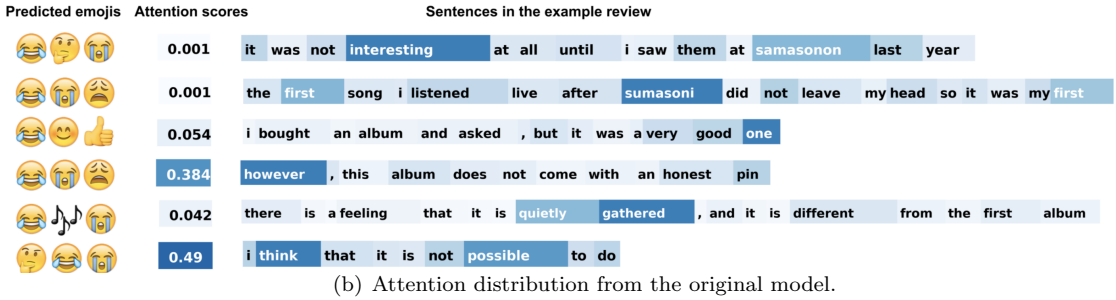
To better understand the sentiment information learned by the emojis prediction, we further conduct an empirical experiment at the word representation level. Recall that after the word representation learning phase, every single word can be represented by a unique vector. In our approach, these word vectors are then fine-tuned through the distant-supervised emoji-prediction phase. Next, we would like to evaluate whether the sentiment information is captured by the new word representations under the effects of emojis. We sample 50 English words with distinct emotional polarity from MPQA[9] subjectivity lexicon based on their frequency in our corpus. These words are manually labeled in positive or negative polarity from MPQA, so they can be regarded as the ground-truth for further evaluating the word representations. We expect an informative representation approach can embed the words with same sentiment polarity closely into the vector space. To better measure and illustrate the similarity, we calculate the similarity score between every two words as the cosine distance of the corresponding representations. Following the *scipy* package in Python[10], the cosine distance between vectors u and v is computed as

[9]http://mpqa.cs.pitt.edu/lexicons/subj_lexicon, retrieved on May 12, 2018.

[10]https://docs.scipy.org/doc/scipy/reference/generated/scipy.spatial.distance.pdist.html, retrieved on May 12, 2018.



(a) Attention distribution from the simplified model.



(b) Attention distribution from the original model.

Figure 4: Case study: Effect of emojis in document comprehension.

$$cosine\_distance = 1 - \frac{u \cdot v}{||u||_2||v||_2},$$

Based on the cosine distances, we perform a hierarchical clustering algorithm [47] and visualize the clustering results in Figure 3(a) and 3(b). The darkness of each cell indicates the similarity between the two words. The darker the cell is, the more similar representations the two words have.

As is illustrated in Figure 3(a), we use the naive representations learned from the word representation phase and can find that the words with different sentiment can not be clearly separated. For example, in the bottom right part, the positive "pleasing" and the negative "picky" are represented similarly in the vector space. This phenomenon indicates that the simple word embeddings learned in this phase are not reliable enough to capture the sentiment information. Just because these words are all usually used to express sentiment, the word representation mechanism represents them similarly. In contrast, in Figure 3(b), we can easily observe two obvious clusters generated by our fine-tuned representation model. In Figure 3(b), the top left corner cluster contains the positive words, while the bottom right corner contains the negative words. Only one positive word "sure" is incorrectly clustered with negative words. By checking the contexts of this word in our corpus, we find it is usually co-used with both positive and negative words, causing the representation model represents it with an ambiguous representation. The correct cluster of nearly all the words indicates that under the effects of emojis, the new word representations capture more useful sentiment information, which can be very informative and beneficial to our further sentiment classification as is shown in Table 3.

4.4.3 Document Comprehension Efficiency

We then explore how emojis can benefit document comprehension. We select a representative case that is incorrectly classified by the simplified model but correctly classified by our model, which can help us study the effect of emojis. This case is selected from the Japanese test samples and we use the translated document for illustration. As is shown in Figure 4, this document consists of six coherent sentences. We use the darkness of the cells to indicate the importance (attention score) of words in each sentence. The darker the word is, the higher its attention score is. We also list the attention score of each sentence in

this document and the top 3 emojis our model predicts for each sentence with the highest probabilities, which may indicate the predicted polarity of each sentence.

Although the whole document indicates the dissatisfaction to the quality of the album, it is not that easy to identify this intent directly from each single sentence. Specifically, the third sentence indicates the expectation of the author for this album, and the fifth sentence describes how the album is different from the first one in the author's view. Both these two sentences can be regarded as a positive sentiment if we consider themselves separately without context, which probably explains why the simplified model fails to classify the whole document correctly. For further explanation, we can turn to Figure 4(a) that demonstrates how the simplified model processes for sentiment classification in document level. Here, the extreme attention is addressed on the fifth sentence. However, as we illustrated above, the fifth sentence may not express the consistent sentiment with the whole document's sentiment. Also, the simplified model tends to focus more on the neural words like "song" or "album" instead of other sentimental words at the sentence level.

On the contrary, due to the incorporation of emojis, our approach works with the proper logic (see Figure 4(b)). To induce the appropriate sentence representation with considering the role of each word in it, this model addresses its attention on the according emotional adjectives, such as "interesting", "not possible", and disjunctive conjunctions such as "however". Thus, it manages to indicate the sentiment of each sentence as we expected, which can be further explained by the predicted emojis on the left. Specifically, 😭 and 😩 predicted with the fourth and sixth sentence indicate the negative intent of the author, while 👍 and 😊 foretold in the third sentence indicate author's opposite attitude. In addition, noticed that 😂 is predicted for the first five sentences. It is reasonable as it is the most used emoji in our corpus and accounts for nearly 10% of the total emoji usage. Although the document contains both positive and negative sentences, our model seems to capture the correct logic of this document by attending to the disjunctive turning at the forth sentence. As Figure 4(b) turns out, our model addresses less attention on the third and fifth sentences, while centering upon the fourth and the last sentences. By comparison, we can find that the emoji-texts bring in additional knowledge to the language comprehension and thus make the attention mechanism more effective.

5 Discussion

So far, we have presented the performance of Ermes on the review data sets and demonstrated the power of emojis in our learning process. Indeed, there are some issues that could potentially affect the effectiveness and efficiency of our approach, and some further discussions are necessary.

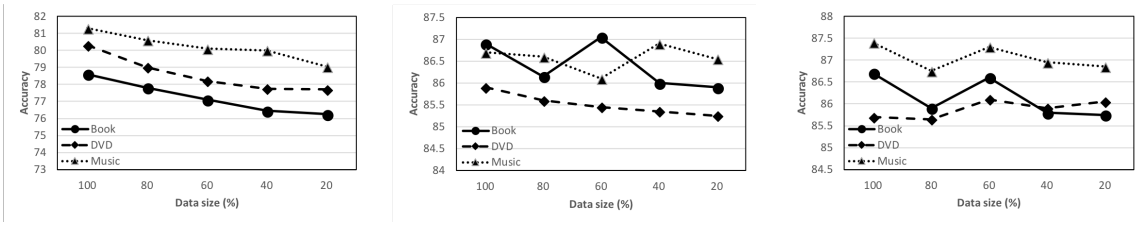
• **Data Volume Sensitivity**. First, as we crawl large-scale Tweets for learning representations, we want to investigate whether and how our approach works well with less volume of data. Furthermore, as we use emoji texts to supply the sentiment knowledge from the 2,000 labeled English reviews, we should explore whether our approach really decreases the demand of the sentiment-labeled data.

• **Generalizability**. Second, although we follow the existing efforts to evaluate on the Amazon review data sets, we should apply our Ermes on other kind of labeled data to demonstrate the generalization of our approach.

We discuss the two issues one by one.

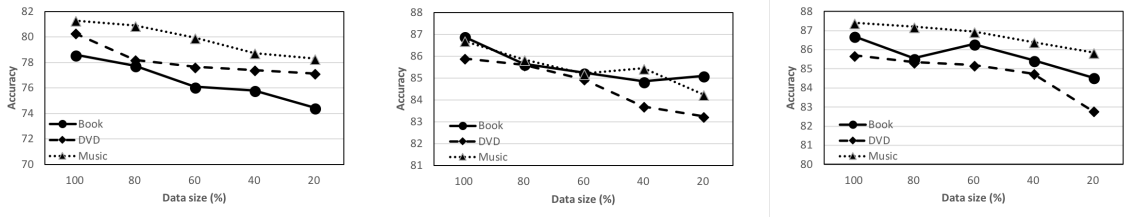
5.1 Data Volume Sensitivity Analysis

First, we investigate the influences of the unlabeled data size. The English-side model can be reusable to any other English-target language pair as we can have it learned already. We only need to scale down the data size of Tweets and emoji Tweets of target languages simultaneously and observe the changes of performance on benchmarks. Detailedly, we use 80%, 60%, 40%, 20% of the collected Tweets to re-train the target-side models in unsupervised and distant-supervised phases and maintain the supervised phase unchangeable. We summarize the results in Figure 5. For the difficult Japanese tasks, when we scale down the unlabeled data, the performance gets slightly worse. Comparing the results of 20% and 100%



(a) Japanese (b) French (c) German

Figure 5: Results obtained by varying the size of unlabeled data.



(a) Japanese (b) French (c) German

Figure 6: Results obtained by varying the size of labeled data.

data, the accuracy differences in three domain tasks are 2.35%, 2.6%, and 2.55%, respectively. For the French and German tasks, the results under different data size settings are more consistent. For example, the accuracy of 100% unlabeled data is only 0.15% higher than the one of 20% data in French music task. Moreover, although we decrease the size of unlabeled data, the performance on French and German fluctuates less than 1%. Most importantly, we find our approach can outperform the existing approaches on all the nine tasks even with the 20% unlabeled data. It indicates that when the target languages have limited unlabeled resources or the researchers have no patience to collect so many Tweets, our approach still works.

Furthermore, as is mentioned before, although the volume of labeled English resources are relatively richer than that of other languages, it is still fairly limited. Therefore, if our model can be robust even with less labeled resources (i.e., less than 2,000 labeled documents in source language), it will be encouraging. To this end, we also scale down these labeled data size by 80%, 60%, 40%, and 20%. As shown in Figure 6, the performance gets slightly worse with the decreasing of the labeled English data in almost all the tasks. Similar to Figure 5, we also notice some minute increase when scaling down the labeled English data on French and German tasks. It further shows that with large-scale weakly labeled data applied in distant-supervised phase, the model relies less on the labeled data. What's more, it is encouraging that our model with 20% labeled data (i.e., 400 English documents) can outperform the existing approaches with the total 2,000 documents on almost all the tasks. Now, we can safely conclude our model is robust to the limited labeled data.

5.2 Generalizability Analysis

Most of the previous cross-lingual sentiment studies [17, 18, 20, 29] used the Amazon review data set for evaluation. For comparisons with these existing work, we also adopt this data set in the main body of this paper. Admittedly, sentiment classification for review data is very valuable, as it can help enterprises capture the views of customers about their products or their competitors. However, sentiment classification for other domains such as social media is also important. Can our approach still work well on this domain? For evaluating the generalizability of our approach, then we apply on a representative

Table 4: The sentiment distribution in crawled labeled English, French, and German Tweets.

Language	Positive	Neutral	Negative
English	7,059 (34.21%)	10,342 (50.13%)	3,231 (15.66%)
French	718 (23.13%)	1,399 (45.07%)	987 (31.80%)
German	1,057 (14.94%)	4,441 (62.80%)	1,573 (22.24%)

Table 5: The accuracy (%) on French and German Tweets.

Language	Ermes	MT-CNN [27]	Uniform Guess
French	69.63	53.52	45.07
German	80.85	65.35	62.80

kind of social media data – Tweets.

Due to their short and informal nature, sentiment classification for Tweets is considered to be a big challenge [48]. As the cross-lingual studies on Tweets are very limited, we only take the recent cross-lingual method (MT-CNN) for comparison. It is proposed by Deriu *et al.* [27] and also relies on large-scale unlabeled Tweets and translation technique. It firstly translates the source language resources into the target language and then trains a sentiment classification for the target language. The training process contains three phases. In the unsupervised phase, it uses raw Tweets to create word embeddings just like us. In the distant-supervised phase, it leverages :) and :(as weak labels and applied a multi-layer convolutional neural network (CNN) to adapt the word embeddings. In the supervised phase, it finally trains the model on manually annotated Tweets. This work and our work have a coverage of French and German, so we use these two languages as target languages for comparison.

Instead of re-training with the large-scale unlabeled Tweets, we use the pre-trained representation models for both approaches[11]. Based on the representation mechanism, we then use the same training data and test data to evaluate the two approaches. As Deriu *et al.* only released the Twitter ID of the labeled Tweets in different languages and some Tweets can not be crawled now, we can not apply our model to the crawled data and then directly compare the results with the reported results in their paper [27]. For fair comparison, we also reproduce their method on the crawled data. We list the sentiment distribution of the final crawled labeled English, French and German in Table 4. According to the distribution, we can get naive accuracy baselines for French and German by uniform guess, i.e., 45.07% for French and 62.80% for German.

Results of the two models are summarized in Table 5. Our approach still performs well on the Tweet task. It outperforms the existing approach by 16.11 on French task and 15.5% on German task, and outperforms the naive baseline by 24.56% on French task and 18.05% on German task. Although we use the same training and test set for both approaches, we are still concerned whether the pre-trained representations can bring in unfairness. More specific, if we use more unlabeled Tweets for representation learning than their approach, our outstanding performance may attribute to the advantage of data size. To answer this question, we refer to [27] to find the data size they used. We find that they use 300M raw Tweets and 60M Tweets containing :(and :) for representation learning. By contrast, We use only a total of 81M raw Tweets and 13.7M emoji-Tweets. What need to mention is that compared to emojis, emoticons are significantly less used in Twitter [49]. It indicates that although they use about 4.3 times weak labeled data more than us, they need to spare more than 4.3 times efforts to collect them. To sum up, using significantly less data, our approach can obviously outperforms the existing approach on cross-lingual sentiment classification for Tweets.

[11]The pre-trained model of Deriu *et al.* is released on the Github: http://github.com/spinningbytes/deep-mlsa, retrieved on May 12, 2018.

6 Implications and Limitations

As we have demonstrated the encouraging performance of our approach, then we want to discuss the implications of our study and some threads to our evaluation.

6.1 Implications

In ubiquitous computing community, user-generated texts such as Tweets have been considered as a special passive "sensor" of users' sentiments and emotions [8]. Compared to the active sensors such as ecological momentary assessment (EMA) [50] and traditional passive sensors such as wearable sensors [51], using texts to infer sentiment is less costly and easier to scale. However, due to the research "inequality" among languages, this convenient "sensor" technique is absent in many non-English languages. Yet the urgent demand in these according non-English countries to it is never absent. For example, it was reported in some non-English countries such as Japan, mental health concerns are quite serious [52], which calls for an efficient approach to sensing and detecting the disorders of users' emotional states. The text-based sentiment analysis exactly serves this need. For one thing, users tend to generate texts frequently in daily life, which lays a solid foundation for us to sense their sentiment instability accurately and timely. For another, text-based sentiment analysis shows its unique advantages in passive and effortless sensing. Considering the lack of text-based sentiment research in these non-English languages and its obvious advantages, our cross-lingual sentiment analysis can have potential effects and the preliminary results may shed new light on this area.

We want to further discuss the insights of emojis, since they exactly play an important role in our approach. As a new ubiquitous language, emojis are widely adopted across the world [23], attracting a lot of researchers to study their usage patterns [23, 30, 53] and non-verbal functions [24, 31, 32]. These empirical findings lead us to move a step further to expand emoji's abilities as a novel "sensor" in the real world. Based on the previous findings that emojis are widely used to express sentiment across languages [23, 24], in this study, we explore this "sensor" ability of emoji from an sentiment probing perspective. More specifically, our study exploits emojis' ubiquitous and emotional characteristic in diverse languages directly considering emojis as weak sentiment labels and shows promising results. Our successful attempt can in turn encourage the researchers in ubiquitous computing and human-computer interaction community to excavate the "sensor" ability of emoji to user profiles, with their abundant findings of emoji usage differences across cultures [23], ages [53], genders [30] and so on. Considering the similar "research inequality" in user profiling [54] (i.e., the existing text-based user profiling studies mainly focus on English users), using ubiquitous emojis to infer the background, gender, age of users can be generalized to diverse languages and thus narrows the current research gap.

6.2 Limitations

Next, we discuss some limitations or threats to validity in our evaluations.

Generalizability. Currently, we have evaluated our cross-lingual sentiment classification approach on Amazon reviews and Tweets. However, even some popular and well evaluated sentiment analysis approaches are demonstrated to have poor performance on data sets with specific domain knowledge such as software engineering tasks [55]. In the future, we plan to apply our approach on more various tasks and further evaluate the generalization.

Tokenizer and Translation Tool. Indeed, our approach relies on the existing tools such as *MeCab* and *Google Translate*. Although these tools are widely adopted in previous work, they still may not perform perfectly and thus can affect our classification performance. As tokenizing and translation are also hot research topics in natural language processing, we can apply alternative effective tools in future work to alleviate this limitation.

Diverse Emoji Usage. Emojis have been demonstrated to have various non-verbal functions, such as expressing sentiment, adjust tones, or express irony. For different usage purposes, emojis sometimes are used in sentiments with opposite sentiments [24]. The habits of emoji usage are influenced by cultures [23],

gender [30], age [53], platforms [56, 57], etc. Although expressing sentiment is the most used function [24], the uncertain and various emoji usage still leads to noise in distant-supervised phase. In future work, we plan to train a classification of emoji usage purposes and filter out the emoji-texts in which emojis are not the proxies of sentiments to further improve the performance.

7 Related Work

We then present the background and literature related to our work.

Prevalence of Emojis. Emojis, also known as ideograms or smileys, can be used as compact expressions of objects, topics, and emotions. Being encoded in Unicode, they have no language barriers and are diffused on the Internet rapidly [23, 30]. The prevalence of emojis has attracted researchers from various research communities such as ubiquitous computing, human-computer interaction, computer-mediated communication, and Web mining [33, 23, 53, 58, 32, 31, 30, 59, 56, 57, 24]. Many research efforts have been devoted to studying their ubiquitous usage across apps [33], across platforms [56, 57], and across countries and languages [23, 30]. The various non-verbal functions of emojis are an important factor of their wide adoption. Emojis were proposed to replace content words, provide the situational and additional emotional information, adjust tone, express intimacy, etc [24, 31, 32]. Especially, expressing sentiment is demonstrated as the most popular intention for using emojis [24], so it is believed that emojis can be benign proxies of representing sentiment [28, 2]. Considering the ubiquitous usage of emojis across languages and their sentiment expression function, we make the first effort to use emojis as weak labels of sentiment to improve the cross-lingual sentiment analysis.

Sentiment Analysis. Sentiment analysis is an important field that studies the opinions, sentiments, evaluations, appraisals, attitudes, and emotions of people [60]. Various advanced sentiment analysis techniques have been proposed in the past a few years. Many well-known sentiment analysis tools simply leverage the polarity of single words to determine the overall sentiment score of text, such as SentiStrength [61] and LIWC [62]. However, these methods are far from sufficient. For example, sometimes sentences without sentiment words can also imply opinions [60]. In subsequent research, more complex features and learning algorithms are employed. Recently, the emerging of deep learning has promoted the research progress. Many researchers attempt to use advanced neural network algorithms to solve the sentiment analysis tasks [63, 64, 65, 1, 66]. Supervised machine-learning or deep-learning based methods usually need a large volume of labeled data to train the model. However, it is time-consuming and error-prone to label sentences manually [2]. To counter the scarcity of labeled data, many researchers use distant-supervised learning, which uses emotional expressions and emoticons as weak sentiment labels [26, 27, 2, 28]. However, the hashtags are too language-specific to be general, and the emoticons have been gradually replaced by increasingly popular emojis [49]. In this study, to alleviate the dependence on large-scale labeled resources, we propose a distant learning framework with emojis as weak labels to learn representations for cross-lingual sentiment analysis.

Cross-Lingual Text Classification. Due to the the imbalanced labeled resources among different languages, there is an urgent need to design algorithms in a cross-lingual view, so as to tackle various text classification tasks such as Web page classification [67], topic categorization [68], and sentiment analysis [29] in different languages. Cross-lingual text classification aims to exploit knowledge in source language (usually refers to English) with relatively sufficient labeled data to assist classifications in target language with limited annotated texts. Due to the discrete characteristic of nature language, directly learning a classification model from one-hot representations of texts is extremely difficult [17]. Thus many researchers divided the learning process into two stages as encoding texts in source and target language into continuous representations, and leveraging these representations for the final classification task [39, 68]. Furthermore, to bridge the linguistic gap between the source and target language, most efforts introduced the translation oracle at different levels to map the representations in each language into a unified space during the representation constructing process [69, 70, 71]. Specifically, the performance heavily relies on the generated pseudo parallel texts from the existing imperfect translation method. In practice, to minimize the representation gap between pseudo parallel corpora in word [72, 39, 69, 19],

sentence [70], and document [71, 29, 73] level, various studies spared no effort to design loss functions and thus induced the models to learn a proper representation. Based on the representations, researchers use only the labeled data from source language to train the classifier for the specific classification task. However, the task-specific information like sentiments which hold language-specific characteristic can not be easily transferred in this way, since these specific usages rarely exist in the machine translation texts of other languages. In this study, instead of using imperfect translation tools to generate parallel texts for a unified representation learning, we introduce language-specific knowledge by training representation learning models for the source and target languages respectively. In specific, we use the easily-accessed emoji texts to incorporate sentiment information in the representation learning process for each language, which effectively capture the implicit sentiment expression based on the diverse emotional emoji usage with corresponding texts.

8 Conclusion

We have presented Ermes, an emoji-powered representation learning framework for cross-lingual document sentiment classification. We apply Word2Vec to large-scale Tweets to learn word representations for both source and target languages, which capture semantic relationships between words. Then based on the word representations, we use an attention-based stacked bi-directional LSTM model and emoji-Tweets to capture sentiment information and create informative sentence representations. Finally, based on the learned representation models, we use labeled English data to train a sentiment classifier for other languages. Our Ermes approach are comprehensively evaluated on various benchmarks and outperforms the existing cross-lingual sentiment analysis methods.

References

[1] Y. Wang, M. Huang, X. Zhu, and L. Zhao, "Attention-based LSTM for aspect-level sentiment classification," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016*, 2016, pp. 606–615.

[2] B. Felbo, A. Mislove, A. Søgaard, I. Rahwan, and S. Lehmann, "Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017*, 2017, pp. 1615–1625.

[3] L. Gong, B. Haines, and H. Wang, "Clustered model adaption for personalized sentiment analysis," in *Proceedings of the 26th International Conference on World Wide Web, WWW 2017*, 2017, pp. 937–946.

[4] P. Rodrigues, I. S. Silva, G. A. R. Barbosa, F. R. dos Santos Coutinho, and F. Mourão, "Beyond the stars: towards a novel sentiment rating to evaluate applications in web stores of mobile apps," in *Proceedings of the 26th International Conference on World Wide Web, WWW 2017*, 2017, pp. 109–117.

[5] G. Balikas, S. Moura, and M. Amini, "Multitask learning for fine-grained twitter sentiment analysis," in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2017*, 2017, pp. 1005–1008.

[6] F. Wu, J. Zhang, Z. Yuan, S. Wu, Y. Huang, and J. Yan, "Sentence-level sentiment classification with weak supervision," in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2017*, 2017, pp. 973–976.

[7] K. C. Herdem, "Reactions: Twitter based mobile application for awareness of friends' emotions," in *The 2012 ACM Conference on Ubiquitous Computing, UbiComp 2012*, 2012, pp. 796–797.

[8] K. Saha, L. Chan, K. de Barbaro, G. D. Abowd, and M. D. Choudhury, "Inferring mood instability on social media by leveraging ecological momentary assessments," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, IMWUT*, vol. 1, no. 3, pp. 95:1–95:27, 2017.

[9] J. Yang, L. A. Adamic, M. S. Ackerman, Z. Wen, and C. Lin, "The way i talk to you: sentiment expression in an organizational context," in *Proceedings of the 2012 International Conference on Human Factors in Computing Systems, CHI 2012*, 2012.

[10] N. Diakopoulos and D. A. Shamma, "Characterizing debate performance via aggregated twitter sentiment," in *Proceedings of the 28th International Conference on Human Factors in Computing Systems, CHI 2010*, 2010, pp. 1195–1198.

[11] M. Gamon, "Sentiment classification on customer feedback data: noisy data, large feature vectors, and the role of linguistic analysis," in *Proceedings of 20th International Conference on Computational Linguistics, COLING 2004*, 2004.

[12] Y. Liu, X. Huang, A. An, and X. Yu, "ARSA: a sentiment-aware model for predicting sales performance using blogs," in *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2007*, 2007, pp. 607–614.

[13] M. McGlohon, N. S. Glance, and Z. Reiter, "Star quality: aggregating reviews to rank products and merchants," in *Proceedings of the Fourth International Conference on Weblogs and Social Media, ICWSM 2010*, 2010.

[14] J. Bollen, H. Mao, and X. Zeng, "Twitter mood predicts the stock market," *Journal of Computational Science*, vol. 2, no. 1, pp. 1–8, 2011.

[15] B. O'Connor, R. Balasubramanyan, B. R. Routledge, and N. A. Smith, "From tweets to polls: linking text sentiment to public opinion time series," in *Proceedings of the Fourth International Conference on Weblogs and Social Media, ICWSM 2010*, 2010.

[16] A. Tumasjan, T. O. Sprenger, P. G. Sandner, and I. M. Welpe, "Predicting elections with twitter: what 140 characters reveal about political sentiment," in *Proceedings of the Fourth International Conference on Weblogs and Social Media, ICWSM 2010*, 2010.

[17] P. Prettenhofer and B. Stein, "Cross-language text classification using structural correspondence learning," in *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, ACL 2010*, 2010, pp. 1118–1127.

[18] M. Xiao and Y. Guo, "Semi-supervised representation learning for cross-lingual text classification," in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013*, 2013, pp. 1465–1475.

[19] A. P. S. Chandar, S. Lauly, H. Larochelle, M. M. Khapra, B. Ravindran, V. C. Raykar, and A. Saha, "An autoencoder approach to learning bilingual word representations," in *Advances in Neural Information Processing Systems 27, NIPS 2014*, 2014, pp. 1853–1861.

[20] H. Pham, T. Luong, and C. D. Manning, "Learning distributed representations for multilingual text sequences," in *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing, VS@NAACL-HLT 2015*, 2015, pp. 88–94.

[21] X. Zhou, X. Wan, and J. Xiao, "Attention-based LSTM network for cross-lingual sentiment classification," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016*, 2016, pp. 247–256.

[22] O. Bojar, R. Chatterjee, C. Federmann, Y. Graham, B. Haddow, S. Huang, M. Huck, P. Koehn, Q. Liu, V. Logacheva, C. Monz, M. Negri, M. Post, R. Rubino, L. Specia, and M. Turchi, "Findings of the 2017 conference on machine translation," in *Proceedings of the Second Conference on Machine Translation, WMT 2017*, 2017, pp. 169–214.

[23] X. Lu, W. Ai, X. Liu, Q. Li, N. Wang, G. Huang, and Q. Mei, "Learning from the ubiquitous language: an empirical analysis of emoji usage of smartphone users," in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp 2016*, 2016, pp. 770–780.

[24] T. Hu, H. Guo, H. Sun, T. T. Nguyen, and J. Luo, "Spice up your chat: the intentions and sentiment effects of using emojis," in *Proceedings of the Eleventh International Conference on Web and Social Media, ICWSM 2017*, 2017, pp. 102–111.

[25] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *Computer Science*, 2013.

[26] K. Liu, W. Li, and M. Guo, "Emoticon smoothed language models for twitter sentiment analysis," in *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2012*, 2012.

[27] J. Deriu, A. Lucchi, V. D. Luca, A. Severyn, S. Müller, M. Cieliebak, T. Hofmann, and M. Jaggi, "Leveraging large amounts of weakly supervised data for multi-language sentiment classification," in *Proceedings of the 26th International Conference on World Wide Web, WWW 2017*, 2017, pp. 1045–1052.

[28] J. Zhao, L. Dong, J. Wu, and K. Xu, "Moodlens: an emoticon-based sentiment analysis system for chinese tweets," in *Proceedings of the 18th ACM International Conference on Knowledge Discovery and Data Mining, KDD 2012*, 2012, pp. 1528–1531.

[29] X. Zhou, X. Wan, and J. Xiao, "Cross-lingual sentiment classification with bilingual document representation learning," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016*, 2016, pp. 1403–1412.

[30] Z. Chen, X. Lu, W. Ai, H. Li, Q. Mei, and X. Liu, "Through a gender lens: learning usage patterns of emojis from large-scale android users," in *Proceedings of the Web Conference 2018, WWW 2018*, 2018, pp. 763–772.

[31] H. Cramer, P. de Juan, and J. R. Tetreault, "Sender-intended functions of emojis in US messaging," in *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services, MobileHCI 2016*, 2016, pp. 504–509.

[32] H. Pohl, C. Domin, and M. Rohs, "Beyond just text: semantic emoji similarity modeling to support expressive communication," *ACM Transactions on Computer-Human Interaction, TOCHI*, vol. 24, no. 1, pp. 6:1–6:42, 2017.

[33] C. Tauch and E. Kanjo, "The roles of emojis in mobile phone notifications," in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp Adjunct 2016*, 2016, pp. 1560–1565.

[34] M. Hermans and B. Schrauwen, "Training and analysing deep recurrent neural networks," *Proceedings of advances in Neural Information Processing Systems*, pp. 190–198, 2013.

[35] S. Hochreiter and J. Schmidhuber, "Long short-term memory." *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[36] S. Hochreiter, "The vanishing gradient problem during learning recurrent neural nets and problem solutions," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 6, no. 2, pp. 107–116, 1998.

[37] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," in *Proceedings of the 30th International Conference on Machine Learning, ICML 2013*, 2013, pp. 1310–1318.

[38] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proceedings of the 3rd International Conference on Learning Representations, ICLR 2014*, 2014.

[39] T. Luong, H. Pham, and C. D. Manning, "Bilingual word representations with monolingual quality in mind," in *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing, VS@NAACL-HLT 2015*, 2015, pp. 151–159.

[40] N. Ljubesic and D. Fiser, "A global analysis of emoji usage," in *Proceedings of the 10th Web as Corpus Workshop, WAC@ACL 2016*, 2016, pp. 82–89.

[41] X. Chen, X. Qiu, C. Zhu, P. Liu, and X. Huang, "Long short-term memory neural networks for chinese word segmentation," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015*, 2015, pp. 1197–1206.

[42] R. Caruana, S. Lawrence, and C. L. Giles, "Overfitting in neural nets: backpropagation, conjugate gradient, and early stopping," in *Proceedings of advances in neural information processing systems 13, NIPS 2000*, 2000, pp. 402–408.

[43] A. Graves, A. Mohamed, and G. E. Hinton, "Speech recognition with deep recurrent neural networks," in *IEEE International conference on acoustics, speech and signal processing, ICASSP 2013*, 2013, pp. 6645–6649.

[44] D. P. Kingma and J. Ba, "Adam: a method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014.

[45] J. Blitzer, R. T. McDonald, and F. Pereira, "Domain adaptation with structural correspondence learning," in *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing, EMNLP 2006*, 2006, pp. 120–128.

[46] Q. V. Le and T. Mikolov, "Distributed representations of sentences and documents," in *Proceedings of the 31th International Conference on Machine Learning, ICML 2014*, 2014, pp. 1188–1196.

[47] Z. Bar-Joseph, D. K. Gifford, and T. S. Jaakkola, "Fast optimal leaf ordering for hierarchical clustering," in *Proceedings of the Ninth International Conference on Intelligent Systems for Molecular Biology, July 21-25, 2001, Copenhagen, Denmark*, 2001, pp. 22–29.

[48] A. Giachanou and F. Crestani, "Like it or not: a survey of twitter sentiment analysis methods," *ACM Computing Surveys*, vol. 49, no. 2, pp. 28:1–28:41, 2016.

[49] U. Pavalanathan and J. Eisenstein, "Emoticons vs. emojis on twitter: a causal inference approach," *CoRR*, vol. abs/1510.08480, 2015.

[50] C. Mihaly and L. Reed, "Validity and reliability of the experience-sampling method," in *Flow and the foundations of positive psychology*. Springer, 2014, pp. 35–54.

[51] P. Adams, M. Rabbi, T. Rahman, M. Matthews, A. Voida, G. Gay, T. Choudhury, and S. Voida, "Towards personal stress informatics: comparing minimally invasive techniques for measuring daily stress in the wild," in *Proceedings of the 8th International Conference on Pervasive Computing Technologies for Healthcare, PervasiveHealth 2014*, 2014, pp. 72–79.

[52] K. Tomoko, "Japanese mental health care in historical context: why did japan become a country with so many psychiatric care beds?" *Social Work*, vol. 52, no. 4, pp. 471–489, 2016.

[53] R. Zhou, J. Hentschel, and N. Kumar, "Goodbye text, hello emoji: mobile communication on wechat in china," in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, CHI 2017*, 2017, pp. 748–759.

[54] M. Ciot, M. Sonderegger, and D. Ruths, "Gender inference of twitter users in non-english contexts," in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013*, 2013, pp. 1136–1145.

[55] B. Lin, P. Sharma, F. Zampetti, G. Bavota, M. D. Penta, M. Lanza, and R. Oliveto, "Sentiment analysis for software engineering: how far can we go?" in *Proceedings of the 40th International Conference on Software Engineering, ICSE 2018*, 2018, p. to appear.

[56] H. J. Miller, J. Thebault-Spieker, S. Chang, I. L. Johnson, L. G. Terveen, and B. J. Hecht, ""blissfully happy" or "ready to fight": varying interpretations of emoji," in *Proceedings of the Tenth International Conference on Web and Social Media, ICWSM 2016*, 2016, pp. 259–268.

[57] H. J. Miller, D. Kluver, J. Thebault-Spieker, L. G. Terveen, and B. J. Hecht, "Understanding emoji ambiguity in context: the role of text in emoji-related miscommunication," in *Proceedings of the Eleventh International Conference on Web and Social Media, ICWSM 2017*, 2017, pp. 152–161.

[58] H. Pohl, D. Stanke, and M. Rohs, "Emojizoom: emoji entry via large overview maps," in *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services, MobileHCI 2016*, 2016, pp. 510–517.

[59] W. Ai, X. Lu, X. Liu, N. Wang, G. Huang, and Q. Mei, "Untangling emoji popularity through semantic embeddings," in *Proceedings of the Eleventh International Conference on Web and Social Media, ICWSM 2017*, 2017, pp. 2–11.

[60] B. Liu, *Sentiment analysis and opinion mining*, ser. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers, 2012.

[61] M. Thelwall, K. Buckley, G. Paltoglou, D. Cai, and A. Kappas, "Sentiment in short strength detection informal text," *JASIST*, vol. 61, no. 12, pp. 2544–2558, 2010.

[62] J. W. Pennebaker, L. E. Francis, and R. J. Booth, "Linguistic inquiry and word count: Liwc," *Lawrence Erlbaum Associates Mahwah Nj*, 1999.

[63] J. Wang, L. Yu, K. R. Lai, and X. Zhang, "Dimensional sentiment analysis using a regional CNN-LSTM model," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016*, 2016.

[64] Q. Qian, M. Huang, J. Lei, and X. Zhu, "Linguistically regularized LSTM for sentiment classification," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017*, 2017, pp. 1679–1689.

[65] M. Yang, W. Tu, J. Wang, F. Xu, and X. Chen, "Attention based LSTM for target dependent sentiment classification," in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, AAAI 2017*, 2017, pp. 5013–5014.

[66] A. Salekin, Z. Chen, M. Y. Ahmed, J. Lach, D. Spruijt-Metz, K. de la Haye, B. Bell, and J. A. Stankovic, "Distant emotion recognition," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, IMWUT*, vol. 1, no. 3, pp. 96:1–96:25, 2017.

[67] X. Ling, G. Xue, W. Dai, Y. Jiang, Q. Yang, and Y. Yu, "Can chinese web pages be classified with english data source?" in *Proceedings of the 17th International Conference on World Wide Web, WWW 2008*, 2008, pp. 969–978.

[68] J. T. Zhou, S. J. Pan, I. W. Tsang, and S. Ho, "Transfer learning for cross-language text categorization through active correspondences construction," in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, AAAI 2016*, 2016, pp. 2400–2406.

[69] H. Zhou, L. Chen, F. Shi, and D. Huang, "Learning bilingual sentiment word embeddings for cross-language sentiment classification," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015*, 2015, pp. 430–440.

[70] K. M. Hermann and P. Blunsom, "Multilingual models for compositional distributed semantics," in *Proceedings of the 52nd annual meeting of the association for computational linguistics, ACL 2014*, 2014, pp. 58–68.

[71] M. Faruqui and C. Dyer, "Improving vector space word representations using multilingual correlation," in *Proceedings of the 14th conference of the european chapter of the association for computational linguistics, EACL 2014*, 2014, pp. 462–471.

[72] I. Vulic and M. Moens, "Bilingual word embeddings from non-parallel document-aligned data applied to bilingual lexicon induction," in *Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international Joint Conference on natural language processing of the asian federation of natural language processing, ACL 2015*, 2015, pp. 719–725.

[73] D. Tang, B. Qin, and T. Liu, "Document modeling with gated recurrent neural network for sentiment classification," in *Proceedings of the 2015 conference on empirical methods in natural language processing, EMNLP 2015*, 2015, pp. 1422–1432.