# Social Media Sentiment Analysis:<br>Multilingual Methodology and Monolingual Application

# (ソーシャルメディアにおける感情分析:<br>多言語の方法論と単言語の応用)

by

Yujie LU

September, 2017

A thesis submitted to
the Graduate School of Enviroment and Information Sciences,
Yokohama National University
for the Degree of Doctor of Philosophy in Engineering

Principal Advisor: Professor Tatsunori MORI

# Abstract

The surge of social media use, such as Twitter, introduces new opportunities for understanding and gauging public mood across different cultures. However, the diversity of expression in social media presents a considerable challenge to this task of opinion mining, given the limited accuracy of sentiment classification and a lack of intercultural comparisons. Previous Twitter sentiment corpora have only global polarities attached to them, which prevents deeper investigation of the mechanism underlying the expression of feelings in social media, especially the role and influence of rhetorical phenomena.

To this end, we construct an annotated multilingual corpus for deeper sentiment understanding (the MDSU corpus, for short) that encompasses three languages (English, Japanese, and Chinese) and four international topics (iPhone 6, Windows 8, Vladimir Putin, and Scottish Independence); our corpus incorporates 5422 tweets. During the construction, we propose a novel annotation scheme that embodies the idea of separating emotional signals and rhetorical context, which, in addition to global polarity, identifies rhetoric devices, emotional signals, degree modifiers, and subtopics. Besides, to address low inter-annotator agreement in previous corpora, we propose a pivot dataset comparison method to effectively improve the agreement rate. With manually annotated rich information, our corpus can serve as a valuable resource for the development and evaluation of automated sentiment classification, intercultural comparison, rhetoric detection, etc.

Based on observations and analysis of the MDSU corpus, we present three key conclusions. First, languages differ in terms of emotional signals and rhetoric devices, and the idea that cultures have different opinions regarding the same objects is reconfirmed. Second, each rhetoric device maintains its own characteristics, influences global polarity in its own way, and has an inherent structure that helps to model the sentiment that it represents. Third, the models of the expression of feelings in different languages are rather similar, suggesting the possibility of unifying multilingual opinion mining at the sentiment level.

The multilinguality of social media leads to the urgent need for multilingual sentiment analysis (MSA) to unveil cultural differences. The lack of benchmark datasets that support the evaluation to the methods of MSA curbs the development of it. Fortunately, the MDSU corpus can be a perfect training/test dataset. So far, traditional methods resorted to machine translation—translating texts in other languages to English, and then adopt the methods once worked in English. However, this paradigm is conditioned by the quality of machine translation. In this thesis, we propose a new deep learning paradigm to assimilate language differences for MSA. We first pre-train monolingual word embeddings separately, then map word embeddings in different spaces into a shared embedding space, and finally train parameter-sharing deep neural networks for MSA. The experimental results show that our paradigm is effective. Especially, our convolutional neural network model using transformed word embeddings outperforms a strong baseline by around 2.3% in term of classification accuracy.

Last but not least, we apply monolingual sentiment analysis to unfolding public mood on social issues from microblogging for sector index prediction. We first train a low-dimensional support vector machine classifier using surrounding information for Twitter sentiment classification. Then, we generate public mood time series by aggregating tweet-level weighted daily mood (WDM) based on the sentiment classification results. Further, we evaluate our WDM time series against the real stock index during two kinds of time periods (i.e., fluctuating and monotonous periods) by both static cross-correlation coefficient and dynamic vector auto-regression. The experiments on “food safety” issue show that the proposed WDM method outperforms the word-level baseline in predicting stock movement, especially during the fluctuating periods.

# Acknowledgements

First and foremost, I would like to express my deepest gratitude to my supervisor, Professor Tatsunori Mori. His constant encouragements through the last four years of my Ph.D. studies give me courage to face difficulties. Without his careful instruction and full support, this doctoral thesis would have been impossible.

I also would like to thank Professor Tomoharu Nagao, Professor Naoyoshi Tamura, Associate Professor Takashi Tomii, and Lecturer Shinichi Shirakawa for serving on my thesis committee. Their constructive feedback and comments are extremely helpful in improving the thesis manuscript.

I am greatly indebted to the past and present members of Mori Laboratory for their kind and selfless help. Especially, I would like to thank Mr. Shigeru Ikeda for his professional technical and clerical support, Mr. Hideyuki Shibuki for his insightful research advice, and Mr. Kotaro Sakamoto for his stimulating discussions. The days I spent together with them are full of joy.

I pay my regards to all of those who helped me in any respect during the completion of this thesis. Special thanks to China Scholarship Council, Yokohama National University (YNU), and the Graduate School of Environment and Information Sciences, YNU for giving me various forms of financial assistance. Many thanks to the PACLIC 29 Organizing Committee/PACLIC Steering Committee, and the Association for Natural Language Processing for allowing me to use my previously published papers as parts of the thesis.

Finally, I would like to thank my family for their unconditional love. My parents have spared no effort to support me all the time, and my grandparents' expectations from my childhood have always motivated me.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

## 1.1 Background and Motivation

According to the following definition from Liu [35], sentiment analysis aims to analyze people's attitudes toward certain given objects.

> *Sentiment analysis, also called opinion mining, is the field of study that analyzes people's opinions, sentiments, evaluations, appraisals, attitudes, and emotions towards entities such as products, services, organizations, individuals, issues, events, topics, and their attributes.*

There are many similar terms for sentiment analysis used by different researchers, such as opinion mining, opinion extraction, sentiment mining, affect analysis, emotion analysis, subjectivity analysis, and review mining. Although these terms all fall under the umbrella of sentiment analysis or opinion mining, they differ slightly from each other in emphasis. In this thesis, we choose to use the term "sentiment analysis" since our research focuses on understanding the expression of feelings.

Sentiment (or opinion, evaluation, appraisal, attitude, and emotion) is the main topic of this thesis. Why should we care so much about sentiment analysis? There are a couple of reasons.

First, everyday decision-making processes are strongly related to others' opinions (i.e., the word-of-mouth effect). Our perceptions of the world are vulnerable to our attitudes toward it. The opinions of other people often have a great influence on our attitudes to objects in daily life, resulting in changes in our behaviors.

The good news is that the Internet offers us abundant online opinions nowadays, but their amount is too great for users to digest. Therefore, it is essential to find an automatic way to analyze/summarize large-scale opinion texts, in which sentiment analysis plays a central role.

Besides, many applications are underlain by sentiment analysis at different levels, such as investigation of consumer reactions to products, stock index/price prediction, public polling, and election forecasting. Some only need the collective sentiment of each text collection (collective sentiment will be discussed in Chapter 4), while others need document-level or aspect-level sentiment analysis. This thesis mainly concerns document-level sentiment analysis (i.e., classifying the global polarity of a tweet).

Last but not least, the research field of artificial intelligence has recently gained attention in both academia and industry. As a representative technological achievement of artificial intelligence, Chatbot has been in use for many years. We can talk to Siri/Cortana anytime we want. Researchers have been trying to provide them with reasoning through large-scale knowledge bases and smart algorithms, and although there is room for improvement, this has worked well to some extent. However, one problem is that these chatbots do not appear to have emotions. In the near future, a robot will not only know what something is, but also should hold a feeling for it. To this end, sentiment analysis can be a key supporting technology.

## 1.2 Research Object—Why Twitter?

As a research field of natural language processing (NLP), the history of sentiment analysis is not as long as that of machine learning, question answering, or other fields. In fact, there was little work focused on sentiment analysis before 2000. Since then, it has increased in significance and established itself as a major area of NLP on which many existing applications are based. An important factor contributing to this is the appearance of opinion-related user-generated content (UGC), such as shopping websites, forum discussions, and blogs. Researchers have done much work on movie reviews, product reviews, etc.

Nowadays, social media such as Twitter and Facebook have further accelerated this progress. Social media provide us with a tremendous number of easy-to-access opinionated texts, which brings both new opportunities and challenges. In this work, we will use texts from Twitter (or tweets) as our research object for the following reasons.

(1) Strong influence. According to the *New York Times*, during the 2016 United States presidential election, Twitter ruled among the various social media. On the day of the 2016 U.S. presidential election, Twitter proved to be the largest source of breaking news, with 40 million election-related tweets sent by 10 p.m. that day.[1] Since then, U.S. President Donald J. Trump has been using Twitter as a means of communicating with the public.

(2) Large scale. Twitter can provide a large amount of data on various topics (as of 2016, Twitter had more than 319 million monthly active users[2]), making it an effective way to access people's opinions on almost all subjects, including public figures, products, and events.

(3) Easy accessibility. While the amount of data is important, its availability is also crucial. Twitter is impressive in terms of the amount of data it provides, but more importantly, its well-designed RESTful API[3] makes it easy and free for researchers to access those messages.

(4) Timeliness. In addition to quantity, speed also matters. Besides RESTful API, Twitter offers Streaming API,[4] which allows users to obtain tweets in real time. Unlike Restful API, which passively responds to developers' access requests, Streaming API can actively push related messages to developers.

(5) Flexible Expression

[1] https://www.nytimes.com/2016/11/09/technology/for-election-day-chatter-twitter-ruled-social-media.html?_r=0

[2] https://about.twitter.com/company

[3] https://dev.twitter.com/rest/public

[4] https://dev.twitter.com/streaming/overview

- Tweets are constrained to be no more than 140 characters, which causes users to post them in a casual way. Informal expressions, including slang, acronyms, spelling errors, hashtags, emoticons, Unicode emojis, letter-repeating words, and all-caps words, are ubiquitous in tweets.
- Short does not mean simple. In addition to the informality of expression, tweets display an abundance of special language phenomena, such as rhetorical context, discourse context, whole-part context, and temporal context. These linguistic devices are commonly used to express users' feelings.

(6) Multilinguality. Social media, for the first time in history, can provide us with multilingual opinionated texts. Twitter now supports more than 40 languages[5]. Through the effective use of these multilingual opinions, we can carry out macro-perspective cultural comparisons that were previously very time-consuming and costly.

Of the afore-mentioned points, (1)–(4) are advantages of Twitter that make it an excellent data source for researchers in social media; (5) poses many new challenges to the research field of NLP; and (6) has allowed us to put sentiment analysis in a multilingual setting. These features make tweets a special research object for NLP. Note that although we use tweets as the source data for this thesis, the proposed methods and main results could be extended to other data sources as well.

## 1.3 Position and Outline

Depending on the number of languages processed, sentiment analysis can be divided into two categories: monolingual sentiment analysis (one language) and multilingual sentiment analysis (two or more languages). Since the methods of monolingual sentiment analysis in social media have been much discussed in previous work, we shall focus on multilingual sentiment analysis (hereafter referred to as MSA) in this thesis (see Chapter 5).

[5] https://about.twitter.com/company

The application part of this thesis is carried out on only one language (i.e., Mandarin Chinese), and the proposed methodology of MSA can be applied to this language. However, given that the performance of MSA with a given language is low compared to the state-of-the-art monolingual sentiment analysis, we will continue to employ customized monolingual methods in the application part (see Chapter 6). Although it is a monolingual application, the same concepts can be extended to other languages.

In this thesis, we first construct a fine-grained annotated corpus for deeper sentiment understanding in a multilingual setting (denoted as the MDSU corpus) to investigate the main factors that may influence tweet-level sentiment, then propose a novel deep learning paradigm for MSA developed on the MDSU corpus to achieve better MSA, and finally introduce a monolingual application for stock index prediction using sentiment analysis technology.

The thesis is organized as follows. In Chapter 2, we present related work on sentiment analysis, introduce the technology used in the deep learning methodology for NLP, and summarize the main contributions of our work. In Chapter 3, we detail the process of the construction of the MDSU corpus. In particular, we describe the annotation scheme that separates emotional signals and rhetorical context and the pivot dataset comparison method (denoted the PDC method) used to improve inter-annotator agreement. In Chapter 4, we perform various analyses on the constructed MDSU corpus to reveal the basic principles behind the expression of feelings in social media. In Chapter 5, we propose a novel deep learning paradigm for MSA. We first unify separately pre-trained monolingual word embedding spaces, and then train parameter-sharing deep learning methods for MSA. In Chapter 6, we apply sentiment analysis technology to social media to predict the sector stock index. We design a weighted daily mood (WDM) time series, and evaluate its predictive power in both fluctuating and monotonic periods. Lastly, we state the conclusions of the study, point out its deficiencies, and discuss future work in Chapter 7.

# Chapter 2

# Sentiment Analysis and Related Technology

In this chapter, we first summarize the development of sentiment analysis, including traditional sentiment analysis and Twitter sentiment analysis (both machine learning methods and deep learning methods). This related work acts as an overall summary of sentiment analysis. Chapters 3, 5, and 6 will discuss the related work for each task in detail. In particular, MSA will be introduced in Section 5.2. After the related work, we introduce the basic technical components that we will use in deep learning methods.

## 2.1 Sentiment Analysis

### 2.1.1 Traditional Sentiment Analysis

Pang et al. [57] and Turney et al. [81] are generally regarded as founding the research area of sentiment analysis. Their two works represent the two main methodologies of sentiment analysis in its early stages, supervised methods and unsupervised methods. Pang et al. [57] fed machine learning methods, including support vector machine (SVM), maximum entropy (ME), and naive Bayes (NB), with features such as n-gram and parts of speech to classify the polarity of texts. On the other hand, Turney et al. [81] calculated the comprehensive polarity of a text by summing up the similarity between the keywords in the text and the seed words, which is known as the SO-PMI algorithm.

It soon became apparent that supervised methods (i.e., machine learning methods) stood out in document-level sentiment analysis. Compared with unsupervised methods counting and aggregating positive and negative words in different ways, machine learning methods generally obtained higher accuracy. Especially, the best accuracies on both the IMDB dataset and the polarity dataset were obtained by SVM classifiers [76].

However, this does not mean that unsupervised methods have no advantages. In fact, compositional models based on the "principle of compositionality" have been found promising. Compositional models hypothesize that the polarity of a sentence is a function of the polarities of its parts [49, 31].

The objects of traditional sentiment analysis include movie reviews and product reviews. Its purpose is not limited to the classification of the global polarity, but involves such tasks as aspect-value pair detection and opinion summarization. Broader overviews of traditional sentiment analysis are presented in [56] and [35].

### 2.1.2 Twitter Sentiment Analysis

Recent studies of sentiment analysis have focused on social media, especially tweets. Machine learning methods have constituted the main methodology for Twitter sentiment analysis. Researchers have tried different features in order to improve the performance of their systems. The SemEval Task reports [51, 61] also pointed out that participants leveraged various features depending heavily on sentiment lexicons and obtained a best accuracy of around 70% in a 3-way setting[1].

As an early attempt, Go et al. [20] annotated a noisy training set based on emoticons in tweets and carried out experiments analogous to those that Pang et al. [57] performed on movie reviews. Three machine learning methods using bag-of-words features, including NB, ME, and SVM, were tested, and they showed that an SVM classier fed with binarized unigram features achieved the best accuracy. This method is strong though simple, and has been proven to be a hard baseline

[1] The global polarity of a tweet can be one of three types, positive, negative, and neutral.

by many follow-up studies [1, 39].

Besides bag-of-words features, Xie et al. [92] proposed a set of *weibo*[2]-specific features, such as the number of emoticons, for an SVM classifier, and achieved an accuracy of around 67% in a 3-way setting. Wang and Li [2] proposed three-layered features that aggregate synonyms and highly-related words for an SVM classifier to help reduce feature dimensions and indicated that it performed better than SVM classifiers using n-gram and POS tags.

Agarwal et al. [1] proposed a Partial Tree kernel method based on their abstraction tree structure. They compared the method with the hard baseline SVM using unigram and a heavy feature engineering method. The results showed that the tree kernel method performed better than the other two methods. The authors also performed a feature analysis, which showed that word-polarity-related features (or polarity words) are the most important features.

Mukkherjee and Bhattacharyya [50] proposed a discourse-based bag-of-words model that takes advantage of lightweight discourse relations for Twitter sentiment analysis. They considered discourse relations such as connectives and conditionals, and semantic operators such as models and negations. The results showed that their method performed better than a basic bag-of-words models without using discourse information.

Xiang and Zhou [91] built a topic-based sentiment mixture model. They first generated topic distributions using latent Dirichlet allocation (LDA) , then trained topic-specific sentiment classifiers for each topic clustering, and the mixture model finally determined the sentiment class of a tweet. According to their report, their method achieved a higher performance than the top system in the task of Sentiment Analysis in Twitter in SemEval-2013, with an averaged F score at 71.2%.

### 2.1.3 Deep Learning Method

Due to the prevalence of deep learning in recent years, many different network structures have been advanced for sentiment analysis. Here we summarize these

[2] http://weibo.com

studies.

Socher et al. [73] and Kim [30] introduced their attempts to use recursive neural network and convolutional neural network (CNN) methods, respectively, for sentence-level sentiment classification, achieving rather impressive results.

Socher et al. [73] proposed a recursive neural tensor network (RNTN) to realize a compositional model using the Stanford Sentiment Treebank corpus. It outperformed many other baselines in both sentence-level classification and phrase-level prediction, and was proved to have the ability to represent the effects of negation.

However, the disadvantage of recursive neural networks is that they require an external parse tree, which is difficult to obtain. Kim [30] proposed a CNN method using static/non-static word vectors for sentence-level classification tasks. He tested the CNN on a couple of sentiment-related datasets and concluded that it improved the performance on many of those tasks.

Later, researchers began to apply other deep learning methods to Twitter sentiment analysis.

Kalchbrenner et al. [29] proposed the Dynamic Convolutional Neutral Network (DCNN), which they tested in four different experiments. The results showed that the DCNN performed well in both sentiment classification of traditional text (e.g., movie reviews) and in Twitter sentiment predictions on the STS corpus (i.e., the Stanford Twitter Sentiment corpus).

Wang et al. [88] proposed a long short-term memory (LSTM) recurrent network. According to their report, their LSTM method outperformed most data-driven approaches and feature-engineering approaches on the STS corpus. They also reported that their models are able to capture the special functions of words (e.g., negation), and to distinguish words with opposite polarity.

Severyn and Moschitti [69] explored deep CNNs. Their network is quite similar to that of Kim [30], but they pre-trained word embeddings on an unsupervised corpus and further tuned them on another supervised corpus. This made their learning start from a good point, and the results showed that their method achieved accuracies that could rank in the first two positions in Semeval-2015 Task 10.

## 2.2 Word Vectorization

Traditionally, words in NLP tasks (corresponding to concepts/objects in the real world) are treated as discrete symbols.[3] For example, 'cat' may be represented as *ID135* and 'dog' as *ID246*. As we can see, these encodings are arbitrary and provide no useful information regarding the relationships (e.g., similarity) between the individual symbols [78]. Moreover, representing words as discrete IDs leads to data sparsity (e.g., one-shot representation), and usually means that we need more data to successfully train statistical models. This problem is even more severe in the case of tweets since they contain more infrequent words [67].

To overcome the curse of dimensionality, low-dimensional distributed representations for words have been proposed [65, 7], and have become extremely successful with the popularity of deep learning. Distributed representations depend in some way on the *Distributional Hypothesis*, which claims $Context \approx Meaning$ (i.e., words that occur in similar contexts tend to have similar meanings). In distributed representations, similar words are close in the vector space, which makes model generalization to new patterns easier and estimation more robust [45].[4] Distributed representations are also known as word embeddings, or word vectors.[5]

Word embeddings obtained from local contexts can capture syntactic and semantic relations between words. It is found that the regularities are observed as constant vector offsets between pairs of words sharing a particular relationship. As an example, for the singular/plural relation, we observe that $X_{apple} - X_{apples} \approx X_{car} - X_{cars}$. More impressively, this also applies to semantic relations. In Figure 1, the vector offsets for the two word pairs illustrate the gender relation. If we inquire "$King - Man + Woman = ?$," then we will get "Queen" by simple vector computation.

[3] For image and audio processing systems, raw pixels or audio spectrograms can be directly used as inputs.

[4] There are two categories of methods to obtain distributed representations: count-based methods (e.g., latent semantic analysis) and predictive methods (e.g., neural probabilistic language models). In practice, the latter seems to perform better.

[5] In this thesis, we mainly use the term 'word embeddings' in the following.

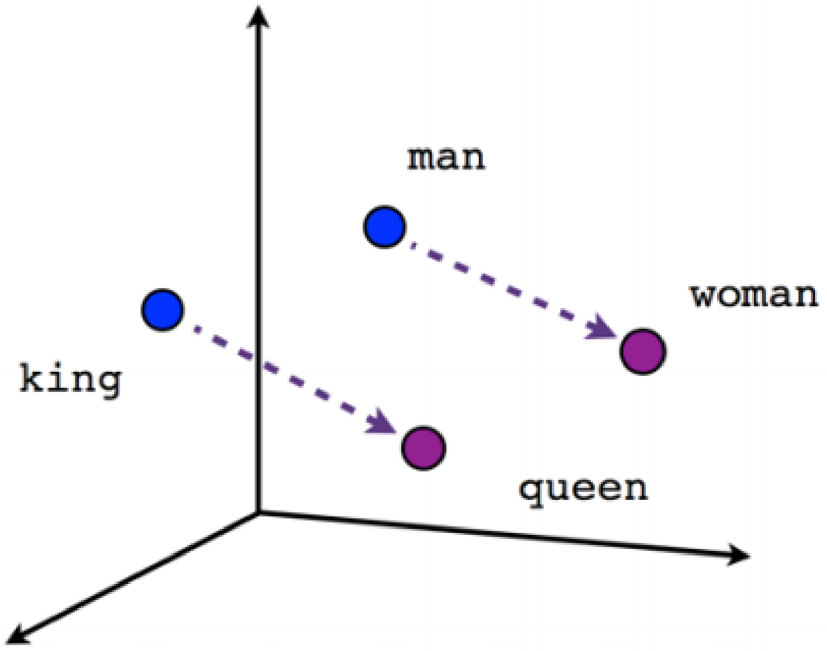


Figure 1: Illustration of Gender Relations Represented by Word Embeddings

Mikolov et al. [47] were not the first to use continuous vector representations of words, but they did show how to reduce the computational complexity of learning such representations. They proposed two basic architectures to training such word embeddings, which are the Continuous Bag-of-Words (CBOW) model and the Skip-gram model.

Figure 2 [47] shows their architectures. The CBOW model takes the context as $w_t$ (i.e., the weighted summation of the word vectors around $w_t$) as input to fit $w_t$, while the Skip-gram model takes the word vector $w_t$ as input to predict the context of $w_t$ (i.e., to minimize the summed prediction error across all context words) [47].

Here, we explore the Skip-gram model further. The objective function for the Skip-gram model is given by Eq. (2–1).

$$J(\theta) = -\frac{1}{T}\sum_{t=1}^{T}\sum_{\substack{-m\leq j\leq m \\ j\neq 0}} log\ p(w_{t+j}|w_t;\theta) \tag{2–1}$$

where $T$ is the size of the corpus and $m$ is the window size of the context. A popular probability measure for $p(w_{t+i}|w_t)$ has been the softmax function:

$$p(w_O|w_I) = \frac{exp({\mathbf{v}'_{w_O}}^T \mathbf{v}_{w_I})}{\sum_{j'=1}^{V} exp({\mathbf{v}'_{w'_j}}^T \mathbf{v}_{w_I})} \tag{2–2}$$

where $\mathbf{v}_{w_I}$ and $\mathbf{v}'_{w_O}/\mathbf{v}'_{w'_j}$ are the input and output vector representations of $w$, respectively, and $V$ is the size of the vocabulary.

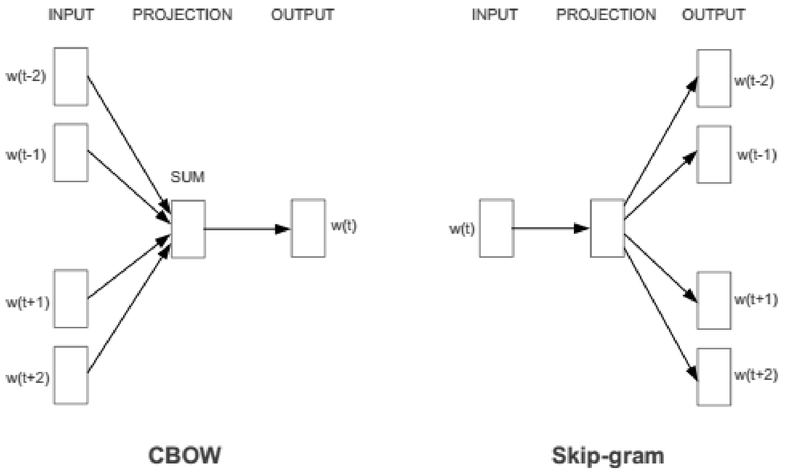


Figure 2: CBOW and Skip-gram Architectures

In fact, Eq. (2–2) is impractical because the cost of computing $\nabla log\ p(w_{t+j}|w_t)$ is proportional to $V$, which is often large ($10^5$–$10^7$). To speed up the training process, two major strategies – hierarchical softmax and negative sampling – have been devised [46]. Negative sampling is very straightforward. It samples a few words from all the words in a vocabulary,[6] and only updates the sampled words rather than all the words. Hierarchical softmax uses a binary Huffman tree to represent all the words in the vocabulary, so it only needs $\lceil log(V) \rceil$ units of $\mathbf{v}'_{n(w,j)}$ ($n(w,j)$ means the $j$-th unit on the path from root to $w$, so $\mathbf{v}'_{n(w,j)}$ is the vector representationo of the unit $n(w,j)$.) instead of $V$ columns of $\mathbf{v}'_{w'_j}$ in terms of the number of parameters between the hidden layer and the output layer.

Both architectures have implemented by *Word2Vec*.[7] Beside *Word2Vec*, many other tools, such as FastText,[8] can generate word embeddings by variants of these two architectures. In this study we use FastText to train our customized word

[6] Apparently, the output word $w_O$ is kept in the sample.

[7] https://github.com/dav/word2vec

[8] https://github.com/facebookresearch/fastText

embeddings. The trained vectors can be used both as an end in themselves and as a representational basis for downstream NLP tasks. In this thesis, the trained word embeddings are the input to our deep learning methods for sentiment analysis.

Since the training of word embeddings is an unsupervised task, there is no standard way to objectively measure the quality of the training result. One possible way is to evaluate the word embeddings on word similarity benchmark datasets; another more general way is to evaluate them on end applications.

## 2.3 Document Modelling

In the last section, we described the distributed presentations of words. In this section, we continue to discuss how we can present documents (e.g., tweets) using word embeddings.

Document vectorization converts text content into a numeric vector representation that can be utilized as a feature representation and then be used to train a machine learning model. Since sentiment analysis is a classification task, we need to transform a tweet into a vector (i.e., a list of features).

The bag-of-words (BoW) model represents a piece of text (such as, a sentence, a paragraph or a document) as a vector of word features. Traditional BoW methods use word features such as tf (term frequency), tf-idf (term frequency-inverse document frequency), binarized tf, and different variants of these [41]. Although the BoW model is surprisingly effective in many classification tasks, one of its problems is data sparseness.

The development of vectorized word representation (as depicted in Section 2.2) has led to more powerful continuous vector representations of documents. One such network is the neural bag-of-words (NBoW) model. For a text $X$, a hidden vector $z$ obtained by averaging of the input word vectors is used to represent $X$. This is actually a linear combination of the binarized tf vectors and the input word vectors [70].

Besides the NBoW model, researchers have proposed more sophisticated networks to model documents. The layers in CNNs interleave convolutional layers and pooling layers. Filters at lower layers extract n-grams at every position in the text, and filters at higher layers can capture syntactic or semantic relations between non-continuous phrases that are far apart in the text [29]. The top fully-connected layer represents the document. Convolutional and pooling architectures allow us to encode arbitrarily large items as fixed-size vectors that capture their most salient features, but they achieve this by sacrificing most of the structural information [28].

The recurrent neural network (RNN) is mainly used a language model, but it may also be viewed as a way to model a document with a linear structure. The layer computed at the last word represents that document [29]. Moreover, recursive networks allow the use of trees and preserve the structural information [28].

In Section 5, we will use CNN and RNN to model our tweets for multilingual sentiment classification.

## 2.4 Contributions of this Study

Here, we list the main contributions of our work.

(1) We have constructed the first comprehensive multilingual corpus that deals with rhetorical phenomena for sentiment analysis in social media. The MDSU corpus can either be an ideal testbed for measuring the effectiveness of any sentiment analysis method or serve as training data to experiment with new methods for sentiment analysis.

(2) During the construction process, we proposed a novel annotation scheme embodying the idea of separating emotional signals and rhetorical context, which, apart from global polarity, identifies key components, including rhetoric devices, emotional signals, degree modifiers, and subtopics. Moreover, in order to alleviate the common issue of low inter-annotator agreement in previous corpora, we propose the PDC method to effectively improve the agreement rate.

(3) Based on the analyses of the MDSU corpus, we find that, although languages differ in terms of the use of components (e.g., emotional signals and rhetorical devices) and cultures have different opinions on the same subjects, the models of expression of feelings in different languages are most likely similar, suggesting the possibility of unifying multilingual opinion mining at the sentiment level.

(4) We have proposed a new deep learning paradigm to assimilate the differences between languages for MSA. We first pre-train monolingual word embeddings separately, then map word embeddings in different spaces into a shared embedding space, and then finally train parameter-sharing deep neural networks for MSA. Our CNN model using mapped word embeddings outperforms a state-of-the-art baseline by around 2.3% in term of classification accuracy, showing the effctiveness of our methodology.

(5) We also have investigated how public mood from microblogging on a certain social issue relates to the stock movement of its relevant sector. The evaluations show that our WDM time series has a potential predictive ability for the sector stock index, and that it performs better during fluctuating periods than monotonic periods.

# Chapter 3

# Construction of a Multilingual Annotated Corpus for Deeper Sentiment Understanding in Social Media

## 3.1 Introduction

A vast amount of user-generated content has been created from the prevalence of social media applications, such as Twitter. Here, users post opinions in real time on various topics including products, public figures, and events. The resulting large-scale dataset provides researchers an unprecedented opportunity to leverage social media for different types of scientific studies [35]. Many useful applications have been proposed thus far, such as investigating consumer reaction to products of a company [26], understanding the popularity of political parties and candidates among voters for forecasting election results [79], polling public opinion on social events [54], responding to terrorism according to social emotion [12], and predicting stock price movements [37].

Although there has been some progress in sentiment analysis for social media on which the above applications have been based, two key challenges remain. First, the diverse nature of social media, with its subtle forms of expression, makes it difficult to study sentiment analysis. Recent studies on SemEval datasets [51, 15, 91] have shown that the highest accuracy of Twitter sentiment analysis is approximately 70%, while the same studies applied to traditional text-based datasets have attained 88.3% accuracy on the IMDB dataset and 93.7% accuracy on the polarity dataset [76].

Second, although social media generates a significant amount of multilingual opinions, available to us for the first time, few studies have been conducted on the comparison of cultural differences among these opinions. Balahur and Turchi [3] discussed the implementation of sentiment analysis on multiple languages by simply using machine translation. Further, Volkova, Wilson, and Yarowsky [84] showed how the use of gender information affects sentiment classification in different languages; however, neither of these studies considered the cultural differences in the same opinion targets (i.e., evaluation objects).

To tackle the first challenge, we need to understand the key differences between "tweet text" (i.e., text snippets taken from Twitter data) and traditional text, the latter including examples such as newswire feeds and product reviews. Traditional text usually has relatively explicit subjective expressions, while tweets are expressed in a more flexible and casual way; therefore, the sentiment that tweets contain may be implicit and subtle, as shown in example tweet (1) below.

**(1) Wow, with #iPhone6, you can send a message just by talking! In any voice you like. So can my mom's old rotary dial.**

In the first two sentences of the above example, the author is praising iPhone 6, whereas in the third sentence, the author turns to criticism by comparing it with something "old." Overall, this is a sarcastic tweet that strengthens the sense of looking down upon iPhone 6. Such rhetorical phenomena that humans can immediately perceive are hard to recognize and model via natural language processing (NLP) systems. Traditional methods that heavily depend on a polarity dictionary would probably yield a "positive" output since there are more positive words in the example tweet than negative.

Errors in Twitter sentiment analysis are often caused by such sophisticated tweets that contain special phenomena such as rhetoric, which is one of the main reasons for failure in existing systems [72, 92, 90]. Therefore, to fully understand the flexibility of expressions of feelings in social media, it is necessary to observe real tweets by human beings and characterize the underlying context, particularly rhetorical context[1].

Given the above, to reveal clues that suggest the true global polarity of a tweet, we propose a relatively fine-grained annotation scheme based on separating emotional signals and rhetorical context, thus allowing a deeper investigation of the underlying mechanism by using instances with the same language phenomena. In addition to global polarity, our scheme identifies key components that may affect the emotions of tweets, including the use of rhetoric devices, emotional signals, degree modifiers, and subtopics. To briefly illustrate our scheme, an example annotation of the above example tweet is as follows. In the first two sentences, there are three positive signals (i.e., wow, can, and like) and two intensifiers without a specific context (i.e., just and any). Next, the polarity of iPhone 6 is compared to a negative object in the third sentence. The sarcasm identified across the three sentences then finally determines the global polarity of the original tweet as being "negative."

**Wow**(positive)**, with #iPhone6, you can**(positive) **send a message just**(intensifier) **by talking! In any**(intensifier) **voice you like**(positive)**.** [**So can my mom's old**(negative) [**rotary dial**] *(Comparatively equal)*.] *(Sarcastically negative)*

⊙ Global Polarity to iPhone 6: **Negative**

To tackle the second challenge, a multilingual corpus that can support intercultural comparison is necessary. Annotated datasets for sentiment analysis in social media have already been proposed, but these are primarily monolingual.

[1] Other contexts such as part of discourse context (e.g., but, despite) [50], whole-part context, and temporal context also affect global polarity, but occur much less frequently and can be handled by other existing technologies. In this paper, we therefore focus on rhetorical context.

The few multilingual corpora are topic-irrelevant, making it impossible to verify whether there are differences in public mood regarding the same object in different cultures. Therefore, we implement our annotations in a multilingual setting on common international topics. More specifically, we span three languages, i.e., English, Japanese, and Chinese, to magnify the variations between languages; further, to avoid domain limitations, we cover three common genres of evaluation objects, namely products (i.e., iPhone 6 and Windows 8), public figures (i.e., Vladimir Putin), and events (i.e., Scottish Independence).

We apply our annotation scheme to 5422 real tweets. To solve the reported problem of low inter-annotator agreement in previous Twitter sentiment annotation methods, we propose a pivot dataset comparison (PDC) method to improve agreement by correcting understanding errors. Our PDC method represents a good compromise between annotation quality and speed; as detailed in Section 5.3, increases in the Kappa statistic endorses our method's effectiveness and reliability.

Many applications would benefit from better Twitter sentiment analysis systems, which rely on corpora with rich annotations. Our corpus can serve as an ideal testbed for measuring the effectiveness of any sentiment analysis method, especially its rhetoric tolerance and cross-language adaptability, since our corpus includes multilingual tweets in various contexts. Further, our word/phrase-level annotated corpus makes it possible to experiment with new methods for solving special language phenomena in sentiment analysis. As examples, deep leaning methods and the synthesis of multiple rhetoric solvers can be implemented to comprehend the emotions implied in rhetorical contexts based on certain forms of our corpus development. Therefore, the goals of this corpus are to reveal the key principles behind the expression of feelings in social media and explore possible breakthrough points for the aforementioned challenges. By making flexible use of our annotated multilingual dataset, we hope to promote research on sentiment analysis for social media in multilingual settings.

In this chapter, we first describe the construction of such a resource but also report on its analysis, and we do the analyses of the constructed corpus in the next chapter(i.e., Chapter 4). To the best of our knowledge, our work is the first to build a comprehensive multilingual corpus that handles rhetorical phenomena for sentiment analysis in social media.

The remainder of this chapter is organized as follows. In Section 3.2, we discuss related work. In Section 3.3, we describe the data collection and selection processes for our annotations. In Section 3.4, we elaborate our annotation scheme by using examples. We introduce the annotation process and the effect of our PDC method in Section 3.5. In Section 3.6, we present individual differences of different annotators and note the deficiencies of our annotations. Finally, we describe our conclusions and suggest future work in Section 3.7.

## 3.2 Related Work

To date, there are numerous annotated datasets in the field of sentiment analysis. In this section, we describe these datasets and show how our annotation scheme differs from these.

### 3.2.1 Traditional Datasets for Sentiment Analysis

Movie reviews, product reviews, newswire feeds, and so on are traditional study objects for sentiment analysis, which is also known as opinion mining. Pang, Lee, and Vaithyanathan [57] used movie review data to test the effectiveness of machine learning methods for sentiment analysis. The latest version of their dataset consists of 1000 positive and 1000 negative processed reviews. Wiebe, Wilson, and Cardie [89] annotated the MPQA corpus, which contains hundreds of news articles from a wide variety of news sources, by using a fine-grained scheme that centered on private state, e.g., beliefs, emotions, sentiments, and speculations. Liu, Hu, and Cheng [36] gathered thousands of consumer opinions from online customer review sites into the pros and cons dataset, discussing a new technique to identify product features (i.e., attributes). Ganapathibhotla and Liu [18] collected hundreds of comparative sentences from product review websites and online forums, and then proposed an opinion mining method that identifies preferred entities.

Miyazaki and Mori [48] proposed a model for separating attribute-value pairs of products and their corresponding evaluations, discussed a method for decreasing disagreement between annotators, and annotated a collection of product reviews from an online commercial site for information extraction.

The above corpora have proved very valuable as resources for learning about the expression of feelings in general, but do not focus on social media. Unlike traditional long text inputs, our social media corpus consists of short text inputs, each no more than 140 characters; given this relatively limited length, the means of expression have become much more diverse, thus introducing new challenges to annotation.

### 3.2.2 Twitter Datasets for Sentiment Analysis

SemEval 2013 and 2014 tasks [51, 61] offer a dataset comprised of thousands of English tweets tagged with global polarity. Many researchers [3, 91] have performed experiments on it for various purposes. The TASS corpus [83] is a Spanish Twitter corpus consisting of 7219 messages tagged with global polarity and entity polarity (where it exists). The i-sieve corpus [32] and the Sanders corpus [72] are two English tweet datasets offered by private companies, with only the latter being publicly available.

Similar datasets tagged only with global polarity were introduced by Saif, Fernandez, He, and Alani [66], while there are other datasets that use noisy labeling with emoticons [20, 55]. Similarly, sarcasm tweet datasets have been built for English [21, 59] by relying on #sarcasm or #irony hashtags; however, datasets that stem from noisy labeling contain significant levels of noise and bias [13, 58]. Tang and Chen [77] built a Chinese irony microblog dataset containing 1005 ironic messages from a bootstrapping procedure using customary patterns (e.g., degree adverb and positive adjective) that they claimed to be the first irony dataset for Chinese.

In general, Twitter datasets for sentiment analysis are monolingual and labeled only with global polarity, whereas our corpus is carefully tagged with rich information at the word/phrase level. In addition, most current datasets do not consider other rhetoric devices except sarcasm, and datasets with sarcasm have various constraints (e.g., the reliance on specific hashtags or patterns). Conversely, our corpus contains four common rhetoric devices (i.e., comparison, metaphor, sarcasm, and rhetorical question) without restrictions.

### 3.2.3 Multilingual Datasets for Sentiment Analysis

Few previous studies exist on multilingual sentiment annotation for traditional text. Steinberger, Lenkova, Kabadjov, Steinberger, and Goot [74] annotated a sentiment-oriented parallel news corpus in seven European languages, i.e., English, Spanish, French, German, Czech, Italian, and Hungarian, with opinions toward entities in a sentence; however, here, gold-standard annotations were actually performed in English, and then simply projected to the other six languages. They used this corpus to evaluate a prototype system based on their multilingual sentiment dictionaries. Similarly, Kozareva [33] manually annotated a metaphor-rich corpus with polarity and valence scores for four languages (i.e., English, Spanish, Russian, and Farsi) and showed that the proposed method for polarity and valence prediction of metaphor-rich texts is portable and works well for different languages.

In general, multilingual social media engagement is growing. Volkova, Wilson, and Yarowsky [84] constructed a multilingual tweet dataset including English, Spanish, and Russian by using Amazon Mechanical Turk. They compared the variation in gender information in the three languages, showing that gender differences can effectively be used to improve sentiment analysis. Balahur and Turchi [3] constructed a dataset by translating English tweets into Italian, Spanish, French, and German. They tested the performance of their sentiment analysis classifiers for these languages, showing the effectiveness of the joint use of training data from multiple languages.

The languages used in the above corpora, each of which was constructed for a different multilingual study, are relatively close, i.e., they all belong to the Indo-European language family; conversely, our three selected languages are more distant, i.e., they belong to three different language families[2]. Further, compared with previous multilingual Twitter datasets, our corpus has two advantages. First, evaluation objects are the same in our corpus, enabling us to compare public opinion and interest between different cultures. Second, with the help of our fine-grained annotations, we can observe differences in emotional expression between languages.

## 3.3 Dataset Creation

In this section, we describe how we selected our evaluation objects, collected related tweets, and chose representative ones for annotation.

### 3.3.1 Evaluation Objects

To support comparisons of cultural differences[3], such as sentiment distribution, emotion evolution over time, and subtopic composition, we note that common and controversial topics discussed across the three selected languages (i.e., English, Japanese, and Chinese) are preferred. In our study, we considered many candidates and carefully selected four international topics spanning three genres as our evaluation objects; more specifically, we selected iPhone 6 and Windows 8 to represent products (i.e., tangible and intangible, respectively), Vladimir Putin for public figures, and Scottish Independence for events.

The number of evaluation objects is mainly constrained by limited resources (i.e., time/money). Considering our annotation is rather fine-grained, the number of evaluation objects is not allowed to be too large. However, our four evaluation objects cover various genres and are representative, so they meet the requirement

[2] According to https://en.wikipedia.org/wiki/Language_family, English, Japanese, and Chinese belong to Indo-European, Japanese, and Sino-Tibetan language families, respectively.

[3] In this paper, culture is roughly defined by language. This definition is reasonable for Japanese and Chinese because these two languages are primarily used by Japanese people in Japan and Chinese people in China, respectively. As English is currently widely used, it cannot be restricted to a fixed region. Still, it is acceptable to use it to represent Western culture to some extent. Discussions on the differences in English used in different regions is beyond the scope of this paper.

in terms of the purpose of the corpus. Besides, as to the timeliness of evaluation objects, the cooling-down of evaluation objects will not reduce the value of our corpus radically. This is because that we focus more on language phenomenon (e.g., rhetorical context) and the comparison of them between languages, and these will not change greatly during a short time.

Table 1 shows each of these four targets' corresponding query keywords for obtaining tweets[4]. These keywords are the most frequently used representations for the evaluation objects in each language. For brevity, we use the abbreviations listed in the second column of Table 1.

Table 1: Query Keywords Used for Data Collection

| Object | Abbr. | English | Japanese | Chinese |
|---|---|---|---|---|
| iPhone 6 | I6 | #iPhone6 lang:en | #iPhone6 lang:ja | iPhone6 |
| Windows 8 | W8 | #Windows8 lang:en | #Windows8 lang:ja | Win8 系统 |
| Vladimir Putin | PU | #Putin lang:en | プーチン | 普京 |
| Scottish Independence | SI | Scotland Independence lang:en | スコットランド独立 | 苏格兰独立 |

## 3.3.2 Data Collection

Regarding the source of our data, we collected tweets from Twitter[5] via the Twitter REST API for English and Japanese using the same approach as that found in many other studies [51]. Given the scarcity of Chinese tweets on Twitter, we decided to use Weibo[6], a well-known Chinese version of Twitter, as a substitute. For Twitter, we employed Tweepy[7] to access its Search API, whose returned

[4] We noticed afterward that the number of tweets after selection was not high enough for iPhone 6 and Windows 8 in Japanese; therefore, we used other keywords as supplements.

[5] http://www.twitter.com

[6] http://weibo.com

[7] http://www.tweepy.org

results are similar to, but not the same as, its search service[8]. The Twitter API has a seven-day backtracking limit, so it is designed to run daily. Since Weibo's Search API cannot be accessed freely, we resorted to a scraper that fetched results directly from its search service[9]. Because the maximum number of pages returned per day is 50 in Weibo's search service[10], we fetched only original tweets to avoid duplication at the source.

The one-year collection period started on October 19, 2014, and ended on October 18, 2015. For the convenience of management and use, we stored all tweets in a database. Twitter tweets were easily decoded since they use a JSON format, whereas we used a parser to extract desired fields from Weibo tweets because they were sourced directly from HTML files. Table 2 shows the number of tweets collected per object and per language. The table indicates that the numbers of English and Japanese data are comparable to one another, but substantially more than the number of Chinese data. Nonetheless, they are all distributed similarly over the objects; iPhone 6 and Putin apparently attracted more attention than Windows 8 and Scottish Independence across all three cultures, whereas Scottish Independence ranked higher in English than in the other two languages.

Table 2: Number of Tweets Collected between October 19, 2014, and October 18, 2015

| Object | English | | Japanese | | Chinese | |
|---|---|---|---|---|---|---|
| | Total # | Per Day | Total # | Per Day | Total # | Per Day |
| I6 | 2,690,386 | 7,370.9 | 1,883,320 | 5,159.8 | 172,914 | 473.7 |
| W8 | 109,076 | 298.8 | 66,546 | 182.3 | 16,951 | 46.4 |
| PU | 1,006,677 | 2,758.0 | 915,273 | 2,507.6 | 124,121 | 340.1 |
| SI | 183,358 | 502.4 | 27,718 | 75.9 | 2,410 | 6.6 |
| Total | 3,989,497 | 10,930.1 | 2,892,857 | 7,925.6 | 316396 | 866.8 |

[8] https://twitter.com/search-home

[9] http://s.weibo.com

[10] Weibo's retrieval mechanism changed once, which resulted in a drop in the number of Chinese tweets in the last six months of our collecting period.

### 3.3.3 Tweet Selection

As shown in Table 2, the size of the raw data was too big for us to annotate all tweets. Further, social media such as Twitter contains a substantial number of undesirable tweets, such as retweets, commercials, and objective news, all of which are of low value to the annotation stage for sentiment analysis [26]. Therefore, selecting representative tweets was inevitable.

There are two approaches for selecting desired instances from a large amount of data, i.e., exclusive filtering [22] and inclusive filtering [51]. To ensure that the annotation datasets are good estimates of public mood and simultaneously cover the diversity of emotional expressions to the extent possible, we designed a two-stage method to combine the two approaches.

In the first stage, we used exclusive patterns to veto unsatisfactory tweets, i.e., we removed tweets containing exclusive patterns from the raw data. This may cause some over-excluding, but based on our preliminary investigations, most of the tweets containing these patterns were not opinionated. Table 3 shows the exclusive patterns that we used.

Table 3: Patterns Used for Excluding Non-opinionated Tweets

| Patterns | Description |
|---|---|
| `^RT` | Pattern indicating that the tweet is a retweet. |
| `[a-zA-z]+://[^\s]*` | Pattern indicating that the tweet contains a URL reference. |
| `【.+?】` | Pattern indicating that the tweet is a commercial in Japanese or news in Chinese. |
| • English: news \| breaking...<br>• Japanese: 限定 \| 在庫 \| 特価...<br>• Chinese: 分享 \| 资源 \| 共享... | Words indicating that the tweet is objective (i.e., commercial and news) for each language. |

In the second stage, we performed two inclusive selections in a soft way. We first preferred longer tweets since short tweets contain less linguistic richness. Next, we preferred tweets that contained fewer special symbols (e.g., @ and #) based on the observation[11] that the more special symbols that a tweet contains, the less likely it is to be opinionated. The thresholds for tweet length and the number of special symbols depended on the size of the previous remaining set. If the size was large, we selected longer tweets with fewer symbols; otherwise, either selection was skipped.

We found that a large portion of tweets was omitted from the raw data using our two-stage screening process, with 93.1% omitted in the first stage, 75.4% in the second stage, and 98.3% overall. Nonetheless, some inappropriate tweets still existed, so we manually checked the remaining tweets sequentially, filtering out apparently worthless ones until we obtained the designated number of tweets for annotation. In short, we removed three types of tweets: (1) repeated tweets that did not start with RT, (2) obviously objective tweets that did not contain the common veto words shown in Table 3, and (3) off-topic tweets that contained query keywords but did not actually discuss or appraise the evaluation objects. Refer to Appendix A for more details regarding our data selection process.

## 3.4 Annotation Scheme

A well-designed representation scheme is vital to the success of annotation work. Most existing corpora introduced in Section 2 only label global polarity and lack systematic schemes. The private state scheme adopted in the MPQA dataset by Wiebe et al. [89] is one of the few schemes for fine-grained sentiment annotation, focusing on presentation frames for private state expressions (i.e., subjective expressions). The private state scheme is a good reference here, but since our research purpose and text type are both different from those of Wiebe et al. [89], we must take more aspects into account. In this section, we therefore introduce the basic ideas behind our annotation scheme, and then detail our presented an-

[11] An additional experiment on 220 randomly selected English tweets yields similar results, i.e., part of special symbol count/opinioned tweet ratios of 0/0.75...3/0.75...6/0.50...9/0.30...

notation standards with examples[12].

### 3.4.1 Fundamentals

Based on our initial investigation of a certain number of tweets, we found that there are primarily two ways to express human emotion, i.e., direct expression and indirect expression. In direct expression, people express their feelings in a straightforward manner with explicit emotional elements. Conversely, indirect expression may not contain any superficially emotional elements; instead, people utilize rhetoric devices, such as comparisons, metaphors, sarcasm, and rhetorical questions, to express their opinions. Figure 3 exemplifies these two different ways of expression by showing typical tweets using both techniques. Note that these examples are relatively simple; real tweets for annotation are typically much more complex.

To accommodate the characteristics of these two ways of expression, our scheme separates emotional signals (i.e., the elements containing emotion toward the evaluation object) and rhetorical context in a tweet. The polarities of emotional signals are not vulnerable to the tweet's context (Section 4.2), while the rhetorical context is modeled by formulating the common rhetoric devices at the sentence level (Section 4.3); the global polarity of a tweet can then be determined by integrating the polarities of emotional signals and the rhetorical context (Section 4.4). As an example, in the second sentence of tweet (7) shown in Figure 1 (i.e., "Its an almost perfect #antidesign"), the word "perfect" is originally positive and "#antidesign" is clearly negative; thus, the contradiction of polarities (i.e., positive vs. negative) within the sentence forms the sarcasm context[13]. Together, this makes the sentence strongly negative. Here, "perfect" will not be tagged as negative even though it is used in an ironic context. This separation is extremely important for revealing underlying patterns of expressions of feelings.

Further, we use the same scheme for all three languages. This ability is based

[12] In this section, we primarily present English examples; multilingual examples are presented in Section 7.2.

[13] For a detailed explanation on rhetorical contexts, please refer to Section 7.2.

**Direct Expression**

(2) Negative: Getting really p*ssed off with #Windows8 it really is crap! The 'Search' facility doesn't work properly & now its lost some of my pics!

(3) Positive: It's beautiful. The resolution of pictures & videos, the screen size, the slow-motion capability when making videos & more. Amazing #iPhone6

(4) Neutral: I don't want to split Scotland's independence vote, but I'm got no more avenues open to me. I hoped my sister could help. She can't.

**Indirect Expression**

(5) Comparison: Can now definitively say that #Windows8 IS indeed faster and more stable than #Windows7 used both for a while now. Don't be afraid of 8 #fb

(6) Metaphor: As a #Mac user, #Windows8 is figuratively the bane of my existence. Trying to do anything is nigh on impossible.

(7) Sarcasm: Every time I use #Windows8, I become more impressed with how profoundly bad a UX it is. Its an almost perfect #antidesign

(8) Rhetorical Question: last #windows8 update took more time than loading 20 #c64 games with #datasette ...what went wrong in 30 years?

Figure 3: Tweets using Different Means of Expressions

on the hypothesis that although the three languages are different at the word and syntax levels, the ways of expressing feelings are similar. Given that direct proof of this hypothesis is difficult, we resort to a *reductio ad absurdum* method. More specifically, if we find any exception that contradicts the hypothesis, we refuse it; otherwise, we accept it. Because this empirical proof is required to go through both the annotation process and corpus analysis, we present our conclusions at the end of Section 7.

### 3.4.2 Emotional Signals and their Degree Modifiers

Emotional signals (or simply signals) are the basic emotional elements in a tweet. In tweet (3) of Figure 3, words like "beautiful" and "amazing" are positive signals for the iPhone 6 evaluation object. Here, there are three types of emotional signals, defined as follows.

- Positive signals: signals showing good attitudes toward the evaluation object.
- Negative signals: signals showing poor attitudes toward the evaluation object.
- Neutral signals: signals showing neutral or undecided attitudes toward the evaluation object.

The major difference between an emotional signal and a polarity word is that a signal influences global polarity, while a polarity word may or may not have such influence. In tweet (9) shown below, even though "cool" is generally a positive word in any polarity lexicon, it is not considered a positive signal here because it does not constitute a judgment on iPhone 6; hence, "cool" should not be tagged in this case. Further, similar to private state expressions in the MPQA dataset, annotators are not limited to marking any particular words. Signals comprised of multiple words, such as phrases and idioms, and implicit signals not containing any explicit polarity words are also allowed. Examples of such negative signals are "p*ssed off" from tweet (2) and "achilles heel" from tweet (10).

(9) **my cousin thought it would be cool to sit on my phone&see if it bend. And NOPE, it didn't. #iPhone6 I think because i have a case on**

(10) **Kasyanov: #Putin's achilles heel is economy - when regular people feel the pinch, when pension payments aren't met, it starts to crumble**

According to the separation idea, annotators are asked to label signals with their original polarity in everyday use (strictly speaking, in social-media use). In tweet (7), even though "impressed" and "perfect" are used to satirize Windows 8, their polarities should be labeled as "positive." Note that this does not mean that we judge words out of context; on the contrary, we consider not only their meanings but also their roles in context.

Further, we define three types of degree modifiers for two reasons. First, degree words are important surrounding information for signals. Second, degree words can help annotators distinguish the boundary between signals, which is crucial for non-space separated languages, such as Chinese and Japanese. The three types of degree modifiers are then defined as follows.

- Intensifiers: words that strengthen the signals they modify, e.g., very and really.
- Diminishers: words that weaken the signal they modify, e.g., a little and almost.
- Negations: words that reverse the signals they modify, e.g., not.

For each of these degree modifiers, in each language, there is only a limited number of degree expressions in the dictionary, on the order of tens of expressions, especially for negation (which consists of only a few). Degree modifiers are usually explicit, such as "really" in tweet (2), "almost" in tweet (7), and "doesn't" in tweet (2); however, negation can sometimes be implicit. For example, "should" in tweet

(11) below is a negation of the positive signal “fix problems.” In addition, degree modifiers usually appear together with emotional signals, such as “profoundly bad” in tweet (7) and “doesn’t work” in tweet (2). Solely tagged degree modifiers must be avoided.

**(11) @Microsoft really? I updated Win 8.1 because it should fix problems, not generate more troubles!! #windows8 Sucks!!!**

### 3.4.3 Rhetorical Context

Rhetorical phenomena essentially occur at the sentence level. In linguistics, there are approximately 20 classes of rhetoric devices, whereas in our computational linguistics setting, we focus on the four commonly used rhetoric devices, i.e., comparisons, metaphors, sarcasm, and rhetorical questions [8, 18, 19]. For simplicity, these four types of rhetoric devices are defined in a relatively loose manner, as well as a fifth non-rhetoric type, as follows.

- Comparisons: used if the tweet compares the evaluation object with other counterparts. Comparisons also include contrast.
- Metaphors: used if the tweet identifies the evaluation object as being similar to some unrelated thing. Metaphors include similar concepts such as similes, metonymies, and synecdoche.
- Sarcasm: used if the tweet contains sentences stating the contrary of what is actually meant. Irony is also a form of sarcasm in our setting.
- Rhetorical questions: used if the tweet includes a question asked to make a point rather than to elicit an answer. Answer-seeking questions are not considered to be rhetorical questions.
- Non-rhetoric: used if the tweet is a direct expression of feelings.

Each rhetoric device has its own representation frame. For a comparison frame, there are three possible slots. The comparison object slot contains the anchor text of the comparison object in the tweet. The comparison base defaults to the evaluation object. Relative status here means the comparative relation of the comparison object compared with the evaluation object, defined as inferior, equal, or superior. The relative status is given by the combination of the comparison context (e.g., "...er and more..." in tweet (5)) and the signals (e.g., "fast" and "stable" in tweet (5)). The polarity of the comparison base can then be decided by the relative status. For example, the comparison frame for tweet (5) is as follows.

**Rhetoric Type: Comparison**

**Comparison object: #Windows7**

**Comparison base: Windows 8(default)**

**Relative Status: inferior**

Similar to comparison, the metaphor frame also has three slots. The metaphor source slot contains the anchor text of the metaphor source in the tweet. The metaphor target defaults to the evaluation object. Metaphor polarity indicates the polarity of the metaphor source, which is negative, neutral, or positive. Annotators recognize the metaphor context (e.g., "...is figuratively..." in tweet (6)) and label the metaphor polarity (e.g., "negative" is attached to "the bane of my existence" in tweet (6)). The metaphor target then duplicates the metaphor polarity as its own polarity. For example, the metaphor frame for tweet (6) is as follows.

**Rhetoric Type: Metaphor**

**Metaphor source: the bane of my existence**

**Metaphor target: Windows 8(default)**

**Metaphor polarity: negative**

The frames for sarcasm and rhetorical question are similar to one another; each has two slots. The locating sentence slot contains the anchor text of the sentence in which sarcasm or a rhetorical question is located in the tweet. Sentence polarity indicates the polarity of the locating sentence to the evaluation object. The existence of sarcasm can be suggested by the contradictory signal pairs (e.g., (impressed, bad) and (perfect, #antidesign) in tweet (7)). The polarity of a rhetorical question can be obtained by integrating the corresponding context (e.g., "what went...?" in tweet (8)) and the signals (e.g., "wrong" in tweet (8)). Since the contexts of sarcasm and a rhetorical question are not as structured as the former two, their recognition relies on the subjective interpretation of an annotator. As examples, the sarcasm frame for tweet (7) and the rhetorical question frame for tweet (8) are as follows.

**Rhetoric Type: Sarcasm (1)**

> **Locating sentence: Every time I use #Windows8, I become more impressed with how profoundly bad a UX it is.**
>
> **Sentence polarity: negative**

**Rhetoric Type: Sarcasm (2)**

> **Locating sentence: Its an almost perfect #antidesign**
>
> **Sentence polarity: negative**

**Rhetoric Type: Rhetorical question**

> **Locating sentence: what went wrong in 30 years?**
>
> **Sentence polarity: negative**

A tweet can simultaneously contain two or more types of rhetoric devices. For example, tweet (1) in Section 1 (shown again below) includes two rhetoric devices, i.e., comparison and sarcasm. Further, the rhetorical context sometimes spans multiple sentences. Still referring to tweet (1), sarcasm does not locate in any single sentence. In fact, the polarity collision of the first two sentences with the third sentence forms the sarcasm context. To the best of our knowledge, similar research has not been conducted as part of any other sentiment annotation work. This not only allows us to collect explicit rhetorical patterns but also offers us the opportunity to analyze their implicit structures.

(1) **Wow, with #iPhone6, you can send a message just by talking! In any voice you like. So can my mom's old rotary dial.**

### 3.4.4 Global Polarity

Global polarity is fundamental information for a sentiment classification-oriented corpus. In accordance with Go et al. [20], the global polarity of a tweet is defined as "the author's personal feeling to the evaluation object." In our annotations, global polarity is divided into the following three categories.

- Positive: a tweet that shows the author's supportive attitude toward an evaluation object.
- Negative: a tweet that shows the author's non-supportive attitude toward an evaluation object.
- Neutral:

  ① A subjective tweet with attitudes either undecided or mixed (i.e., half positive, half negative).

  ② A non-opinionated tweet, such as a non-comment tweet, an objective tweet, or an irrelevant tweet.

According to the above definition, tweet (2) is a negative tweet, tweet (3) is a positive tweet, and tweet (4) is the first type of neutral tweet. Global polarity is sometimes difficult to determine for ambiguous tweets. For example, the global polarity of tweet (12) can be either negative or neutral depending on how people interpret the second sentence, i.e., either as mocking Scotland or as a pure statement. To secure better global polarities for these ambiguous tweets, instead of using simple majority voting over annotators' original answers, we propose an original and improved method called PDC, which we present in Section 5.2.

**(12) The nationalist criticism of the Smith Comission report is that it isn't independence. That's because Scotland didn't vote for that.**

### 3.4.5 Subtopic Information

It is important to know people's opinions regarding evaluation objects. It is also meaningful to know what type of related subtopics people are concerned about, which can then help us better understand differences between cultures. Therefore, we include subtopic information in our scheme.

Subtopics can be nouns or nominal phrases in tweets, and annotators are encouraged to edit them to create unified forms. If there is no direct subtopic text in a tweet, the annotator is allowed to infer it through summarization. For example, "screen" and "size" are subtopics for tweet (13); here, "screen" can be directly extracted from the first clause of the first sentence, while "size" can be obtained by summarizing the second clause.

Subtopics are not always as easy to discover as nouns, which are aspects or attributes of evaluation objects [25]. In particular, here, "bending" is also a subtopic of tweet (13); however, until we observe that there are a few tweets discussing the bending problem of iPhone 6, it is difficult to identify "bending" as a subtopic at first glance.

**(13) Just picked up an #iPhone6 the screen is beautiful, but my god is it large! Crossing my fingers it doesn't bend!**

## 3.5 Annotation Process

The annotation process has two phases, i.e., independent annotation (Phase 1) and annotation improvement (Phase 2). In this section, we first describe the annotation setup for Phase 1, and then detail the PDC method used in Phase 2. Finally, we analyze the effect of our PDC method.

### 3.5.1 Annotation Setup

For each object and language, we prepare a collection of approximately 450 tweets by the method described in Section 3.3 above. In total, there are 12 collections (i.e., three languages times the four objects). For each collection, three different annotators perform the annotations independently according to a common standard. For each language, there are six annotators and each annotator takes charge of two objects. Table 4 illustrates the allocation of annotators, denoted A1 through A6, for one language.

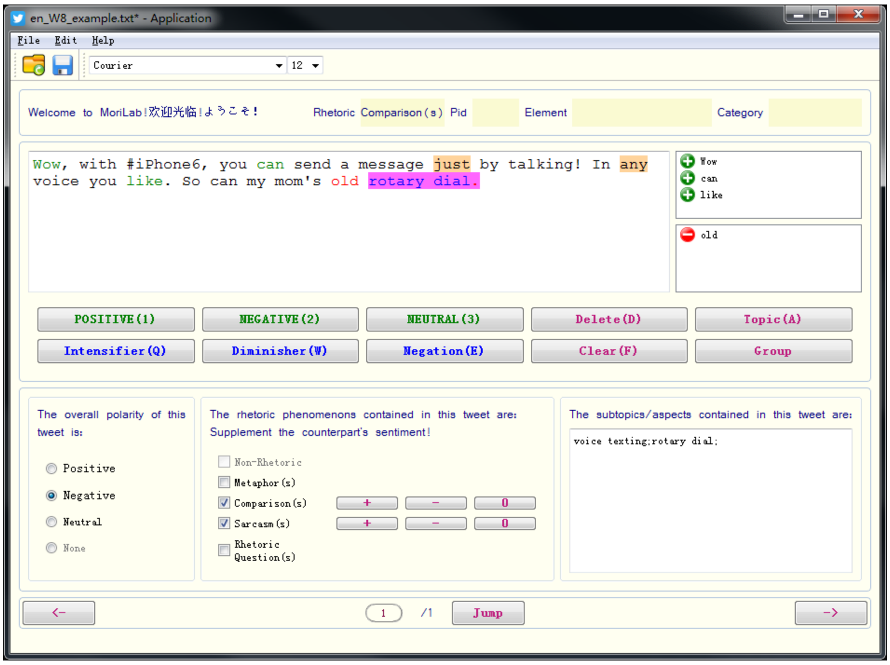
Table 4: Allocation of Annotators for One Language

| | A1 | A2 | A3 | A4 | A5 | A6 |
|---|---|---|---|---|---|---|
| I6 | ✓ | | ✓ | | ✓ | |
| W8 | | ✓ | | ✓ | | ✓ |
| PU | ✓ | | ✓ | | ✓ | |
| SI | | ✓ | | ✓ | | ✓ |

The annotator team consists of 1 supervisor[14] and 18 annotators. Given that expressions of feelings in social media can sometimes be rather subtle, each of our annotators is a native speaker or has the same proficiency as that of a native speaker for each language. More specifically, Japanese annotators are all native undergraduate students, while Chinese annotators are all native graduate students. Considering the geographically wide use of English, the English group consisted of two Americans, one Australian, one Indian, and two Europeans, all with excellent English skills.

To ensure high-quality annotations and maintain a stable annotation speed, each annotator received a three-hour training session with a coding manual before the formal work began; a brief introduction of rhetoric devices with examples was

[14] The first author of this paper supervised the annotation work.



Welcome to MoriLab!欢迎光临!ようこそ!

Rhetoric Comparison(s) Pid Element Category

Wow, with #iPhone6, you can send a message just by talking! In any voice you like. So can my mom's old rotary dial.

Wow
can
like

old

The overall polarity of this tweet is:

○ Positive
● Negative
○ Neutral
○ None

The rhetoric phenomenons contained in this tweet are: Supplement the counterpart's sentiment!

□ Non-Rhetoric
□ Metaphor(s)
☑ Comparison(s)
☑ Sarcasm(s)
□ Rhetoric Question(s)

The subtopics/aspects contained in this tweet are:

voice texting;rotary dial;

1 /1

Figure 4: Interface of the Annotation Tool

also distributed. The coding manual was continuously refined based on discussions of training results among the annotators and the supervisor until a consensus was reached.

Next, all the 18 annotators performed the annotation work independently according to the updated coding manual in a specified room. Each annotator was assigned 18 annotation hours to finish the two objects (i.e., approximately 900 tweets) for which he or she was responsible. The supervisor provided onsite support during the entire annotation period and did not provide any direct directives that may alter an annotator's own judgment. In practice, the supervisor helped to solve problems individuals faced with the annotation tool (described below), answered questions regarding the annotation method, and discussed the meaning of some tweets upon the request of the annotators.

To make operations more convenient for annotators, we developed an annotation support tool that implemented the annotation scheme described in Section 4. With the help of this tool, annotators could complete most of their tasks by mouse clicks and various keyboard shortcuts. Annotators also practiced using the tool as part of their training. Figure 4 shows the general interface of our annotation tool;

also see Appendix B for an example of annotation results in XML[15]. Annotators performed their annotations tweet by tweet until their tasks were completed. The annotation procedure for one tweet is summarized as follows (see Appendix C for the full details of the code manual).

(1) Annotators first quickly glimpsed the tweet, and then focused on the beginning of the tweet.

(2) From the beginning to the end of the tweet, annotators read and judged each word. If any emotional signal, degree modifier, or subtopic presented itself, the corresponding tag was added.

(3) After finishing step (2), annotators determined the global polarity of the tweet.

(4) Annotators chose the rhetoric devices that occurred in the tweet, supplementing the necessary information for each selected rhetoric device.

(5) Finally, annotators unified the forms of subtopics obtained in step (2) or summarized the tweet if no subtopics were identified.

### 3.5.2 PDC

The majority decision of the three original global polarities of each tweet in Phase 1 would be used as the final decision in previous studies on traditional text; however, for social media like Twitter, inter-annotator agreement on global polarity at this stage has been reported to be low [68, 6], so global polarities decided by simple majority voting may be insufficient given that annotators can make understanding errors (i.e., misunderstandings) and human errors (i.e., misoperations) in their independent annotations.

A quick way to correct possible errors in Phase 1 is to ask annotators to recheck their annotations, but this only works for human errors. For understanding errors, since sentiment annotation is rather subjective, from our experience, annotators

[15] This example is the final version from the gold standard, which is introduced in the next section.

are apt to stay with their old way of thinking and make few changes. Consider the following two tweets regarding Windows 8 as examples. One annotator misunderstood “that” in tweet (14) as modifying “Windows 10 DRM[16],” and incorrectly tagged tweet (14) as “positive”; in actuality, “that” modifies “steps to fix DRM,” so tweet (14) is actually “neutral.” Another annotator mistook “linx7” as “Linux” and incorrectly tagged tweet (15) as “positive”; in actuality, “linx7” is a Windows 8 tablet, so tweet (15) is “negative.” It is difficult to correct these understanding errors with self-checks. To address this problem, the comparison method is more feasible to implement in that annotators can quickly and precisely locate their errors by comparing their annotations with reference annotations. For tweets (14) and (15), if we show annotators that they are more likely to be “neutral” and “negative,” and give them instructions about where the problem may lie, it becomes easier for them to recognize their understanding errors.

**(14) @GabeAul Went through all the steps to fix #Windows10 DRM that worked in #Windows7 and #Windows8, and then some, but no luck. Weird!**

**(15) Trying not to #lol as toms losing it trying to suss his #linx7 #windows8 🐒 #notsomuchofabargainnow**

To realize this idea, we propose the PDC method, which involves the following two steps. First, we generate good reference annotations and collect them in a pivot dataset. Second, we use these annotations for comparisons. Each of these two steps is further described below.

- Step 1: Manual Merging

  The supervisor first goes through the meaning of each tweet in the corpus[17]. If the supervisor has a disparate opinion from the majority decision, or if

[16] DRM stands for digital rights management.

[17] This created a large amount of labor for the supervisor, but efficient quality management cannot be carried out if the supervisor has little involvement with the tweets.

the majority decision is undecided (i.e., all three original answers for global polarity differ), a fourth judgment is made by a new native annotator[18]. If the majority decision of the four answers of a tweet differs from its original three answers, the global polarity of this tweet is temporarily changed to the majority decision of the four answers[19]. Meanwhile, the components (i.e., emotional signals, degree modifiers, rhetorical contexts, and subtopics) are manually merged by integrating the three original annotations[20] and can be altered according to the fourth judgment. Results of this manual merging constitute the pivot dataset.

- Step 2: Pivot Dataset Comparison

  Based on comparisons with the pivot dataset, with both the global polarity and components shown to the annotators, we ask all annotators to revise their own original annotations. Although the supervisor is allowed to give instructions here, the annotators themselves finally decide whether to change or stay with their original answers to maintain the independence of each re-judgment. Components related to global polarity (i.e., emotional signals, degree modifiers, and rhetorical contexts) are updated along with changes in global polarity, if necessary. As a tradeoff between time cost and resulting benefits, only tweets with global polarities that differ from the pivot dataset are re-judged. After revisions are made, the pivot dataset is updated in turn by the majority decisions of the three annotators' updated answers, which is called the gold standard.

From above, in addition to the original datasets, the PDC method produces three new datasets, i.e., the pivot dataset, the revised datasets, and the gold standard. The entire process of the PDC method and the relationships among these datasets are depicted in Figure 5, where ①② correspond to Step 1, and ③④

[18] The new annotator is still one of the original annotators, but he or she was responsible for the other two objects, so the annotation standard did not change at all.

[19] If the new majority decision becomes undecided, the supervisor's opinion will be considered.

[20] The supervisor refers to the three original component tags of a tweet, and then considers whether to keep/delete/modify any tag that appears in the original annotations or add new ones according to the same annotation schema.

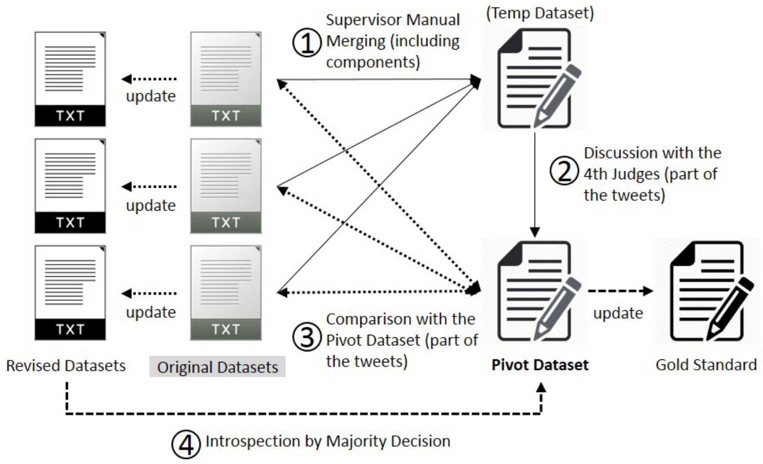

TXT
update
Revised Datasets
Original Datasets
① Supervisor Manual Merging (including components)
(Temp Dataset)
② Discussion with the 4th Judges (part of the tweets)
③ Comparison with the Pivot Dataset (part of the tweets)
Pivot Dataset
update
Gold Standard
④ Introspection by Majority Decision

Figure 5: PDC Method and its Resulting Datasets

correspond to Step 2. Another advantage of our PDC method is that it is introspective. Since the pivot dataset involves human judgment, the process introduces new errors as well, but incorrect global polarities can be fixed in reverse if two or more annotators refuse to make any changes. Therefore, the gold standard is taken as the optimal dataset in this paper.

As for subtopic information, format errors (e.g., typos) in each tweet are fixed during the merging of the pivot dataset. To avoid notation discrepancies across tweets, the supervisor further manually calibrates subtopic expressions to unified shapes in the gold standard. There are two types of unification here, i.e., (1) unifying subtopics in different shapes to the same shape (e.g., DevoMax, #devomac, and devo-max are unified as DevoMax[21] for Scottish Independence) and (2) unifying subtopics with the same meaning but different expressions (i.e., synonyms), e.g., dropping, falling, and slipping are unified as dropping for iPhone 6.

As a comparison, consider how the PDC method differs from the naive method of rechecking once there is a difference between the annotators' original answers. In terms of time cost, since the PDC method only asks annotators to recheck tweets with global polarities that differ from the pivot dataset, the number of tweets that must be rechecked is less than that of the naive method. To illustrate, in the example shown in Figure 6, the PDC method saves one time of rechecking in which the original answer is neutral. In terms of quality benefits,

[21] "DevoMax" stands for maximum devolution.

since the pivot dataset has already recognized understanding errors through Step 1, the comparison with reference annotations can help annotators promptly and accurately locate possible understanding errors. Conversely, as mentioned above, self-checking via the naive method is inefficient for detecting understanding errors. Note that our PDC method can even discover issues when all original answers are identical but incorrect. The disadvantage of the PDC method is that it requires additional time to obtain the pivot dataset, making it impractical for large-scale corpora. Simply put, the PDC method moves much labor from the annotators to the supervisor (and the fourth judges when necessary) such that annotators can focus on the tweets that very much need their attention. The empirical analysis of the effect of the PDC method is described in the next section.

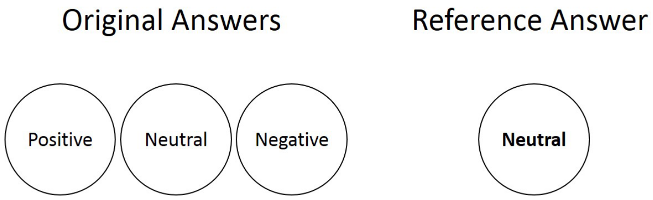


Figure 6: Typical Case of the Original Answers and a Reference Answer

### 3.5.3 Effect of the PDC Method

Table 5 shows Cohen's Kappa statistics for the global polarity of Phase 1 (i.e., independent annotations) and Phase 2 (i.e., applying the PDC method). As shown in the table, the inter-annotator agreement rates were relatively low for all three languages in Phase 1 (i.e., 0.482, 0.600, and 0.576 for English, Japanese, and Chinese, respectively), indicating that the agreement level of the original annotation was moderate (i.e., 0.4–0.6), with English being the language for which consensus was most difficult to achieve.

On the other hand, Cohen's Kappa statistics increased to a substantial (i.e., 0.6–0.8) or almost perfect (i.e., 0.8–1.0) level (i.e., by 0.283, 0.224, and 0.238 for English, Japanese, and Chinese, respectively) after the PDC method was applied. Among all collections, the lowest value was 0.693 (i.e., substantially reliable), whereas the highest was 0.855 (i.e., almost perfect). These results justify the

Table 5: Average Kappa Statistics for Global Polarity

| Object | English | | | Japanese | | | Chinese | | |
|---|---|---|---|---|---|---|---|---|---|
| | Phase 1 | Phase 2 | +/− | Phase 1 | Phase 2 | +/− | Phase 1 | Phase 2 | +/− |
| I6 | 0.504 | 0.784 | 0.280 | 0.611 | 0.855 | 0.244 | 0.440 | 0.694 | 0.254 |
| W8 | 0.541 | 0.836 | 0.295 | 0.547 | 0.783 | 0.237 | 0.506 | 0.840 | 0.334 |
| PU | 0.323 | 0.694 | 0.371 | 0.521 | 0.760 | 0.239 | 0.617 | 0.847 | 0.230 |
| SI | 0.456 | 0.693 | 0.237 | 0.406 | 0.740 | 0.334 | 0.569 | 0.796 | 0.226 |
| Overall | 0.482 | 0.765 | 0.283 | 0.600 | 0.824 | 0.224 | 0.576 | 0.814 | 0.238 |

main idea presented in Section 5.2 and demonstrate the effectiveness of our PDC method. By comparing results with the pivot dataset, the most obvious errors are easily revised, including human errors, tagging irreverent tweets as emotional, and mistaking author mood as the evolution of the object. Many more challenging understanding errors are also detected here, such as incorrect tagging due to a lack of background knowledge, mistaking situation analysis as an opinion, and misunderstanding from carelessness.

Table 6: Number of Tweets Requiring Rechecking in Different Settings

| Setting | English. | Japanese | Chinese | Avg. |
|---|---|---|---|---|
| The naive method | 432 | 338 | 370 | 380 |
| The PDC method (compared) | 212 | 154 | 164 | 176 |
| The PDC method (changed) | 139 | 99 | 99 | 112 |
| The PDC method (ideal) | 200 | 147 | 153 | 166 |

Next, we discuss the effect from the perspective of time cost. Table 6 shows the number of tweets that should be or have been rechecked in different settings. In Phase 2, each annotator spent three hours on comparison, comparing 176 tweets with the pivot dataset on average (i.e., 212 for English, 154 for Japanese, and 163 for Chinese). Among these, each annotator changed 112 tweets on average (i.e., 139 for English, 99 for Japanese, and 99 for Chinese). In contrast, each annotator should compare 380 tweets on average (i.e., 432 for English, 338 for Japanese, and

370 for Chinese) if using the naive method, which is far more than that in the PDC method. Further, we computed the ideal numbers that should be compared when using the gold standard as reference answers, showing that each annotator should at least have compared 166 tweets on average in Phase 2 (i.e., 200 for English, 147 for Japanese, and 153 for Chinese). As mentioned in Section 5.2, the pivot dataset introduces new errors, which is why there is a gap between the ideal and reality; however, compared with the naive method, our PDC method largely reduced the number of tweets that annotators needed to recheck.

## 3.6 Individual Differences and Annotation Deficiencies

In Section 5.3, we explained how the PDC method improved inter-annotator agreement by eliminating both human and understanding errors. In this section, we list reasons for the disagreement caused by the individual differences of annotators. Finally, we state the deficiencies of our annotations.

### 3.6.1 Individual Differences of Annotators

Although the inter-annotator agreement was substantially improved, there still were some ambiguous tweets with polarities varying from individual to individual, and even from time to time for the same individual. For these tweets, the annotation disagreement is not from errors but from the individual differences of annotators, i.e., their changing interpretation of tweets with intrinsic ambiguities. We present below different types of ambiguities and differences that may cause disagreement.

(1) Subjectivity Ambiguities

Some tweets were just on the borderline between subjective and objective. As an example, some annotators viewed tweet (23) below as an objective statement, whereas others viewed it as implying an opinion.

**(23) Bottom line. Until the BBC is brought onboard or booted out Scotland will not gain independence Huge audience believes all that is broadcast**

(2) Relevance Ambiguities

Some tweets can be either relevant or irrelevant. One could consider tweet (24) below to be about problems with YouTube. Conversely, it could also be interpreted as stating that iPhone 6 gave rise to the troubles with YouTube.

**(24) Can someone explain to me why YouTube videos can't run fluidly anymore?? Grr, what is this! #iPhone6**

(3) Understanding Differences

Some signals can be interpreted in multiple ways. As an example, "new born baby" in tweet (25) below could refer to preciousness and fragility at the same time.

**(25) When Someone hands you an I phone whiteout a case it feels like your handling a new born baby #iPhone6**

(4) Thinking Differences

Different ways annotators think may lead to deviations in understanding. In general, tweet (26) below is considered positive because iPhone 6 makes the author feel "cool"; however, one of the annotators thought that a phone being used as a means to show off was sad; thus, he or she classified tweet (26) as negative instead.

**(26) Think I want to buy an #iPhone6 . not because I like them.. but because apparently it makes me cool.. and I just wanna be cool.. that's all**

(5) Cultural Differences

The background of an annotator might influence his or her understanding of the given tweets. For tweet (27) below, Western people tended to think of "communist" in a negative sense, whereas Asian annotators may just think of it as a neutral political conception.

**(27) haters am throwin deuces yeah its peace ,coz am chilled lyk a buddhist long live #putin u tha last communist**

(6) Rhetorical Ambiguities

Rhetoric devices, especially sarcasm and rhetorical questions, rely heavily on the subjective interpretation of the annotators. To illustrate this, tweet (28) below can be understood in either a sarcastic or non-sarcastic way, and tweet (29) below can be a rhetorical or non-rhetorical question according to how each annotator thinks.

**(28) I seriously love how huge my phone is. When I talk on it, it takes the whole side of my face. Every time its like getting a hug. #iPhone6**

**(29) what if #Putin is doing all this just to make sure no one is stupid enough to want to clean up after him in the next term?**

(7) Weight Ambiguities

For tweets containing both positive and negative signals, annotators may have different preferences. In tweet (30) below, some argued that the positive points dominated, whereas others insisted that "pointless" was the overall conclusion.

**(30) Performance and Safety feel, Of course the proud feel is awesome for #iPhone6. Still phone without charge in it is pointless**

### 3.6.2 Annotation Deficiencies

In this section, we summarize deficiencies in our annotation work. We aim to improve upon the following issues in our future work.

(1) Net Slang and Abbreviation Understanding

Net slang and abbreviations have become increasingly prevalent in social media. It is sometimes difficult even for native annotators to interpret their meanings. As an example, "is the shit" in tweet (31) below means great or awesome rather than bad. In addition, the meaning of "UA" in tweet (32) below is somewhat blurred, since "UA" is an abbreviation for many different objects. To alleviate this understanding issue, we allowed annotators to access the Internet as required; however, problems still remain owing to overall working time limits, causing annotators to not be able to search as much as they would like to.

**(31) You know technology is the shit when someone's granddad be looking to by an #Iphone6 and I ain't talking about Boondocks**

**(32) @ArianaGicPerry Apparently, #Putin said he left early it was a long flight and he needed more sleep, etc, and and no one is upset over UA!**

(2) Undefined Expression Patterns

Although we designed different types of tags to record the information related to emotional expressions in tweets, which worked well for most of the tweets we encountered, there were still some tweets that were unable to be represented by our annotation frame because their pattern of expression is beyond the scope of our definitions. We describe two subcategories below.

(a) Judging Other Opinions

Toward the end of tweet (33) below, the author judges the opinion of others. This judgment relates to the author's attitude toward iPhone 6, but the signals like "😒" and "dumb" are not its direct evaluations. To infer the author's attitude, an opinion/judgment pair frame is desirable here. Similarly, for other new patterns, specific representation frames should be designed.

**(33) People talking shit bout #iPhone6 bending and shit. Mines is perfectly straight 🍆🌽🍌 been having it for a month 😒 so stfu only dumb people**

(b) Literary expressions

In tweet (34) below, there is only one negative signal, i.e., "perversion," and no other signals or rhetorical context; however, we can still, in a way, attribute a positive attitude to the author. Such literary expression involving common knowledge may be beyond our annotation scheme; a more comprehensive scheme based on other sentiment-oriented theories is needed here.

**(34) Any version of Scotland whose finances are guaranteed by English banks is a perversion of independence.**

(3) Absence of Predefined Rules

Confusing patterns of tweets poped up repeatedly during our annotation work. As an example, how do we decide the global polarity of tweet (35) below, which simultaneously includes an issue and a solution? Similarly, how do we decide the global polarity of tweet (36) below in which the author hates part of the object but loves as a whole? The same annotator chose different answers for the same pattern from time to time due to a lack of suitable rules, which erodes the inter-annotator agreement. We handled this issue by resorting to onsite discussions; however, predefined rules in the coding manual are preferable.

**(35) Finally got the #iPhone6 talking again... But it still won't**

**let me delete texts w/o jumping thru hoops! But it works again!! :)**

(36) **While I do like my #iPhone6, I really do not like the new location of the hold button to the side vs. on top. Move it back to the top!**

## 3.7 Conclusion and Future Work

In this chapter, we described the construction of a multilingual annotated corpus for deeper sentiment understanding in social media; our corpus consists of 12 collections of tweets (i.e., three languages times four objects), with a total of 5422 tweets. We initially put forth an annotation scheme that separates emotional signals and rhetorical context for Twitter sentiment annotation, and the PDC method to improve inter-annotator agreement. As a result of these measures, the average Cohen's Kappa for global polarity of the MDSU corpus reached up to an almost perfect level (0.801), proving the high quality of the corpus.

As discussed in Section 3.6.2, there is still much room for improvement in terms of our annotation; therefore, we will continue to refine our corpus as part of our future work. Besides, our gold-standard corpus will be distributed openly in a proper way to serve the various purposes of different researchers in various fields.

# Chapter 4

# Corpus Analysis of the MDSU Corpus

## 4.1 Introduction

In this chapter, we present the analyses of the MDSU corpus[1]. In Section 4.2, we provide a basic analysis of the corpus to reveal the key differences between languages and topics. In Section 4.3, we discuss how annotattion components affect collective sentiment. In Section 4.4, we elaborate the characteristics of retoric devices and their inherent structures in the expression of feelings. Finally, we describe our conclusions and suggest future work in Section 4.5.

Based on the observations and analysis of the MDSU corpus, we have the following three findings.

(i) Languages differ in terms of their use of emotional signals and rhetoric devices, and the idea that cultures have different opinions regarding the same objects is reconfirmed (Section 4.2).

(ii) Each rhetoric device has its own characteristics, influences global polarity in its own way, and has an inherent structure that helps to model the sentiment that it represents (Section 4.4).

[1] Note that our analysis here has been conducted on the gold standard unless otherwise specified. A similar analysis of the original datasets is reported in SIG Technical Reports (2015-NL-222)[38].

(iii) Models of expression of feelings in different languages are most likely similar, suggesting the possibility of unifying multilingual opinion mining at the sentiment level (see the last paragraphs of Sections 3.4.1 and 4.4).

## 4.2 Basic Analysis of the Corpus

In this section, we describe the basic analysis of the annotated corpus. We compare components of expressions of feelings between languages, as well as public mood and people's concerns regarding the same evaluation objects in different cultures.

Table 7 presents an overview of the gold standard. From the table, we first observe that the final number of tweets in each collection fluctuated by around 450. Second, the average numbers of characters per tweet were 126.47 for English, 74.91 for Japanese, and 89.72 for Chinese, which are longer than the general Twitter average [52] owing to the selection strategy in Section 3.3. Further, considering that the information content of characters in each language differs (i.e., hieroglyphic characters usually contain more information content than alphabetic characters), we investigated the average number of morphemes per tweet; since the morpheme is the smallest unit of meaning, it is more comparable between languages. For Japanese and Chinese tweets, we used Mecab[2] and NLPIR[3] to segment the remaining text of tweets after extracting the emojis and emoticons via regular expressions; for English, we employed TweetTokenizer[4], which is customized for English tweets with the unit being a word. As shown in Table 7, the average numbers of morphemes/words for English, Japanese, and Chinese were 22.07, 32.79, and 48.55, respectively, which reverses the order of the average number of characters[5]. Note that the average numbers of characters in a morpheme/word are 4.64 for English, 2.03 for Japanese, and 1.66 for Chinese.

[2] http://taku910.github.io/mecab/

[3] http://ictclas.nlpir.org/

[4] http://www.nltk.org/api/nltk.tokenize.html

[5] Apart from normal words, special symbols in tweets, such as Unicode emojis (e.g., ❤), emoticons (e.g., :-), (((o (*°▽°*) o)))) are regarded as morphemes, but all punctuation marks are ignored.

Table 7: Basic Statistics of the Gold Standard

| Language | Object | Number of Tweets | Average Number of Characters | Average Number of Morphemes/Words |
|---|---|---|---|---|
| English | I6 | 451 | 121.13 | 23.18 |
| | W8 | 454 | 127.67 | 21.80 |
| | PU | 449 | 129.26 | 22.26 |
| | SI | 449 | 127.82 | 21.20 |
| | avg. | **450.75** | **126.47** | **22.07** |
| Japanese | I6 | 435 | 69.11 | 26.98 |
| | W8 | 465 | 68.71 | 28.56 |
| | PU | 458 | 72.46 | 35.13 |
| | SI | 443 | 89.36 | 41.29 |
| | avg. | **450.25** | **74.91** | **32.98** |
| Chinese | I6 | 450 | 91.82 | 49.46 |
| | W8 | 455 | 79.99 | 43.31 |
| | PU | 444 | 101.01 | 55.71 |
| | SI | 469 | 86.06 | 45.58 |
| | avg. | **454.5** | **89.72** | **48.45** |

### 4.2.1 Emotional Signals and Degree Modifiers

Table 8 shows the number of signals and their modifiers per tweet for each collection. As stated in Section 4.2 above, all these signals influence global polarity. First, we find that Chinese users generally use more emotional signals in a tweet than English and Japanese users; except for Scottish Independence, the sums of emotional signals for the other evaluation objects, including iPhone 6, Windows 8, and Putin, presented such a tendency. Further, the use of neutral signals was much less prevalent than that of the other two emotional signals for all three languages.

Table 8: Average Number of Signals and Their Modifiers per Tweet

| Language | Object | Emotional Signal | | | | Degree Modifier | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | Positive | Neutral | Negative | Sum | Intensifier | Diminisher | Negation | Sum |
| English | I6 | 1.12 | 0.06 | 0.69 | 1.86 | 0.54 | 0.06 | 0.22 | 0.81 |
| | W8 | 0.73 | 0.11 | 1.12 | 1.96 | 0.39 | 0.06 | 0.24 | 0.69 |
| | PU | 0.49 | 0.02 | 0.97 | 1.47 | 0.13 | 0.01 | 0.18 | 0.32 |
| | SI | 0.86 | 0.03 | 0.72 | 1.61 | 0.26 | 0.02 | 0.19 | 0.47 |
| Japanese | I6 | 0.85 | 0.05 | 0.91 | 1.81 | 0.29 | 0.04 | 0.25 | 0.58 |
| | W8 | 0.62 | 0.05 | 0.94 | 1.60 | 0.25 | 0.04 | 0.34 | 0.63 |
| | PU | 0.72 | 0.04 | 0.31 | 1.08 | 0.18 | 0.02 | 0.06 | 0.26 |
| | SI | 0.14 | 0.01 | 0.07 | 0.22 | 0.04 | 0.01 | 0.04 | 0.09 |
| Chinese | I6 | 1.78 | 0.05 | 1.33 | 3.15 | 0.79 | 0.11 | 0.43 | 1.33 |
| | W8 | 0.96 | 0.04 | 2.15 | 3.15 | 0.80 | 0.09 | 0.37 | 1.27 |
| | PU | 1.33 | 0.02 | 0.91 | 2.26 | 0.27 | 0.03 | 0.16 | 0.45 |
| | SI | 0.47 | 0.08 | 0.30 | 0.85 | 0.13 | 0.02 | 0.13 | 0.28 |

Each degree modifier modified its targeted emotional signal in its own way. As shown in Table 8, the frequency of modifiers essentially had the same order as emotional signals, with Chinese ranking first, followed by English and Japanese; again, except for Scottish Independence, the sums of degree modifiers for the other evaluation objects followed this trend. Further, the number of diminishers was much smaller than that of intensifiers in all three languages, implying that expressions of feelings in social media tend to be intense rather than reserved. As to negation, for the products (i.e., iPhone 6 and Windows 8), Chinese users (i.e., 0.43 and 0.37) used negation more than Japanese users (i.e., 0.25 and 0.34), who in turn used negation more than English users (i.e., 0.22 and 0.24); for Putin and Scottish Independence, English users (i.e., 0.18 and 0.19) used negation more than Chinese users (i.e., 0.16 and 0.13), who in turn used negation more than Japanese users (i.e., 0.06 and 0.04, respectively).

### 4.2.2 Rhetorical Context

Table 9 shows the distribution of the four rhetoric devices in each collection. We first observe that the rhetoric occurrence rate across the entire corpus was 21.9%. English users used rhetoric devices to express their feelings most often (i.e., 30.6%), followed by Chinese users (i.e., 23.9%) and then Japanese users (i.e., 11.2%). This indicates that rhetorical phenomena occur more frequently in English and Chinese tweets.

In terms of the type of rhetoric device, comparison was the most frequently used rhetoric device in all three languages, followed by rhetorical questions, sarcasm, and metaphors, implying that individuals in different cultures prefer to provide their opinions on an object by comparing it with its competitors. Further, the top two rhetoric devices (i.e., comparisons and rhetorical questions) formed the majority of the rhetoric used, accounting for 64.5%, 83.2%, and 82.6% of the total number for English, Japanese, and Chinese, respectively. In addition, the frequency of metaphors used in each language was relatively low (i.e., 13.75%, 5.75%, and 7.75% for English, Japanese, and Chinese, respectively).

Sarcasm was where the largest difference among languages occurred in terms of rhetoric use. It accounted for 26% of all rhetoric occurrences in English, but only 9.5% and 5.4% for Chinese and Japanese, respectively. This phenomenon may be attributed to the following two reasons[6]. First, it may be due to cultural

[6] People may argue that the more negative the public opinion is, the more sarcasm there is; however, we found that the proportion of sarcasm is not necessarily linearly correlated with

Table 9: Average Number of Rhetoric Devices per Collection

| Language | Object | Metaphor | Comparison | Sarcasm | Rhetorical Question | Total Number | Occurrence Rate |
|---|---|---|---|---|---|---|---|
| English | I6 | 17 | 41 | 25 | 36 | 119 | 26.4% |
| | W8 | 18 | 100 | 22 | 45 | 185 | 40.7% |
| | PU | 12 | 38 | 53 | 43 | 146 | 32.5% |
| | SI | 8 | 9 | 44 | 41 | 102 | 22.7% |
| | avg. | **13.75** | **47** | **36** | **41.25** | **138** | **30.6%** |
| Japanese | I6 | 5 | 49 | 1 | 11 | 66 | 15.2% |
| | W8 | 8 | 37 | 6 | 15 | 66 | 14.2% |
| | PU | 9 | 36 | 3 | 10 | 58 | 12.7% |
| | SI | 1 | 3 | 1 | 7 | 12 | 2.7% |
| | avg. | **5.75** | **31.25** | **2.75** | **10.75** | **50.5** | **11.2%** |
| Chinese | I6 | 7 | 101 | 14 | 39 | 161 | 35.8% |
| | W8 | 9 | 87 | 12 | 29 | 137 | 30.1% |
| | PU | 10 | 28 | 15 | 43 | 96 | 21.6% |
| | SI | 5 | 6 | 3 | 27 | 41 | 8.7% |
| | avg. | **7.75** | **55.5** | **11** | **34.5** | **108.75** | **23.9%** |

Occurrence Rate = $\frac{\#rhetoric\ devices\ in\ a\ collection}{\#tweets\ in\ a\ collection}$

factors, such as habitual patterns of expression, which can be endorsed by the fact that Western people are known for critical thinking, and that irony, as a hallmark, appears in many Western literary classics. This may explain why the occurrence of sarcasm in reference to Windows 8 varies among languages, though public opinions on Windows 8 are very close, as shown in Table 10.

Second, it may be due to subtopic composition, particularly the things that people talk about. As shown in Table 9, the amount of sarcasm shown for Scottish Independence in English was much higher than that for iPhone 6. We studied English tweets regarding Scottish Independence and found that there was a football game between Scotland and England after the independence referendum. The fact that Scotland voted "no" for independence but still booed the British national anthem brought a lot of sarcastic mocking from the English. Hence, a resolution of the sarcasm context is especially important for the English language.

### 4.2.3 Global Polarity

The collective sentiment (denoted as the PN ratio) for an object is used to represent public opinion, measuring the degree of happiness of a group of people [54, 37]. The PN ratio of object X of a collection is defined as

$$\text{PN ratio (X)} = \frac{\#\text{positive tweets of X in the collection}}{\#\text{negative tweets of X in the collection}} \tag{4–1}$$

By definition (1), if the PN ratio is greater than one, people are happy with the object, while a value less than one indicates the opposite. When the size of the collection is too small or the polarity distribution is skewed, the numerator or denominator tends toward zero. In such instances, they are set to one in practice. Table 10 shows the global polarity distribution (i.e., the number of tweets for each polarity) of each collection.

We first observe the PN ratios to determine whether there is any difference in public mood between cultures. As a hit product of the renowned Apple Inc., iPhone 6 was welcomed by English users (i.e., 1.53) and Chinese users (i.e., 1.41),

public opinion by conducting a correlation analysis between the proportion of sarcasm and public opinion.

Table 10: Global Polarity Distribution of Each Evaluation Object

| Object | Language | Positive | Negative | Neutral | PN Ratio (SD) |
|---|---|---|---|---|---|
| I6 | English | 197 | 129 | 125 | 1.53 |
| | Japanese | 120 | 187 | 128 | 0.64 |
| | Chinese | 205 | 145 | 100 | 1.41 |
| | avg. | **174** | **154** | **118** | **1.13** (0.48) |
| W8 | English | 70 | 256 | 128 | 0.27 |
| | Japanese | 57 | 250 | 158 | 0.23 |
| | Chinese | 81 | 283 | 91 | 0.29 |
| | avg. | **69** | **263** | **126** | **0.26** (0.03) |
| PU | English | 52 | 249 | 148 | 0.21 |
| | Japanese | 184 | 64 | 210 | 2.88 |
| | Chinese | 174 | 139 | 131 | 1.25 |
| | avg. | **137** | **151** | **163** | **0.91** (1.35) |
| SI | English | 184 | 140 | 125 | 1.31 |
| | Japanese | 31 | 33 | 379 | 0.94 |
| | Chinese | 106 | 71 | 292 | 1.49 |
| | avg. | **107** | **81** | **266** | **1.32** (0.28) |
| Total Number | | 1461 | 1946 | 2015 | 0.75 |

whereas Japanese users (i.e., 0.64) showed an unfavorable attitude, primarily due to its frequent malfunctions. As for Windows 8, all three cultures reached a high degree of consensus, complaining about its user-unfriendly design and experience (i.e., 0.27, 0.23, and 0.29 for English, Japanese, and Chinese users, respectively).

Individuals were evidently divided over Putin. Japanese users (i.e., 2.88) and English users (i.e., 0.21) markedly opposed one another, whereas Chinese users (i.e., 1.25) adopted a pro-center stance. People appreciate Putin for his all-round personal abilities, but dislike him for his dictatorship. Regarding Scottish Independence, both English users (i.e., 1.31) and Chinese users (i.e., 1.49) showed their support for independence. Inspecting the tweets more closely, we found that English users talked about Scottish identity, whereas Chinese users emphasized the democratic practices, sometimes as an example of schadenfreude. Japanese users (i.e., 0.94) were almost neutral on this issue. As a conclusion, we note that the three cultures had different opinions on three objects and similar opinions on one object, so it is clear that public mood variance does exist between cultures.

Further, we observed that far fewer Japanese tweets (i.e., 64) and Chinese tweets (i.e., 177) had explicit opinions on Scottish Independence versus English tweets (i.e., 324). This is in accordance with the low public attention in the former two regions discussed in Section 3.2, since the issue was more important for European people, especially those in Britain. The decreasing number of non-neutral tweets also explains the sharp decrease in rhetoric occurrences in Scottish Independence in Japanese (i.e., 2.7%) and Chinese (i.e., 8.7%) shown in Table 9. On the contrary, the percentages of non-neutral tweets are similar for the other three objects since users generally took the same stance (i.e., as outsiders for Putin and as customers for Windows 8 and iPhone 6). This suggests that user stance or interest relationship should be taken into account in a cross-cultural setting.

Last, we note that the entire corpus is well-balanced, with 0.75 inclined to the negative side, making it a suitable learning resource for three-class sentiment classification.

### 4.2.4 Subtopic Information

Table 11 shows the average number of subtopics per tweet for each collection, revealing that Japanese users (i.e., 3.55) introduce a few more subtopics than Chinese users (i.e., 3.22) and English users (i.e., 2.91). Along with the small number of emotional signals, this may imply that Japanese users focus more on sharing than on judging. Further, the average number of subtopics over the corpus (i.e., 3.23) demonstrates that although a tweet is limited to 140 characters, it still consists of approximately three subtopics. As an example, tweet (13)[7] in Section 4.5 is a tweet that contains three subtopics, i.e., screen, size, and bending. Our findings here may weaken the conclusions of topic-related Twitter research [71], which assume that a tweet has only one subtopic.

Table 11: Subtopic Number for Each Topic and Language

| Language | I6 | W8 | PU | SI | Avg. |
|---|---|---|---|---|---|
| English | 2.59 | 3.38 | 3.28 | 2.38 | 2.90 |
| Japanese | 2.64 | 3.82 | 3.86 | 3.82 | 3.54 |
| Chinese | 3.02 | 3.11 | 3.33 | 3.40 | 3.22 |
| avg. | 2.75 | 3.44 | 3.49 | 3.20 | 3.22 |

To see how subtopic components differ between cultures, the top 10 subtopics in each collection are shown in Tables 12 through 15. In Tables 12 and 13, we observe that even though there were some exceptions, the products (i.e., iPhone 6 and Windows 8) shared many similar subtopics, such as phone, acquisition, Apple, case, screen, and apps for iPhone 6 and Windows 7, laptop, user experience, updating, and Microsoft for Windows 8. This suggests that the same subtopic list can be shared among different languages for products when using topic-relevant methods.

As for Putin, the cultures seemed to have their own interest points. Although there are common subtopics, such as Obama, Russia, and US, English users pri-

[7] (13) Just picked up an #iPhone6 the screen is beautiful, but my god is it large! Crossing my fingers it doesn't bend!

Table 12: High-frequency Subtopics of iPhone 6 Tweets

| Rank | English | Freq. | Japanese | Freq. | Chinese | Freq. |
|---|---|---|---|---|---|---|
| 1 | phone | 33 | Apple | 29 | 手机 (cell phone) | 41 |
| 2 | screen | 31 | iPhone* | 28 | 弯曲问题 (bending) | 25 |
| 3 | size | 30 | ケース (case) | 27 | 三星 (Samsung) | 24 |
| 4 | case | 28 | アップデート (update) | 18 | 屏幕 (screen) | 24 |
| 5 | dropping | 25 | 機種変 (model change) | 18 | 国内上市 (domestic launching) | 22 |
| 6 | acquisition | 21 | 携帯 (cell phone) | 18 | 购买 (purchasing) | 22 |
| 7 | battery | 18 | アプリ (application) | 17 | 土豪金 (golden) | 20 |
| 8 | upgrade | 17 | 画面 (screen) | 17 | 手感 (feel) | 18 |
| 9 | Apple | 14 | Android | 14 | 换手机 (phone change) | 16 |
| 10 | camera | 13 | iPhone 5 | 14 | 外观 (appearance) | 16 |

*iPhone here means a kind of phone.

marily gave general political opinions on the Ukraine, West, and world, whereas Japanese users focused more on 柔道 (judo), 空手 (karate), and called Putin a 政治家 (politician) and 大統領 (president). Chinese users mentioned APEC and G20 summit meetings much more, likely because APEC was held in Beijing and the G20 summit was widely reported in China during our data-collection period.

For Scottish Independence, subtopics between the third-party regions (i.e., Japanese and Chinese areas) and interested regions (i.e., English areas) differed greatly. The third-party regions discussed the issue at a macroscopic level, including campaign, referendum, and England, whereas interested regions mentioned more specific subtopics, such as the Scotland versus England football match, SNP, and Westminster. What surprised us is that the largest subtopic for the English, i.e., the Scotland versus England football match, was hardly referred to by Japanese or Chinese users. Therefore, subtopic variance should be taken into consideration when developing topic-relevant methods for public figures and events.

Table 13: High-frequency Subtopics of Window 8 Tweets

| Rank | English | Freq. | Japanese | Freq. | Chinese | Freq. |
|---|---|---|---|---|---|---|
| 1 | laptop | 58 | PC | 75 | Windows 7 | 84 |
| 2 | PC | 51 | Windows 7 | 49 | 电脑 (PC) | 67 |
| 3 | Microsoft | 45 | タブレット (tablet) | 36 | 重装系统 (reinstalling) | 43 |
| 4 | updating issues | 42 | OS | 30 | 系统适应 (adaptation) | 41 |
| 5 | user experience | 37 | ユーザエクスペリエンス (user experience) | 29 | 用户体验 (user experience) | 41 |
| 6 | apps | 30 | アップデート (update) | 27 | 换系统 (system change) | 40 |
| 7 | technical issues | 28 | デスクトップ (desktop) | 24 | 兼容性 (compatibility) | 40 |
| 8 | tablet | 26 | Microsoft | 23 | Windows 10 | 29 |
| 9 | Windows 7 | 23 | 設定 (settings) | 22 | 系统更新 (updating) | 29 |
| 10 | Windows 10 | 22 | 使い慣れ (user habit) | 21 | 微软 (Microsoft) | 27 |

Table 14: High-frequency Subtopics of Vladimir Putin Tweets

| Rank | English | Freq. | Japanese | Freq. | Chinese | Freq. |
|---|---|---|---|---|---|---|
| 1 | Russia | 75 | ロシア (Russia) | 62 | 俄罗斯 (Russia) | 47 |
| 2 | Ukraine | 39 | 日本 (Japan) | 39 | G20 | 40 |
| 3 | annual speech | 19 | 空手 (karate) | 33 | 奥巴马 (Obama) | 37 |
| 4 | West | 19 | 柔道 (judo) | 31 | 美国 (America) | 31 |
| 5 | Obama | 17 | 大統領 (president) | 27 | APEC | 29 |
| 6 | world | 16 | 安倍 (Abe) | 26 | 制裁 (sanction) | 27 |
| 7 | US | 16 | オバマ (Obama) | 23 | 油价 (oil price) | 27 |
| 8 | economic problems | 13 | キレネンコ (Usavich) | 14 | 早退 (leaving early) | 21 |
| 9 | rubles | 13 | アメリカ (America) | 13 | 乌克兰 (Ukraine) | 17 |
| 10 | Russians | 13 | 政治家 (politician) | 13 | 中国 (China) | 16 |

## 4.3 Influence on Collective Sentiment

Much of the research in the field relies on the PN ratio to represent public mood regarding a certain object. Nonetheless, since the global polarity of a tweet is difficult to obtain, the word-level PN ratio is often used as a substitute for the tweet-level PN ratio [9, 71]. In this section, we verify whether this substitution is valid for use with our corpus and reveal the influence of components on collective sentiment.

For ease of reference, we use WPN to denote the word-level sentiment ratio based on polarity lexicons (see Appendix D for the dictionaries that we used); SPN to denote the sentiment ratio based on hand-labeled emotional signals, which acts as an upper bound for WPN; and GPN to denote the sentiment ratio based on hand-labeled global polarity. Note that GPN is the same as the PN ratio. By counting how many positive or negative words or signals occur in a collection, we can arrive at values for WPN and SPN. More specifically, the WPN and SPN of

Table 15: High-frequency Subtopics of Scottish Independence Tweets

| Rank | English | Freq. | Japanese | Freq. | Chinese | Freq. |
|---|---|---|---|---|---|---|
| 1 | Scotland v England | 52 | 独立投票 (independence vote) | 97 | 公投 (referendum) | 157 |
| 2 | football | 31 | イギリス (British) | 59 | 英国 (British) | 49 |
| 3 | voting no but boo | 29 | イングランド (England) | 30 | 卡梅伦 (Cameron) | 26 |
| 4 | national anthem | 28 | 独立運動 (independence campaign) | 28 | 大英帝国 (British Empire) | 21 |
| 5 | referendum | 25 | 日本 (Japan) | 14 | 爱丁堡 (Edinburgh) | 21 |
| 6 | England | 23 | カタルーニャ (Catalonian) | 13 | 英格兰 (England) | 19 |
| 7 | SNP | 22 | ウェールズ (Wales) | 12 | 民调 (polls) | 16 |
| 8 | Scotland fans | 21 | アイルランド (Ireland) | 12 | 英镑 (pound) | 16 |
| 9 | Westminster | 19 | ポンド (pound) | 12 | 加泰罗尼亚 (Catalonian) | 15 |
| 10 | game | 19 | 北海油田 (North Sea Oil) | 11 | 美国 (America) | 14 |

object X for a collection are defined as

$$\mathrm{WPN(X)} = \frac{\text{\#positive words of X in the collection}}{\text{\#negative words of X in the collection}} \tag{4–2}$$

$$\mathrm{SPN(X)} = \frac{\text{\#positive signals of X in the collection}}{\text{\#negative signals of X in the collection}} \tag{4–3}$$

### Similarities among WPN, SPN and GPN

Table 16 compares the three sentiment ratios. First, it shows that SPN has a stronger correlation and smaller gap (i.e., $r = 0.92$ on average, gap $= -0.19$ on average) with GPN than WPN does (i.e., $r = 0.76$ on average, gap $= -0.26$ on average) in all three languages; however, despite WPN being poorer than SPN, there is no statistically significant difference among GPN, SPN, and WPN (i.e., paired t-tests, all $p > 0.05$). In other words, SPN and WPN can both be possible substitutes for GPN, but SPN is more accurate. Therefore, it is acceptable to use WPN to represent public opinion in opinion-mining applications.

Table 16: Comparison of WPN, SPN, and GPN

| Language | Ratio Type | Mean | Gap with GPN | Correlation with GPN (Correlation with SPN) | *p*-value of Paired t-test with GPN |
|---|---|---|---|---|---|
| English | GPN | 0.83 | — | | |
| | SPN | 0.99 | −0.16 | 0.97 | 0.246 |
| | WPN | 1.07 | −0.24 | 0.88 (0.97) | 0.406 |
| Japanese | GPN | 1.17 | — | | |
| | SPN | 1.49 | −0.32 | 0.83 | 0.420 |
| | WPN | 1.52 | −0.35 | 0.61 (0.95) | 0.514 |
| Chinese | GPN | 1.11 | — | | |
| | SPN | 1.21 | −0.10 | 0.97 | 0.224 |
| | WPN | 1.31 | −0.20 | 0.79 (0.86) | 0.341 |

*Mean of WPN, SPN and GPN, Correlation Coefficient, and *p*-value of Paired t-tests are calculated over the 4 collections each language.

We also found that the correlation between WPN and SPN was relatively high and the gap between them was small (i.e., $r = 0.93$ on average, gap $= -0.07$ on average). Table 17 shows matching results of polarity words and emotional signals. Since emotional signals are allowed to be phrases (e.g., makes a...difference), we assume that if a polarity word hits any word of an emotional phrase, then it is a successful match[8]. Further, the polarities of both sides should be identical, meaning that if a positive word hits a negative signal, it is not regarded as a successful match.

The gap between WPN and SPN occurs primarily for two reasons. First, there was a failure in detecting emotional signals using polarity dictionaries. Table 17 indicates that the average signal matching rates reached only 44.3%, 33.4%, and 33.2% for English, Japanese, and Chinese, respectively. These mediocre results have occurred because many of the emotional phrases are composed of non-polarity words and some signals have not yet been registered in the polarity dictionaries. Second, many polarity words were mistaken as emotional signals. From Table 17, we observe that the average word mismatching rates were 53.6%, 83.0%, and 73.3% for English, Japanese, and Chinese, respectively, all of which are more than half. Here, the polarity words are not necessarily evaluating the objects, but rather can be narrative or off-topic, which accounts for the extremely high word mismatching rates for Scottish Independence in Chinese and Japanese, since both collections have a limited number of non-neutral tweets (Table 10).

Incorrectly registered non-opinionated words in polarity dictionaries can also further worsen the problem, since solutions to both problems above require high-quality polarity dictionaries. In our experience, WPN changes largely from dictionary to dictionary[9]. As for topic consistency, we regard it as an inherent gap between SPN and WPN, with WPN calculated only via simple counting, i.e., involving no topic-oriented technology. Finally, although there is plenty of room for improvement to use WPN as a proxy for GPN for all three languages, its adaptability in English is basically better than that in Japanese and Chinese.

[8] If two or more polarity words hit the same phrase, we have one match for emotional signals and two or more matches for polarity words. Hence, the intersection numbers of polarity word-based matching can be slightly larger than those of emotional signal-based matching. The numbers for both situations are listed in Table 17.

[9] Low-quality dictionaries can generate rather meaningless results, so all three dictionaries we selected have been checked manually by their providers.

Table 17: Number of Polarity Words and Their Intersection with Emotional Signals per Tweet

| Language | Object | Positive Word | Negative Word | Positive Intersection (Signal/Word) | Negative Intersection (Signal/Word) | Signal Matching Rate | Word Mismatching Rate |
|---|---|---|---|---|---|---|---|
| English | I6 | 0.98 | 0.73 | 0.52 / 0.52 | 0.36 / 0.36 | 48.8% | 48.4% |
| | W8 | 0.84 | 0.84 | 0.33 / 0.33 | 0.62 / 0.63 | 51.4% | 43.0% |
| | PU | 0.76 | 0.90 | 0.20 / 0.20 | 0.40 / 0.40 | 41.4% | 63.5% |
| | SI | 0.72 | 0.66 | 0.24 / 0.25 | 0.28 / 0.28 | 33.3% | 61.2% |
| | avg. | **0.83** | **0.78** | **0.32 / 0.33** | **0.42 / 0.42** | **44.3%** | **53.6%** |
| Japanese | I6 | 0.75 | 0.67 | 0.25 / 0.25 | 0.27 / 0.27 | 29.3% | 63.6% |
| | W8 | 0.75 | 0.74 | 0.13 / 0.14 | 0.37 / 0.37 | 32.4% | 66.1% |
| | PU | 1.76 | 0.95 | 0.29 / 0.30 | 0.11 / 0.12 | 39.2% | 84.6% |
| | SI | 2.34 | 1.12 | 0.06 / 0.06 | 0.04 / 0.04 | 47.9% | 97.1% |
| | avg. | **1.40** | **0.87** | **0.18 / 0.19** | **0.20 / 0.20** | **33.4%** | **83.0%** |
| Chinese | I6 | 1.64 | 1.45 | 0.49 / 0.49 | 0.39 / 0.39 | 28.2% | 71.7% |
| | W8 | 1.06 | 1.32 | 0.24 / 0.24 | 0.77 / 0.77 | 32.5% | 57.5% |
| | PU | 2.05 | 1.41 | 0.52 / 0.52 | 0.34 / 0.35 | 38.2% | 74.9% |
| | SI | 1.46 | 0.79 | 0.14 / 0.14 | 0.09 / 0.09 | 29.6% | 89.8% |
| | avg. | **1.55** | **1.24** | **0.35 / 0.35** | **0.40 / 0.40** | **32.2%** | **73.3%** |

$$\text{Matching Rate} = \frac{\sum \#Positive\ and\ Negative\ Signal{-}Based\ Intersection}{\sum \#Positive\ and\ Negative\ Signals\ (see\ Table\ 8)}$$

$$\text{Mismatching Rate} = \frac{\sum \#Positive\ and\ Negative\ Word{-}Based\ Intersections}{\sum \#Positive\ and\ Negative\ Words}$$

### Influence of Components on GPN−SPN

Because both GPN and SPN were calculated from the gold standard, the gap between them can be regarded as originating from the context of the tweets[10]. Hence, we can use the gap between GPN and SPN (denoted GPN−SPN) to approximate context influence. If a particular type of context has no influence on global polarity, GPN−SPN will be similar regardless of whether it is present or not. We therefore conducted a one-way analysis of variance (ANOVA) to examine the influence of the presence or absence of certain types of components (i.e., independent variables), including degree modifiers and rhetoric devices, on GPN−SPN (i.e., a dependent variable).

Table 18 shows the GPN−SPN difference and the results of our ANOVA, showing that intensifiers and diminishers together (i.e., Modifiers−Negation) had little influence on the collective sentiment ratio (i.e., $p = 0.986 > 0.05$) and that their GPN−SPN difference was trivial (i.e., $-0.003$). Conversely, the influence of negation was significant (i.e., $p = 0.004 < 0.01$)[11]. Here, the GPN−SPN of the non-negation collection was small (i.e., 0.069), while it was large (i.e., $-0.579$) for the negation collection. For rhetorical phenomena, we found that sarcasm had the same influence as negation on collective sentiment (i.e., $p = 0.001 < 0.01$); although other rhetoric devices (i.e., Rhetoric−Sarcasm) were not statistically significant (i.e., $p = 0.639 > 0.05$), their overall GPN−SPN difference was not trivial (i.e., $-0.302$).

We performed similar ANOVA analyses for each language. Table 19 details the results here by language. The table indicates that Modifiers−Negation did not have a significant influence on collective sentiment for all three languages (i.e., $p > 0.05$), as expected. Surprisingly, the influence of Negation was significant for Japanese and Chinese (i.e., $p = 0.042$ and 0.017, respectively), but not for English (i.e., $p = 0.739$). This occurred perhaps because other contexts offset the influence of negation in English. For rhetoric devices, it appears that there was

[10] We ignore quantization error (e.g., two positive tweets and one negative tweet (GPN = 2) may have five positive and two negative signals (SPN = 2.5) since our collection is quite large.

[11] Some opinions were toward opposites of Scotland in Scottish Independence (e.g., England); we temporarily regarded these opposites as negation here.

Table 18: Difference of GPN−SPN and Results of our ANOVA

| Factor | Mean of GPN−SPN (Presence) | Mean of GPN−SPN (Absence) | $p$-value |
|---|---|---|---|
| **Modifiers** | 0.024 | −0.293 | 0.087 |
| Modifiers−Negation | −0.190 | −0.193 | 0.986 |
| Diminisher | −0.173 | −0.189 | 0.942 |
| Intensifier | −0.134 | −0.206 | 0.663 |
| Negation | −0.580 | 0.075 | **0.004** |
| **Rhetoric Devices** | −0.336 | −0.034 | 0.140 |
| Rhetoric−Sarcasm | −0.124 | −0.223 | 0.639 |
| Comparison | −0.188 | −0.160 | 0.911 |
| Metaphor | −0.192 | −0.068 | 0.717 |
| Rhetorical Question | −0.139 | −0.319 | 0.313 |
| Sarcasm | −0.119 | −0.775 | **0.001** |

*Mean of GPN−SPN and $p$-value of ANOVA are calculated over all 12 collections.

Table 19: Results of ANOVA by Language

| Factor | English | Japanese | Chinese |
|---|---|---|---|
| **Modifiers** | 0.498 | 0.141 | 0.162 |
| Modifiers−Negation | 0.425 | 0.907 | 0.387 |
| Diminisher | 0.232 | 0.305 | 0.665 |
| Intensifier | 0.487 | 0.844 | 0.904 |
| Negation | 0.739 | **0.042** | **0.017** |
| **Rhetoric Devices** | **0.032** | 0.664 | **0.022** |
| Rhetoric−Sarcasm | 0.330 | 0.551 | **0.035** |
| Comparison | 0.923 | 0.733 | 0.227 |
| Metaphor | 0.732 | 0.257 | 0.131 |
| Rhetorical Question | 0.335 | 0.726 | 0.027 |
| Sarcasm | **0.002** | 0.334 | 0.064 |

$^*p$-value of ANOVA is calculated over the 4 collections of each language.

a significant difference for both Chinese and English (i.e., $p = 0.032$ and 0.022, respectively), but not for Japanese (i.e., $p = 0.664$)[12].

In addition, we also conducted a two-way ANOVA to see how negation, rhetoric, and their interaction affect collective sentiment throughout the corpus. Results show that the interaction between negation and rhetoric had little influence on GPN−SPN (i.e., $p = 0.496 > 0.05$), while GPN−SPN was significantly different in terms of the presence of both negation (i.e., $p = 0.000 < 0.001$) and rhetoric (i.e., $p = 0.013 < 0.05$). From the above analyses, it indicates that we cannot deny that either of negation and rhetoric has influence on collective sentiment.

[12] However, for some cases in which low rhetoric occurrences appear in Table 9, the GPN and WPN values (Presence) may lack representativeness; thus, their results are somehow less reliable.

## 4.4 Content Analysis of Rhetoric Devices and Their Characteristics

Wiegand et al. [90] summarized the influence of negation on sentiment in English, but little effort has been focused on discussing the role of rhetoric devices in opinion mining. Rhetoric is typically sparse in traditional long texts and it is not necessarily alter the polarity of the long text containing it even if it appears; however, for short (i.e., no longer than 140 characters) but complete opinion pieces like tweets, the presence of a rhetoric device may entirely change the polarity of a tweet; consider here the cross-sentence sarcasm context in tweet (1) described in Section 4.3 above. Further, Table 9 shows that rhetorical tweets accounted for more than 20% of the corpus, indicating that rhetoric can no longer be neglected in social media. Therefore, it is necessary to inspect the structure of each rhetoric device and clarify rhetoric influence on sentiment.

### Metaphor

Metaphor is a figure of speech in which an entity that ordinarily designates one thing (i.e., the source entity) is used to designate another (i.e., the target entity) in a different domain [34, 42]. A body of work exists on metaphor identification, but its use for analyzing sentiment is limited. Kozareva [33] proposed an N-gram method and a lexicon-based method to classify the sentiment of metaphorical sentences in political text, but both methods are heavily dependent on the hand-labeled interpretation of metaphor entities (i.e., sources and targets), which is unavailable in general applications.

Table 20 presents the contingency table for metaphor and polarity, showing that the difference between the polarity distributions of metaphorical and non-metaphorical tweets is significant*** (i.e., $\chi^2$ test, $p < 0.001$). Further, metaphor tweets only account for 2% of the entire corpus and most of them (i.e., 95.4%) are subjective tweets.

Table 21 lists typical examples of metaphor tweets and their inherent structures in the three given languages. Based on these examples, the underlined linguistic

Table 20: Contingency Table of Metaphor * Polarity

| Metaphor | | Polarity | | | Total |
|---|---|---|---|---|---|
| | | Positive | Negative | Neutral | |
| Absence | Number | 1422 | 1881 | 2010 | 5313 |
| | in Metaphor % | 26.76% | 35.40% | 37.83% | 100% |
| | in Total % | 26.23% | 34.69% | 37.07% | 97.99% |
| Presence | Number | 39 | 65 | 5 | 109 |
| | in Metaphor % | 35.78% | 59.63% | 4.59% | 100% |
| | in Total % | 0.72% | 1.20% | 0.09% | 2.01% |
| Total Number | | 1461 | 1946 | 2015 | 5422 |

clues (i.e., metaphor context) can be used for typical metaphor detection (i.e., detecting source and target entity). The polarity of the target entity (e.g., aspects of the evaluation object) depends on the polarity of the source entity. Therefore, to understand the emotion in a metaphor, the polarity of the source object must be known beforehand (e.g., from other resources).

Moreover, apart from typical metaphors, there are atypical ones. In tweet (16) below, a human knows that "tracking device" is the source entity of a metaphor for iPhone 6, but it is extremely difficult for a system to recognize this metaphor due to a lack of explicit linguistic clues. To solve this challenge, further efforts are needed.

**(16) The government invented a tracking device that every Human being will pay for & carry around at all times. Even pay a monthly fee. #iPhone6**

Further, sometimes tweets with explicit metaphor indicators are not necessarily metaphorical. In tweet (17) below, although there are words such as "like," the sense here is more of an equal comparison between Putin and Hitler rather than a metaphor. We provided the annotators with a brief introduction of rhetoric devices, but such subtle difference is intuitively distinguished by the annotators. In

Table 21: Examples of Tweets Containing Metaphors

| Language | Tweets with Metaphor | Structure |
|---|---|---|
| English | Why is **#Putin** like a wild beast? Well he & his tiger who's been killing goats in China both have no respect for international borders. #tcot | Source: wild beast<br>Target: #Putin |
| Japanese | iPhone に変えた。うん。昔の外車みたい。ミラーは折りたためないわウインカー赤だわ色々オプションやら。ヤナセががんばってくれる前の外車の感じ。でも慣れたら面白いんだらうなぁ☆ **#iPhone6** | Source: 外車<br>(foreign car)<br>Target: #iPhone6 |
| Chinese | **实**在忍受不了 **win8蜗**牛般的速度，**纠结**很久后**还**是**换**回了 7。如果人生也能像系**统**一下，可以重来，随意更**换该**多好！ | Source: **蜗**牛 (snail)<br>Target: win8 |

practice, tweet (17) was tagged as comparison by all three annotators. Compared with the typical metaphorical tweets shown in Table 21, we found that when two entities with comparable properties are in the same domain, the relation between them is probably an equal comparison.

**(17) With each ramped up aggressive speech #Putin looks more and more like Hitler. His speeches against contrived enemies are identical**

### Comparison

Compared with metaphor, comparison is used far more often in tweets. In our corpus, a tweet having a comparative relation (i.e., superior, inferior, or equal) is identified as a comparison tweet regardless of whether there is an explicit comparative expression, such as "than." In particular, contrast is regarded as a form of comparison in our annotation scheme.

Table 22 presents the contingency table for comparison and polarity, showing that the difference between the polarity distributions of comparative and non-comparative tweets is significant*** (i.e., $\chi^2$ test, $p < 0.001$). Comparison tweets account for 10% of the entire corpus and most of them (i.e., 92.9%) are subjective tweets.

Table 22: Contingency Table of Comparison * Polarity

| Comparison | | Polarity | | | Total |
|---|---|---|---|---|---|
| | | Positive | Negative | Neutral | |
| Absence | Number | 1254 | 1656 | 1977 | 4887 |
| | in Comparison % | 25.66% | 33.89% | 40.45% | 100% |
| | in Total % | 23.13% | 30.54% | 36.46% | 90.13% |
| Presence | Number | 207 | 290 | 38 | 535 |
| | in Comparison % | 38.69% | 54.21% | 7.10% | 100% |
| | in Total % | 3.82% | 5.35% | 0.70% | 9.87% |
| Total Number | | 1461 | 1946 | 2015 | 5422 |

Table 23 lists typical examples of comparison tweets and their inherent structures in the three given languages. Shown in the table as underlined text, the typical comparison context is not difficult to detect. In general, the polarity of the evaluation object (i.e., comparison base) in a typical comparison can be decided by its status relative to the comparison object. In principle, superior status means positive polarity, inferior status means negative polarity, and equal status means neutral polarity.

Ganapathibhotla and Liu [18] proposed a rule-based method for product reviews to identify preferred entities at the sentence level. Their rules are carefully designed and can be directly applied to the typical examples shown in Table 23; however, their rules are constrained to comparative patterns containing comparatives and superlatives (e.g., better and best). Atypical comparisons, for example, the past versus present comparison in tweet (18) are not addressed by their approach.

**(18) It feels like my life got restarted just the version has been changed from #xp to #Windows8 🕺🕺🕺**

Table 23: Examples of Tweets Containing Comparisons

| Language | Tweets with Comparison | Structure |
|---|---|---|
| English | Can now definitively say that **#Windows8** IS indeed faster and more stable than #Windows7 used both for a while now. Don't be afraid of 8 #fb | Base: #Windows8<br>Object: #Windows7<br>Status: superior |
| Japanese | **あいふぉん 6**よりも優良クライアントある Android欲しい | Base: あいふぉん 6 (iPhone 6)<br>Object: Android<br>Status: inferior |
| Chinese | **普京确实**不**错**，但比起正恩要稍**逊一筹**，文的、武的、**阳**的、**阴**的三胖更全面！ 最**喜**正恩白白胖胖的魔鬼身材搭配天真无邪、**阳**光**灿烂**孩童般的笑容…… | Base: 普京 (Putin)<br>Object: 正恩 (Kim Jeong-eun)<br>Status: inferior |

Further, contrast relations that span multiple sentences, e.g., "Obama vs. Putin" in tweet (19) below call for deeper methods, such as discourse analysis, since tweet polarity is not a simple sum of sentence polarities.

**(19) #Obama is quite a good orator, at the beginning of his presidency especially so. But #Putin is a COMMUNICATOR. Putin can speak AT LENGTH.**

### Sarcasm

Sarcasm conveys the opposite of the given surface meaning (Macmillan Dictionary[13]). Unlike the other three rhetoric devices, sarcasm has been studied to some degree in Twitter sentiment analysis [13, 21, 60, 80]. This is partly driven by the prevalence of hashtags, such as #sarcasm, making it relatively easy to collect sarcastic tweets in English (Section 2.2).

[13] http://www.macmillandictionary.com/ [Accessed: October 10, 2016]

Table 24 presents the contingency table for sarcasm and polarity, showing that the difference between the polarity distributions of sarcastic and non-sarcastic tweets is significant*** (i.e., $\chi^2$ test, $p < 0.001$). Sarcastic tweets account for 3.7% of the entire corpus, all of which are subjective[14]; most of them are negative (i.e., 82.9%). Only a small portion of sarcastic tweets are positive because the criticizing targets of sarcasm are not necessarily the evaluation objects. As an example, in tweet (20) below, the target of sarcasm is “the media,” which in turn expresses supportive emotion for Putin.

**(20) We’re back into #Putin bashing! Great to see such a gross manipulation from the media...never give the full picture ;-)**

Table 24: Contingency Table of Sarcasm * Polarity

| Sarcasm | | Polarity | | | Total |
|---|---|---|---|---|---|
| | | Positive | Negative | Neutral | |
| Absence | Number | 1428 | 1781 | 2014 | 5223 |
| | in Sarcasm % | 27.34% | 34.10% | 38.56% | 100% |
| | in Total % | 26.34% | 32.85% | 37.14% | 96.33% |
| Presence | Number | 33 | 165 | 1 | 199 |
| | in Sarcasm % | 16.58% | 82.91% | 0.50% | 100% |
| | in Total % | 0.61% | 3.04% | 0.02% | 3.67% |
| Total Number | | 1461 | 1946 | 2015 | 5422 |

Table 25 lists typical examples of sarcastic tweets and their inherent structures in the three given languages. Observing these sarcastic tweets, we found that all of them contain contradictory polarity pairs, suggesting that sarcasm can be recognized based on whether there is a contradictive pair of emotional signals inside a tweet. Further, linguistic hints like “😂,” “w” in Japanese tweets and short sentences ending with “!” (e.g., “Creative!”) can be indicators of sarcasm. Note that not every tweet containing a pair of different polarities is sarcastic,

[14] The only neutral tweet is a mixed one (half positive, half negative).

because the contradictoriness can be resolved by having adversatives such as but and although.

Table 25: Examples of Tweets Containing Sarcasm

| Language | Tweets with Sarcasm | Structure |
|---|---|---|
| English | So now on #windows8, any time #skype plays a sound to my speakers, it breaks all speaker sound for everything, even across reboots. Lovely. | breaks vs. Lovely |
| Japanese | #iPhone6 #修理 いったら、#画面割れ の修理だけで良かったら約13000円で済んだんやけど、水没させてるから本体丸ごと交換させられたwおかげさまで約35000円も取られたw AppleCare も入ってないし #SIM フリー やから元々全額自腹やのに… | おかげさまで (thanks to) vs. 取られた (taken) |
| Chinese | WIN8 系**统**好**吖**！ 用的感**觉**起来就是手机系**统**！ 尼**玛**的！ 台式**电脑**用起来像手机什**么节**奏！ | 好**吖** (great) vs. 尼**玛** (damn it) |

Typical sarcasm is relatively solvable because it can be explained by the tweets themselves. A much more difficult situation is when one polarity of the contradictive pair does not exist within the tweet. In tweet (21) below, even though there is only a positive signal "humble" and no contradictory pair, a human can supplement the other signal according to the unpopular behavior of Putin in the real world. This kind of sarcasm, which needs additional background knowledge, is extremely difficult to automatically detect.

**(21) Humble #Putin says it's too early to erect monuments to himself, claims those wanting to name streets after him do so out of good intentions**

### Rhetorical Questions

Interrogatives are used to seek information (i.e., answers), whereas rhetorical questions are used to emphasize claims. There are two types of rhetorical questions, i.e., one that does not need an answer and the other that is answered by the questioner himself. Most previous research on rhetorical questions is limited to their identification [8] and does not consider sentiment analysis.

Table 26 presents the contingency table for rhetorical questions and polarity, showing that the difference between the polarity distributions of rhetorical-question and non-rhetorical-question tweets is significant*** (i.e., $\chi^2$ test, $p < 0.001$). Rhetorical-question tweets account for 6.4% of the entire corpus, with the majority of them being negative (i.e., 72%).

Table 26: Contingency Table of Rhetorical Question * Polarity

| Rhetorical Question | | Polarity | | | Total |
|---|---|---|---|---|---|
| | | Positive | Negative | Neutral | |
| Absence | Number | 1371 | 1697 | 2008 | 5076 |
| | in Rhetorical Question % | 27.01% | 33.43% | 39.56% | 100% |
| | in Total % | 25.29% | 31.30% | 37.03% | 93.62% |
| Presence | Number | 90 | 249 | 7 | 346 |
| | in Rhetorical Question % | 26.01% | 71.97% | 2.02% | 100% |
| | in Total % | 1.66% | 4.59% | 0.13% | 6.38% |
| Total Number | | 1461 | 1946 | 2015 | 5422 |

Table 27 lists typical rhetorical-question tweets and their inherent structures in the three given languages. These rhetorical questions simply emphasize the emotional signals that they contain. For a question, the key task is to determine whether it is rhetorical. Frequently used questioning patterns (e.g., “Why not...?” “难道不...? (Isn’t it...?)”), the position of the question in the tweet, and the context around the question (e.g., emotional signals immediately next to the question) all help with this identification.

Apart from direct emphasis, rhetorical questions can also emphasize the op-

Table 27: Examples of Tweets Containing Rhetorical Question

| Language | Tweets with Sarcasm | Structure |
|---|---|---|
| English | last #windows8 update took more time than loading 20 #c64 games with #datasette ...what went **wrong** in 30 years? | Emphasis: wrong |
| Japanese | おはようございます　プーチン大統領が空手 8 段になったそうですね 大統領で一番**強い**んじゃないか？wwww | Emphasis: 強い (strong) |
| Chinese | iphone6 自带的短信 imessage 简直吊爆了好吗？ 别人发一段语音过来，无需打开直接拿起手机放到耳边就能播放，放完你想回复无需打开直接说话就能回复，基本告别打字了！ [good] 要么不，要么给你最好的！ | Emphasis: 吊爆 (strong, net slang) |

posite of the emotional signals. In tweet (22) below, three rhetorical questions express strong disagreement with what is being asked. It is clear that the direction of the emphasis should be taken into account when analyzing rhetorical questions for sentiment analysis.

**(22) Did #Putin start unnecessary continuous wars in Middle East?! Did Putin destroy our #Constitution?! Did Putin destroy our economy?! WAKE UP**

As discussed in Section 4.1, it is difficult to directly verify that different languages have similar sentiment-expressing models; thus, we paid close attention to the annotation process and corpus analysis to determine whether we could find any decisive evidence to contradict our hypothesis. We found that our annotation scheme fits into the three languages well during the independent annotation phase. Further, we did not find any special expression of feelings that occurred only in one language during the annotation-improvement phase; the content analysis in this section showed that rhetorical context occurred in a similar way in the three given languages. Of course, it may be too optimistic to say that we can accept the hypothesis, since our corpus did not cover all possible instances and the number of languages is limited to three; however, considering that the size of our corpus is rather large and the language families that it contains are varied, it is relatively reasonable to accept our hypothesis that different languages are likely to have similar sentiment-expressing models.

## 4.5 Conclusion and Future Work

The observations and analysis of the MDSU corpus lead us to the following three conclusions: (1) languages differ in terms of their use of emotional signals and rhetoric devices, and the idea that cultures have different opinions about the same objects is reconfirmed; (2) each rhetoric device has its own characteristics, influences global polarity in its own way, and has an inherent structure that helps model the sentiment it represents; and (3) models of expressions of feelings in different languages are most likely similar, suggesting the possibility of unifying multilingual opinion mining at the sentiment level.

We paid much attention to the agreement of global polarity in Section 3.5.3; given that the agreement of fine-grained components (i.e., emotional signals, degree modifiers, rhetorical context, and subtopics) involves so many situations (e.g., tag presence/absence, tag overlap, and tag category), we leave a detailed discussion about it for future work.

We discussed the characteristics of the four common rhetorical devices, and compared the difference of their frequencies of use between languages in this chapter. Regarding application, the annotated rhetorical information in the MDSU corpus can be applied for automatic identification of these rhetorical devices.

Returning to tweet (1)[15] from Section 3.1, we recognize that it is difficult to engineer the features of rhetorical context for machine-learning sentiment analysis. To get rhetorical contexts involved in the learning process will be our future work.

[15] **(1) Wow, with #iPhone6, you can send a message just by talking! In any voice you like. So can my mom's old rotary dial.**

# Chapter 5

# Multilingual Sentiment Analysis using Deep Learning Paradigm in Social Media

## 5.1 Introduction

The prevalence of social media has allowed for the collection of abundant subjective multilingual texts. Twitter is a particularly significant multilingual data source that provides researchers with sufficient opinion pieces on various topics from all over the world. An analysis of these multilingual opinion texts can reveal the cultural variations in public opinions from different areas. Therefore, an efficient multilingual sentiment analysis (MSA) that can process all multilingual texts (mixed monolingual texts) simultaneously is necessary.

There has been substantial research on monolingual sentiment analysis, including sentiment analysis of traditional reviews (product/movie, etc.; [57, 81, 56]) and tweets ([1, 20, 91, 50]). Instead of creating separate models for each language, an MSA should use a single model (with the same parameters for all languages) to process different texts in different languages.

However, compared with monolingual sentiment analysis, the research on MSA has progressed slowly. One of the reasons for this is that there is no benchmark dataset that supports the evaluation of MSA methods (particularly, its cross-language adaptability). As many previous studies have highlighted, open-source sentiment datasets are imbalanced [43, 14, 86, 74]: there are many freely available annotated sentiment corpora for English; however, such corpora for other languages are scarce or even nonexistent. As a compromise, many of the previous multilingual corpora have been built using human/machine translations, which are unrealistic.

In this study, we used the MDSU corpus as our training/test dataset [40]. The MDSU corpus contains three distinct languages (i.e., English, Japanese, and Chinese) and four identical international topics (i.e., iPhone 6, Windows 8, Vladimir Putin, and Scottish Independence), with 5,422 tweets in total. The multilinguality of the corpus makes it the most suitable training/test dataset for MSA.

Moreover, traditional machine learning methods that are effective in monolingual settings are not necessarily effective in multilingual settings, because they usually require heavy language-specific feature engineering that further needs language-specific resources (e.g., polarity lexicons)/tools (e.g., POS taggers and parsers). This prevents the application of many sophisticated monolingual methods to other languages, particularly the minor languages that lack basic NLP tools[1]. Until now, the most typically used methods of MSA have been based on machine translation (MT): first, texts in other languages are translated into English, and then machine-learning methods are developed based on the expanded English texts.

However, this paradigm is conditioned strongly by the quality of the MT. Considering that our processing objects—tweets—contain many informal expressions, it is even more difficult to guarantee an accurate MT. Therefore, we proposed a new deep learning paradigm to integrate the processing of different languages into a unified computation model. First, we pre-trained monolingual word embeddings separately; second, we mapped them in different spaces within a shared embedding space; and finally, we trained a parameter-sharing[2] deep neural network for

[1] Even for languages holding these tools, applying monolingual methods in MSA can possibly cause a computational burden [74].

[2] In this thesis, "parameter-sharing" specifically means that the same model parameters are shared between different languages.

MSA. Our model is presented in Figure 7.

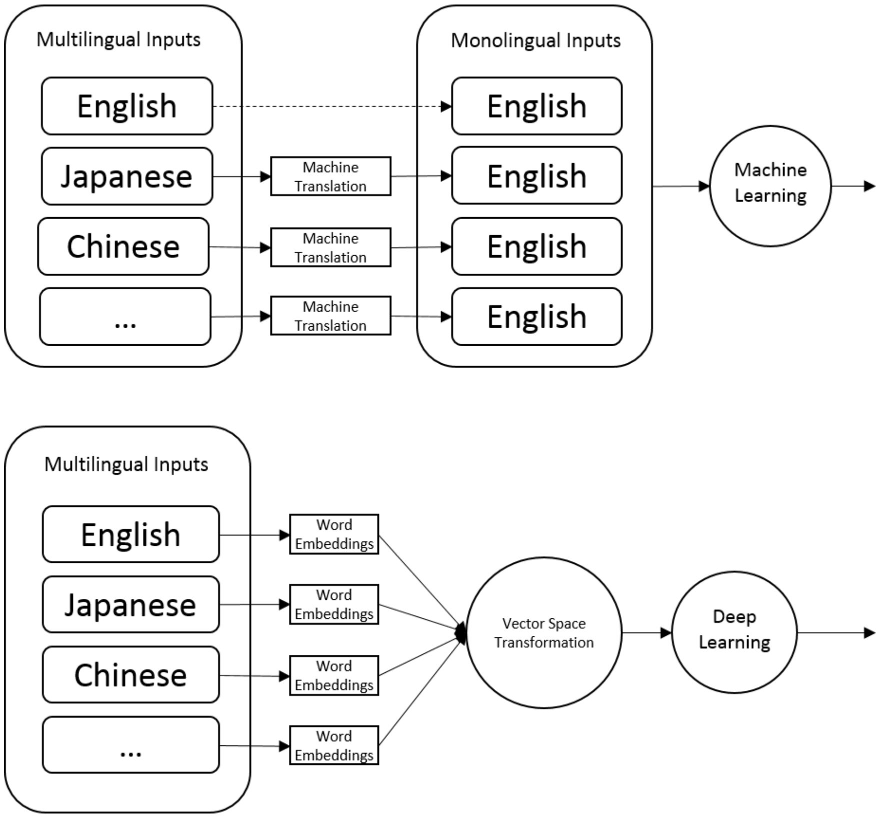


Figure 7: MT-Based Paradigm and Deep Learning Paradigm

Although the study by [63] is most similar to ours in the use of deep learning methods, there are two fundamental differences. First, they only input the raw monolingual word embeddings (an open-source, pre-trained word embedding for English and random word embeddings for other languages) in their deep learning methods; however, we used customized pre-trained word embeddings and further transferred them into a shared space. Second, they created separate models for each language, whereas we developed a single parameter-sharing model for all languages.

To the best of our knowledge, this study is the first to use a deep learning paradigm for MSA. Moreover, because of the use of such a paradigm, the only resources we required were word embeddings for each language and tokenizers for non-spaced languages (e.g., Chinese). We expected this paradigm to assimilate language differences to take full advantage of the size of multilingual datasets (compared with its smaller monolingual parts). In this study, we employed the

LSTM and CNN models. Our parameter-sharing CNN model with adjusted word embeddings outperformed the machine-translation-based baseline by nearly 5.3% and the state-of-the-art baseline by 2.1%, thereby proving its effectiveness.

This paper is organized as follows: in Section 5.2, we discuss the related studies; in Section 5.3, we describe the study methods; in Section 5.4, we presented and discussed the results of the experiments; and finally, in Section 5.5, we draw conclusions.

## 5.2 Related Work

In this section, we introduce MSA-related studies, including those on multilingual subjectivity analysis as well as the MSA of traditional text and social media.

### 5.2.1 Multilingual Subjectivity Analysis

Sentiment analysis in a multilingual framework was first conducted for subjectivity analysis. Mihalcea et al. [43] explored the automatic generation of resources (i.e., lexicon translation and corpus projection) for the subjectivity analysis of a new language (i.e., Romanian). They translated the English polarity lexicon into the target language, assessed the quality of the generated lexicon through an annotation study, and proposed a rule-based target-language classifier using the generated lexicon. The results revealed that the translated lexicon was less reliable compared with the English one, and the performance of the rule-based subjectivity classifier was worse in Romanian than in English. They also conducted a subjectivity annotation on a parallel corpus (English sentences were manually translated to Romanian); the results indicated that in most cases, the subjectivity was preserved during the translation. They projected the subjectivity onto the Romanian part to automatically obtain a Romanian subjectivity corpus and trained Naive Bayes (NB) classifiers. The results revealed that the performance of the NB classifiers in Romanian was worse than in English.

Banea et al. [4] translated the English corpus into other languages (i.e., Romanian, French, English, German, and Spanish) and explored the integration of unigram features from multiple languages into a machine learning approach for subjectivity analysis. They demonstrated that both English and the other languages could benefit from using features from multiple languages. They believed that this was probably because, when one language does not provide sufficient information, another one can serve as a supplement.

### 5.2.2 MSA of Traditional Text

Although there is extensive scope for improvement, translation-based methods have inspired many other studies. The research on MSA began relatively late. Denecke [14] translated German movie reviews into English, developed SentiWordNet-based methods for English movie reviews, and tested the proposed methods on the German corpus. The results revealed that the performance of the proposed methods in MSA was similar to that in monolingual settings. Wan [86] leveraged a labeled English corpus for Chinese sentiment classification. He first machine translated the labeled English corpora and an unlabeled Chinese corpus to the target language, and then proposed a co-training approach to use the unlabeled corpora. His experimental results suggested that the co-training approach outperformed the standard inductive and transductive classifiers. Steinberger et al. [74] annotated entity-opinion pairs in a parallel news article corpus in seven European languages—English, Spanish, French, German, Czech, Italian, and Hungarian (they first did the annotation work for English and then projected those annotations onto other languages). Their simple method to determine the word polarity aggregation for entity-level sentiment analysis was tested on the entity-opinion pairs in the parallel corpus. They created a valuable resource for entity-level sentiment analysis in a multilingual setting; however, their method, as they observed, is preliminary and depends substantially on language-specific polarity lexicons.

### 5.2.3 MSA of Social Media

Recently, the MSA of social media content has been increasing. Balahur and Turchi [3] conducted an MSA of tweets. They first translated English tweets into four languages—Italian, Spanish, French, and German (the texts in the test set were further corrected manually) to create an artificial multilingual corpus. They then tested support vector machine (SVM) classifiers using polarity lexicon-based features on various combinations of the dataset in different languages. The results suggested that the combined use of training data from multiple languages improves the performance of sentiment classification. Volkova et al. [84] constructed a multilingual tweet dataset in English, Spanish, and Russian using Amazon Mechanical Turk. They explored the lexical variations in subjective expression and the differences in emoticon and hashtag usage by gender information in the three different languages; their results demonstrated that gender information can be used to improve the performance sentiment analysis of all the three languages.

Our study is different from the previous studies in the following ways. First, in multilingual datasets from previous studies, datasets of languages other than English have been projected from the English dataset. Banea et al. [4] and Balahur and Turchi [3] have used MT to obtain texts in target languages, which are considerably noisy. Mihalcea et al. [43] and Dwnwcke [14] have directly used parallel corpora to eliminate this noise. However, real multilingual opinion texts would not be in the form of parallel corpora because users usually give their opinions in one language. Therefore, the MDSU corpus in this study includes three distant languages and covers common international topics, which is useful to test the multilingual adaptability of a method.

As for methods, Denecke [14] and Wan [86] have adopted the "MT + machine learning" approach, which unavoidably imports bias during the MT. The abstraction of the word feature in Balahur and Turchi [3] can be applied to other languages, but it requires language-specific polarity lexicons. Banea et al. [4] used unigrams in multiple languages as features, but they might be restricted due to data sparseness issues. Volkova et al. [84] proved the effectiveness of employing gender information, but their classifiers are not designed for multilingual settings. By contrast, our deep learning methods require no polarity lexicons and can unify different languages through a neural text model that uses word embeddings. In addition, the novelty of our study is not in the complexity of the network itself[3] but more in the unification of heterogeneous monolingual word embeddings and the parameter-sharing model for multilingual datasets.

## 5.3 Methods

In this section, we introduce our baseline methods and the proposed deep learning methods (i.e., transformed word embedding + deep learning). The global polarity of the MDSU corpus has three types: positive, negative, and neutral; therefore, our study is technically a three-way classification task[4].

### 5.3.1 Baselines

Our first baseline was MT-based. We used Google Translate[5] to translate Japanese/Chinese tweets into English. Google Translate is a paid service that supports more than 100 languages at various levels. For Japanese and Chinese, neural MT technology was enabled, providing more reliable translation results for the baselines.

The SVM is an efficient model for document classification. The SVM basically detects a hyperplane represented by its normal vector $\mathbf{w}$, which maximizes the margin between two classes. This search then becomes a constrained optimization problem (i.e., a solved convex quadratic programming problem), and the solution can be written as follows:

$$\mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i, \quad \alpha_i \geq 0 \tag{5–1}$$

[3] More sophisticated networks than those used in this study have been proposed for monolingual sentiment analysis.

[4] For brevity, positive/negative/neutral are denoted as +/-/= respectively, in Figures 2 and 3.

[5] https://cloud.google.com/translate/

where $\mathbf{x}_i$ are support vectors lying on the class boundaries, $\alpha_i$ are coefficients of the support vectors and $y_i$ are true values, each of which is $\in \{1, -1\}$. SVM can solve non-linear tasks using kernel trick as well.

The SVM-based learning methods with n-gram features, proposed by Pang et al. [57] and Go et al. [57], have been frequently used as baselines in many monolingual (English) studies. Similar to their settings, we used the default SVM model with a linear kernel and $C = 1$ and fed the binarized unigram/bigram term frequencies as features. The one-vs-one strategy was adopted for multiclass classification. The models were trained with LibSVM [11] via Python scikit-learn library. Following the traditional paradigm, the SVM model trained on all translated tweets in the MDSU corpus is our first baseline, denoted as MT-SVM.

In addition, we re-implemented Banea et al.'s [4] NB model that uses the cumulation of monolingual unigram features[6]. We modified Banea et al.'s method in two ways: first, we used both unigram and bigram as our features; and second, we used all the features instead of parts of them. We denoted this baseline as Banea (2010)*.

### 5.3.2 Deep Learning Paradigm

#### Space Transformation by Translation Matrix

Since there is no comparable open source word embeddings learnt from Twitter data for multiple languages, we independently obtained word embeddings using numerous monolingual texts for each language. However, these monolingual word embeddings were heterogeneous in terms of vector space (the meaning of each dimension was different between languages.). Hence, we attempted to reduce the discrepancy between monolingual word embeddings.

This notion was adopted from Mikolov et al. [45]. In their study, they highlighted that the same concepts have similar geometric arrangements in their respective vector spaces. This implies that if the matrix transformation is adequately performed, monolingual word embeddings in heterogeneous spaces can

[6] To our best knowledge, Banea et al. [4] adopted a state-of-the-art method that does not use language-specific polarity lexicons.

be adjusted to a shared vector space. Thereafter, many other ways to conduct this transformation have been proposed [62]. Following Mikolov et al. [45], we used the *Translation Matrix* method—to obtain a linear projection between the languages using a set of pivot word pairs.

Assume a set of word pairs $\{\mathbf{x}_i, \mathbf{z}_i\}_{i=1}^{n}$, where $\mathbf{x}_i$ and $\mathbf{z}_i$ are the vector representations of word $i$ in the source and target languages, respectively. We aimed to identify a translation matrix $\mathbf{W}_{S\to T}$ that minimized the following object function:

$$\underset{\mathbf{W}_{S\to T}}{\text{minimize}} \sum_{i=1}^{n} ||\mathbf{W}_{S\to T}\mathbf{x}_i - \mathbf{z}_i||^2 \tag{5–2}$$

After $\mathbf{W}_{S\to T}$ was identified, we mapped the vocabulary matrix $\mathbf{Z}$[7] of one language space to another by computing $\hat{\mathbf{Z}} = \mathbf{Z}\mathbf{W}_{\mathbf{S}\to\mathbf{T}}$. For example, we transferred the Japanese vocabulary matrix to the English vector space using $\hat{\mathbf{Z}}_J = \mathbf{Z}_J\mathbf{W}_{J\to E}$.

In this chapter, we developed two types of translation matrix: $\mathbf{W}_{J\to E}$ and $\mathbf{W}_{C\to E}$, to unify our separately pre-trained monolingual word embeddings into a shared one. We selected top $K$ high-frequent word in the English training corpus as our pivot words, translated them into Japanese and Chinese (using Google Translate), and finally obtained the translation matrices using a linear regression algorithm.

Although the linear projection by the *Translation Matrix* method can be considered as a word-level MT, the space transformation is considerably less expensive than building a full-fledged MT system.

### LSTM

RNNs have received tremendous attention in the NLP field and been employed to complete many tasks, including predicting words/phrases, speech recognition, image caption generation, and MT. RNNs are particularly effective in building language models. For example, Mikolov et al. [44] developed a statistical language model based on the Elman network [17].

Traditional neutral networks are stateless, whereas RNNs have the unique property of being “stateful”. Figure 8 illustrates the structure of a vanilla RNN. By

[7] A matrix that consists of word embeddings of all words in the training corpus.

reusing the hidden units in the previous layer, RNNs allow cyclically encoding of past information within the networks. Therefore, they can captures information from an input sequence, as it reads the sequence, one step at a time. This structure enables RNNs to process sequences of inputs with arbitrary length.

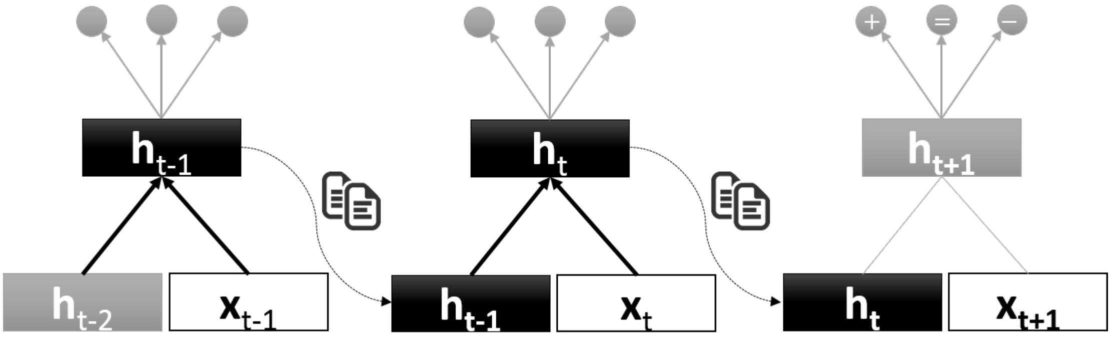


Figure 8: Network Structure of a Vanilla RNN

Let $\mathbf{x}_i \in \Re^k$ be the $k$-dimensional word vector corresponding to the $i$-th word in a tweet; then, a tweet having n words can be represented as $\mathbf{X} = (\mathbf{x}_1, ..., \mathbf{x}_T)$. A RNN consists of a hidden state $\mathbf{h}$ and an optional output $\mathbf{y}$. At each time step $t$, the hidden state $\mathbf{h}_t$ of the RNN is updated as follows:

$$\mathbf{h}_t = f(\mathbf{h}_{t-1}, \mathbf{x}_t) \tag{5–3}$$

where $f$ is a function that takes a signal $\mathbf{x}_t$ as input during time step $t$, updates its current state $\mathbf{h}_t$ based on the influence of $\mathbf{x}_t$ and the previous state $\mathbf{h}_{t-1}$.

Concretely, $\mathbf{h}_t$ and $\mathbf{y}_t$ is updated as follows in the Elman network (a vanilla RNN).

$$\mathbf{h}_t = \sigma(\mathbf{W}_1\mathbf{h}_{t-1} + \mathbf{U}_1\mathbf{x}_{t-1} + \mathbf{b}_1) \tag{5–4}$$

$$\mathbf{y}_t = softmax(\mathbf{W}_2\mathbf{h}_t + \mathbf{b}_2) \tag{5–5}$$

where $\sigma$ is a non-linear activation function, $\mathbf{W}_1$,$\mathbf{W}_2$ are the weight matrices, and $\mathbf{b}_1$,$\mathbf{b}_2$ are bias vectors.

$\mathbf{h}_t$ represents a lossy summary of task-relevant aspects of the past sequence of inputs. After the final word vector $\mathbf{x}_T$ is input in the model, the RNN reaches its final hidden state $\mathbf{h}_T$, which can be regarded as a fix-length representation of an entire tweet[8].

The distribution of the global polarity of a tweet can be determined by the softmax layer using $\mathbf{h}_T$, as follows.

$$\mathbf{y}_T = \frac{exp(\mathbf{w}_j\mathbf{h}_T + \mathbf{b}_2)}{\sum_{k=0}^{K-1} exp(\mathbf{w}_k\mathbf{h}_T + \mathbf{b}_2)} \tag{5–6}$$

where $K$ is the number of classes, $j = 0, ..., K-1$, and $\mathbf{w}_j$ are the rows of the weight matrix $\mathbf{W}_2$.

RNNs are generally trained by stochastic gradient descent using back-propagation through time (BPTT). In a vanilla RNN, such as the Elman network, during the BPTT phase, the gradient signal can end up being multiplied by the number of time steps of BPTT by the weight matrix associated with the connections between the units of the recurrent hidden layer. This means that the magnitude of values in the weight matrix can have a strong impact on the learning process. More specifically, the gradients may vanish or explode during the BPTT. If the values in this weight matrix are extremely low, it can result in a vanishing gradients situation in which the gradient gets so small that learning either becomes very slow or stops completely[10].[9] This can limit the ability of a vanilla RNN to capture long context information. Furthermore, a vanilla RNN only combines the precious hidden state $\mathbf{h}_t$ with the current input $\mathbf{x}_t$, which is not powerful enough to present a complex context.

To avoid these issues, we used an LSTM network. The LSTM model introduces a new structure called a memory block (see Figure 9). A memory block consists of four main elements: input, output, and forget gates and a self-connected cell. The cell is at the center of the LSTM memory block. Gates can be regarded as water valves, which yield values between 0 and 1, describing how much of each component should be let through. An LSTM memory block has three of these gates, to modulate the cell state.

Specifically, the input gate $\mathbf{i}_t$ controls the candidate state of the cell $\tilde{\mathbf{C}}_t$; the

[8] In addition, $\mathbf{h}_T$ acts as the context (denoted as $\mathbf{c}$) in an encoder-decoder model (a sequence-to-sequence model); it is the output of the encoder and the only input to the decoder.

[9] Conversely, the exploding gradients can cause learning to diverge (but this can be alleviated by adding regularization).

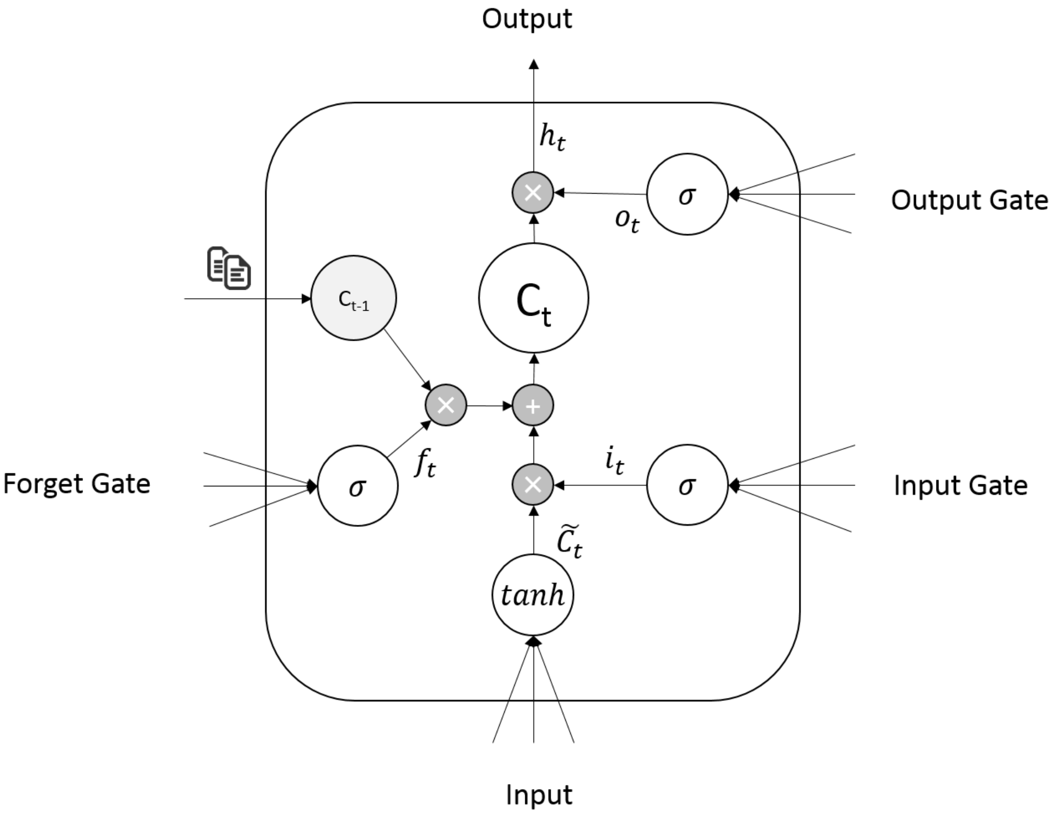


Figure 9: LSTM Memory Block

forget gate $\mathbf{f}_t$ regulates the previous state of the cell $\mathbf{C}_{t-1}$; and the output gate $\mathbf{o}_t$ determined the parts of the cell state $\mathbf{C}_t$ to output.

Eqs.(5–7)–(5–12) describe how a layer of memory blocks is updated at every time step $t$.

$$\mathbf{i}_t = \sigma(\mathbf{W}_i\mathbf{x}_t + \mathbf{U}_i\mathbf{h}_{t-1} + \mathbf{b}_i) \tag{5–7}$$

$$\tilde{\mathbf{C}}_t = \sigma(\mathbf{W}_c\mathbf{x}_t + \mathbf{U}_c\mathbf{h}_{t-1} + \mathbf{b}_c) \tag{5–8}$$

$$\mathbf{f}_t = \sigma(\mathbf{W}_f\mathbf{x}_t + \mathbf{U}_f\mathbf{h}_{t-1} + \mathbf{b}_f) \tag{5–9}$$

$$\mathbf{C}_t = \mathbf{i}_t * \tilde{\mathbf{C}}_t + \mathbf{f}_t * \mathbf{C}_{t-1} \tag{5–10}$$

$$\mathbf{f}_o = \sigma(\mathbf{W}_o\mathbf{x}_t + \mathbf{U}_o\mathbf{h}_{t-1} + \mathbf{b}_o) \tag{5–11}$$

$$\mathbf{h}_t = \mathbf{o}_t * tanh(\mathbf{C}_t) \tag{5–12}$$

where $\mathbf{x}_t$ is the input to the memory block layer at time $t$, $\mathbf{W}_i$,$\mathbf{W}_c$,$\mathbf{W}_f$,$\mathbf{W}_o$,$\mathbf{U}_i$,$\mathbf{U}_c$,$\mathbf{U}_f$,$\mathbf{U}_o$ are weight matrices, and $\mathbf{b}_i$,$\mathbf{b}_c$,$\mathbf{b}_f$,$\mathbf{b}_o$ are bias vectors.

Although LSTM memory blocks have a unique (more complicated) way of computing the hidden state (compared with Eq.(5–4)), they use the same network structure as the RNN. In our work, the classification results were decided according to $\mathbf{y}_T$.

The lengths of both hidden layer and cell layer for LSTM take the same value as the dimensionality of word embeddings.

### CNN

There have been continual debates on which model—the RNN or CNN—is more suited for NLP tasks [93]. Therefore, we use a CNN model for MSA as well.

Different from RNNs, CNNs have a bionic background. They are known to have been inspired by the human visual cortex[10]. For example, edge detection, which is a function of the primary visual cortex, can be simulated by applying convolution operation to an image [87]. In addition, although CNNs are designed for image processing, they can be used for NLP tasks. Nevertheless, CNNs for NLP tasks are generally much simpler than those for image processing.

One of the advantages of CNNs is that they have much fewer parameters than fully connected networks with the same number of hidden units, which makes them much easier to be trained. Our CNN is similar to that of Kim [30]. Our CNN model is presented in Figure 10.

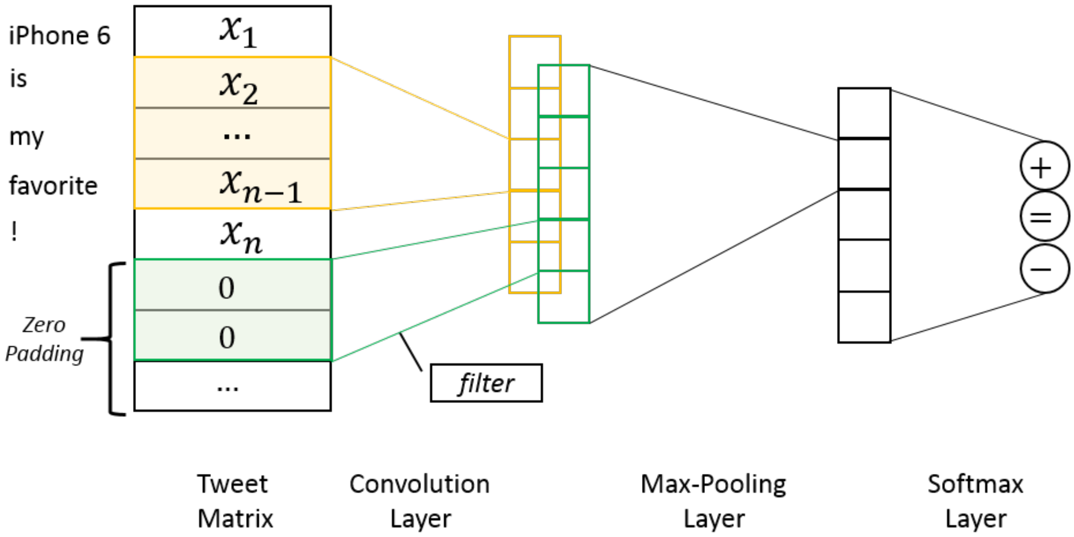


Figure 10: Network Structure of the CNN Model

[10] The visual cortex is a part of the cerebral cortex, which is crucial in processing visual information.

As in RNNs, a tweet having n words was represented as follows:

$$\mathbf{x}_{1:n} = \mathbf{x}_1 \oplus \mathbf{x}_2 \oplus \mathbf{x}_3 \oplus ... \oplus \mathbf{x}_n \tag{5–13}$$

where $\oplus$ is the concatenation operator. Here, the final index of the word vectors in a tweet was $n$ instead of $T$. In general, $\mathbf{x}_{i:i+j}$ meant the concatenation of words $\mathbf{x}_i, \mathbf{x}_{i+1}, ..., \mathbf{x}_{i+j}$.

To unify the matrix representation of tweets in different length, the maximum length of all tweets in the dataset was used as the fixed size for tweet matrices. For shorter tweets, zero word vectors were padded at the back of a tweet matrix.

The layers of the CNN are formed by a convolution operation followed by a pooling operation. Typical convolution operations for image processing include identity, edge detection, blur, and sharpening. Different operations can be achieved using different filters, $\mathbf{w} \in \Re^{hk}$; moreover, some interesting properties might be discovered by introducing random filters [87].

First, we performed a convolution operation to transform a window of $h$ words (i.e., $\mathbf{x}_{i:i+h-1}$) to generate a feature $c_i$. The procedure was formulated as follows:

$$c_i = \sigma(\mathbf{w} \cdot \mathbf{x}_{i:i+h-1} + b) \tag{5–14}$$

where $\mathbf{w}$ denotes a filter map, h is the window size of a filter, $\sigma$ is a non-linear activation function and $b$ is a bias term.

By applying filter $\mathbf{w}$ to each possible window of words in a sentence, we obtained a feature map:

$$\mathbf{c} = [c_1, c_2, ..., c_{n-h+1}] \tag{5–15}$$

Second, we performed a subsampling operation, for which we used the following max-pooling subsampling method based on the idea of capturing the most important feature from each feature map.

$$c_{max} = max\{\mathbf{c}\} \tag{5–16}$$

From Eqs. (5–14)–(5–16), a filter generated one $c_{max}$ from a tweet matrix.

The number of filter maps in our CNN model was 100, and the possible window sizes were $\{3, 4, and 5\}$; thus, our model had 300 different filters in total. The corresponding 300 $c_{max}$ formed the penultimate layer, and was then passed to a fully connected softmax layer to predict the global polarity of a tweet.

## 5.4 Experiments

In this section, we compare our deep learning methods with the baseline methods. We first describe our experimental setup, followed by a discussion of the results.

### 5.4.1 Experimental Setup

**Datasets**

As described in Section 1, we used the MDSU corpus as our training/test dataset. The MDSU corpus was originally built for deeper sentiment understanding in a multilingual setting; therefore, tweets in it were annotated many fine-grained tags in addition to global (overall) polarity. In this chapter, we used global polarities as the classification labels. [40] filtered out apparent non-emotional tweets and prioritized long tweets with rich language phenomenon during data selection; therefore, the tweets in the MDSU corpus are more complex and longer than those in randomly collected or noisy-labeled tweet datasets.

Table 1 presents the global polarity distribution for each language in the MDSU corpus. The polarity distribution of each language although not perfectly uniform, does not differ largely. Moreover, the polarity distribution of the entire corpus is well-balanced, rendering it a suitable corpus for a three-way sentiment classification. The length of a tweet is defined as the number of elements (including words, emoticons, and punctuations) after under-mentioned preprocessing. The maximum length (also the fixed size of the CNN models) of the MDSU corpus is 124: 41 for English, 93 for Japanese, and 124 for Chinese.

**Preprocessing**

The language used in social media is more casual than in traditional media. There are many unique ways of expression on Twitter, such as emoticons, Unicode

Table 28: Polarity Distribution for Each Language in the MDSU Corpus

| Language | Abbreviation | Positive | Neutral | Negative | Total # | Maximum Length |
|---|---|---|---|---|---|---|
| English | EN | 503 | 526 | 774 | 1803 | 41 |
| Japanese | JA | 392 | 875 | 534 | 1801 | 93 |
| Chinese | ZH | 566 | 614 | 638 | 1818 | 124 |
| Total | ALL | 1461 | 2015 | 1946 | 5422 | 124 |

emojis, misspelled words, letter-repeating words, all-caps words, and special tags (e.g., #, @). These may disturb the learning of word embeddings and classification models; therefore, we preprocessed them to unify the elements in different shapes but with same meanings as much as possible.

For all the three languages, we detected Unicode emojis and replaced them with an "EMOJI_CODE" (e.g., we replaced "❤" with "EMOJI_2764"); detected emoticons from easy :-) to complex (((o (*▽*) o))))) using regular expressions[11] and replaced them with "EMOTICON"; and labeled URLs as "URL").

We also performed language-dependent preprocessing. For English, we lowercased English characters and tokenized the tweets with TweetTokenizer[12]; for Japanese, we normalized Japanese characters and tokenized the tweets with Mecab[13]; for Chinese, we transferred traditional Chinese characters to simplified Chinese characters and tokenized the tweets with NLPIR[14].

### Word Embeddings and Translation Matrix

In addition to the annotated MDSU corpus, we accumulated large collections of raw tweets using Twitter RESTful API by the same query keywords during a one-year period (see Section 6.6.1). We first excluded undesirable tweets (e.g., tweets starting with "RT") using the same veto patterns as [40]; then, we checked the

[11] We registered some rare expressions to an ad hoc list.

[12] http://www.nltk.org/api/nltk.tokenize.html

[13] http://taku910.github.io/mecab/

[14] http://ictclas.nlpir.org/

preceding 10 tweets to delete the repeating tweets, because similar tweets usually appear in succession. After filtering out the undesirable tweets, the remaining tweets were preprocessed as previously described. The number of remaining tweets was 232,214 (EN), 264,179 (JA), and 148,052 (ZH). The vocabulary size for each collection of tweets was 63,343 (EN), 49,575 (JA), and 52,292 (ZH).

Our vector representation for words was learnt using FastText[15]. Because the scale of our corpus for word embedding training was relatively small, we set the minimal number of word occurrences as 2. We used the skip-gram model because it generates higher quality representations for infrequent words [45]. The word embeddings for each language were trained separately on its corresponding corpus. Words that were not present in the pre-trained word list were initialized randomly in the deep learning models.

The dimensionality of our word embeddings was 100, and the Japanese/Chinese spaces were transformed by their respective translation matrices. For the translation matrix, we set k as 3500, which implied that the top 3500 English words and their translations were the pivot word pairs. We split the 3500 pivot word pairs into two sets—training set (3000 words) and test set (500 words). The translation matrices were obtained based on the training sets. As a validation, we calculated the change of Euclidean/cosine distances for each word pair in the test set before and after the mapping; Table 29 depicts the decrease in the sum of the two distances.

Table 29: Sum of Embedding Distances of Word Pairs in the Test Set

| Language | | Before Mapping | After Mapping |
|---|---|---|---|
| Japanese | Euclidean Distance | 2,455.94 | 2,135.51 |
| | Cosine Distance | 458.37 | 440.82 |
| Chinese | Euclidean Distance | 2,457.05 | 2,098.76 |
| | Cosine Distance | 496.01 | 490.88 |

[15] https://github.com/facebookresearch/fastText

**Model Hyper-parameters**

All the methods were tested using 10-fold cross validation. For the deep learning models, we randomly selected 10% of the training splits of cross-validation as the developed datasets to tune parameters for an early stopping.

For fair comparison, we empirically set the hyper-parameters for deep learning models as consistent as possible. Both trainings were completed using a stochastic gradient descent (SGD) algorithm for shuffled mini-batches with the Adadelta update rule, with a mini-batch size of 50. The dropout technique is effective in preventing co-adaptation of hidden units by randomly setting a portion of the hidden units to zeroes during feedforward/backpropagation. Therefore, to prevent overfitting, we employed the dropout technique for both deep learning models on their penultimate softmax layers, with a dropout rate of 0.5. We did the same for the dimensionality of word embeddings; the lengths of both the hidden and cell layer for LSTM were 100.

### 5.4.2 Result and Discussion

### 5.4.3 Baselines

Table 30 presents the classification accuracies of baselines.

According to Table 30, the average accuracy of separate SVM classifiers over original datasets was the same as it over translated datasets. This showed that the same method did not necessarily perform worse after being translated by MT for monolingual datasets. In addition, the performance of MT+SVM model (use all translated tweets) was worse than the average accuracy of separate SVM classifiers over original datasets (53.0% vs. 54.5%), showing the limitation of traditional paradigm(i.e., “MT + machine learning”).

For classifiers directly used the cumulation of unigram and bigram, both SVM and Banea (2010)* performed better than MT+SVM by 0.8% and 3.4%, respectively. The increases indicate that the use of cumulation of n-gram is effective; although this may result in the problem of data sparseness [4], it could be mitigated by feature selection.

Table 30: Results of Baselines

| Model | Dataset | Feature | Accuracy |
|---|---|---|---|
| Average | – | – | 0.545 |
| SVM | EN | unigram+bigram | 0.529 |
| SVM | JA | unigram+bigram | 0.596 |
| SVM | ZH | unigram+bigram | 0.509 |
| Average | – | – | 0.545 |
| SVM | (Translated) EN | unigram+bigram | 0.529 |
| SVM | Translated JA | unigram+bigram | 0.591 |
| SVM | Translated ZH | unigram+bigram | 0.515 |
| MT+SVM | Translated ALL | unigram+bigram | **0.530** |
| SVM | ALL | cumulation of unigram+bigram | 0.538 |
| Banea (2010)* | ALL | cumulation of unigram+bigram | **0.564** |

### 5.4.4 Deep Learning Methods

Table 31 presents the classification accuracies of the deep learning models; the input of word embeddings for the models in this Table involved no transformation.

First, our deep learning paradigm performs better than the MT+SVM method (traditional paradigm). Specifically, parameter-sharing LSTM and CNN models outperformed MT+SVM model by 1.2% and 4.3%, respectively. Thus, the deep learning paradigm is more efficient than the traditional paradigm. In addition, the LSTM performed worse than the Banea (2010)* baseline, whereas the CNN excelled. Thus, CNN is more suitable for MSA than LSTM.

Besides, we also conducted the learning separately on each language split. The results revealed that the average accuracies of separate LSTM/CNN classifiers were a little higher than the accuracy of the mixed case (54.4% vs. 54.2%, and 58.1% vs. 57.3%), implying that the deep learning methods did not improve after using the entire dataset. This was a result of the heterogeneity of vector spaces of word embeddings, because the raw word embeddings were learned separately.

Furthermore, we observed that both MT + LSTM and MT + CNN models (trained on the translated datasets and using only English word embeddings) performed worse than the LSTM and CNN models (trained on the original datasets and using multilingual word embeddings). Ideally, if JA/ZH were perfectly translated, the performance should have increased. This suggests that the noises that MT brings in are greater than the heterogeneity of multilingual word embeddings does.

### 5.4.5 Deep Learning Methods using Transformed Word Embeddings

The unification of different vector spaces was expected to further improve the deep learning paradigm. Table 32 presents the classification accuracies of the deep learning models before and after space coordination. According to Table 32, the effectiveness of LSTM and CNN models were divided. We observed that after space transformation, the accuracy of LSTM decreased by 0.6%, whereas

Table 31: Results of Deep Learning Models

| Model | Dataset | Accuracy |
|---|---|---|
| Average | – | 0.544 |
| LSTM | EN | 0.531 |
| LSTM | JA | 0.569 |
| LSTM | ZH | 0.532 |
| MT+LSTM | Translated ALL | 0.541 |
| Parameter-sharing LSTM | ALL | **0.542** |
| Average | – | 0.581 |
| CNN | EN | 0.578 |
| CNN | JA | 0.610 |
| CNN | ZH | 0.553 |
| MT+CNN | Translated ALL | 0.564 |
| Parameter-sharing CNN | ALL | **0.573** |

the accuracy of CNN increased by 1.4%. This suggests that the same vector space transformation does not necessarily suitable for different kinds of network structures.

Overall, the performance of the CNN model fed with transformed word embeddings was most effective.

Table 32: Results of Deep Learning Models Before and After Space Transformation

| Model | Dataset | Word Embedding | Accuracy |
|---|---|---|---|
| Parameter-sharing LSTM | ALL | Raw (Table 31) | **0.542** |
| | ALL | Transformed | 0.536 |
| Parameter-sharing CNN | ALL | Raw (Table 31) | 0.573 |
| | ALL | Transformed | **0.587** |

## 5.5 Conclusion and Future Work

In this chapter, we proposed a novel deep learning paradigm for MSA. We map monolingual word embeddings into a shared embedding space, and used parameter-sharing deep learning models to unify the processing of multiple languages. The tests on a well-balanced tweet sentiment corpus—the MDSU corpus—revealed the effectiveness of our deep learning paradigm. Especially, our CNN model fed with translation matrix-transformed word embeddings achieves a rise of 2.3%, comparing with the strong Banea (2010)* baseline.

Our paradigm provides a great cross-lingual adaptability. Training tweets in any other language can be transferred into vector representation using transformed word embeddings, and then combined with the learning process of the deep learning models.

This study had certain limitations: some components of our paradigm were relatively simple. In the future, we plan to attempt more complex transformation methods and network structures. Moreover, pre-trained monolingual word embeddings can be further tuned using word-level polarities of words in the context that provided in the MDSU corpus. Finally, unsupervised text tokenizers, such as SentencePiece[16], may liberate us from using any language-specific tokenizers, which makes the proposed paradigm for MSA completely language-independent.

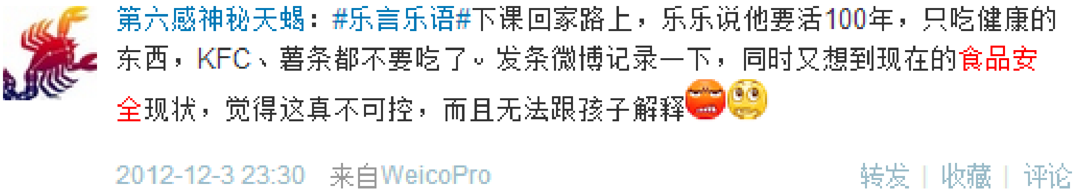
[16] https://github.com/google/sentencepiece

# Chapter 6

# Predicting Sector Index Movement with Microblogging Public Mood Time Series on Social Issues

## 6.1 Introduction

Social media, such as Twitter and Facebook, generate a great number of opinionated texts on a variety of social issues, especially hot or emergency events. Valuable knowledge can be extracted by the mining of the UGC. For example, using public mood entailed in the real-time message streaming, researchers have proposed a wide range of applications, such as election forecast [79], anti-terrorism assistance [12] and consumer confidence survey [54]. In this chapter, we also pay attention to public mood, but use it for stock prediction.


第六感神秘天蝎：#乐言乐语#下课回家路上，乐乐说他要活100年，只吃健康的东西，KFC、薯条都不要吃了。发条微博记录一下，同时又想到现在的食品安全现状，觉得这真不可控，而且无法跟孩子解释

2012-12-3 23:30 来自WeicoPro

Figure 11: Example Tweet of Sina Weibo

As described in Section 3.3, Sina Weibo is a Twitter-like microblogging service in China. Launched in 2009, it now has near 200 million monthly active users[1], which makes it the most dominant social networking service in China. Users discuss all kinds of social topics and express their opinions on the platform. As an example, food safety issue has become a serious social problem and caused much concern in recent years in China. Figure 11 is an example tweet talking about food safety from Sina Weibo, in which the author expresses his dissatisfaction to the situation of food safety in China. Note that besides the text part, there is auxiliary information around the text (called surrounding information).

Previous work showed that indicators from real-time media could conceivably be used to predict changes for many economic indexes [9], and behavioral finance theory suggested that public mood could drive stock market [53]. Hence, we construct public mood time series by analyzing millions of tweets during a time span to predict stock movement in the same period.

Our main contributions are summarized as follows:

- We investigate how microblogging public mood on a certain social issue relates to the stock movement of its relevant sector. In this study, we conduct an experiment on the tweets whose topic is "food safety" from Sina Weibo and Shenzhen Stock Exchange (SZSE) Food & Beverage Index.
- We utilize not only the text part of a tweet, but also the non-text part, namely surrounding information and user information, and show that both sentiment classification and public mood time series can be improved using it.
- We study how the proposed method performs for different period types of stock index. Both cross-correlation coefficient (CCF) and vector auto-regression (VAR) evaluations show that our public mood time series has a better predictive power during fluctuating period than monotonous period.

To the best of our knowledge, this work is the first to predict sector stock index by public mood time series on social issues in Chinese microblogging.

[1] According to financial results for first quarter of 2015 released by Sina Weibo Corp.

## 6.2 Related Work

With the popularity of real-time social media, stock price prediction based on tweets has attracted more and more attention. Past work can be roughly categorized into two classes depending on whether sentiment is used or not.

One class is sentiment-based methodology using general tweets. Bollen et al. [9] generated seven different public mood time series using OpinionFinder and Google Profile of Mood States (GPMOS). Both Granger causality analysis with Dow Jones Industrial Average (DJIA) and a self-organizing fuzzy neural network predictor showed that "calm" dimension had the best predictive effect. Vu et al. [85] experimented with a decision tree classifier with different combinations of features to predict the daily up-and-down movement of the stock price of tech companies. They proved that positive/negative sentiment, bullish/bearish orientation, and stock price change in three previous days were effective features. Si et al. [71] proposed a topic-based method called continuous Dirichlet process mixture model to learn subtopics, drew sentiment time series by aggregating opinion words over the topic chains. The VAR analysis with Standard & Poor's 100 showed the effectiveness of their method.

The other class is non-sentiment-based methodology using financial tweets. Bar-Haim et al. [5] distinguished expert users from non-experts according to the correctness of stock rise prediction against one's bullish posts. The precision of predicting stock rise showed that the per-user Model after expert classification performed better than other pattern-based methods. Ruiz et al. [64] represented financial tweet sets as graphs, and extracted activity features and graph features. The correlation analysis with stock market activities showed that the number of connected components was the best feature, and the correlation with traded volume was stronger than stock price.

Our method belongs to the former. The main difference from previous work is that our public mood time series is based on message-level sentiment analysis on general tweets, and we creatively take non-text information into consideration. Besides, unlike Bollen et al. [9] predicting composite index value or Vu et al. [85]

forecasting individual company stock price, we observe how public mood on social issues affects stock movement at the sector level.

As to the method for (monolingual) sentiment analysis, concerning BoW features unavoidably cause data sparseness problem, similar to Xie et al. [92], we use a SVM classifier with microblog-specific low-dimensional features due to its flexibility and efficiency. However, unlike previous work that only employs the text part of a tweet, we also use non-text information, such as the number of retweet and the number of reply.

## 6.3 Approach Outline

The overall framework of our research is shown in Figure 12. The core of our method is to build a sound public mood time series curve from tweets. This includes two main steps — bullish/bearish orientation representation and daily mood indicator design. Regarding the manifestation of bullish/bearish orientation, instead of using lexicon-based word-level collective sentiment of general tweets [9, 71] or explicit buy/sell transaction of stock tweets [5], we utilize tweet-level collective sentiment of general tweets, since global polarity of a tweet contains more accurate emotion about its related object and general tweets allow us to have a wider base [85]. In our study, tweets are divided into three categories: ‘positive’, ‘negative’ and ‘neutral’. A positive tweet can be a potential ‘bullish’ signal for stock price, and a negative message can be a potential ‘bearish’ signal.

To have a better message-level sentiment classification, we train a customized classifier for our selected topic instead of using existing general tools (e.g. OpinionFinder). We first extract text features and non-text features from tweets and feed the classier with different combinations of them to find the best classifier. Using the customized classifier, we then obtain the global polarity of each tweet. Rather than using simple sentiment ratio as daily mood, we take non-text information into account to design a weighted daily mood indicator. The public mood time series curve can be easily drawn once we have weighted daily mood values of each day. We adopt two different perspectives to evaluate the prediction ability

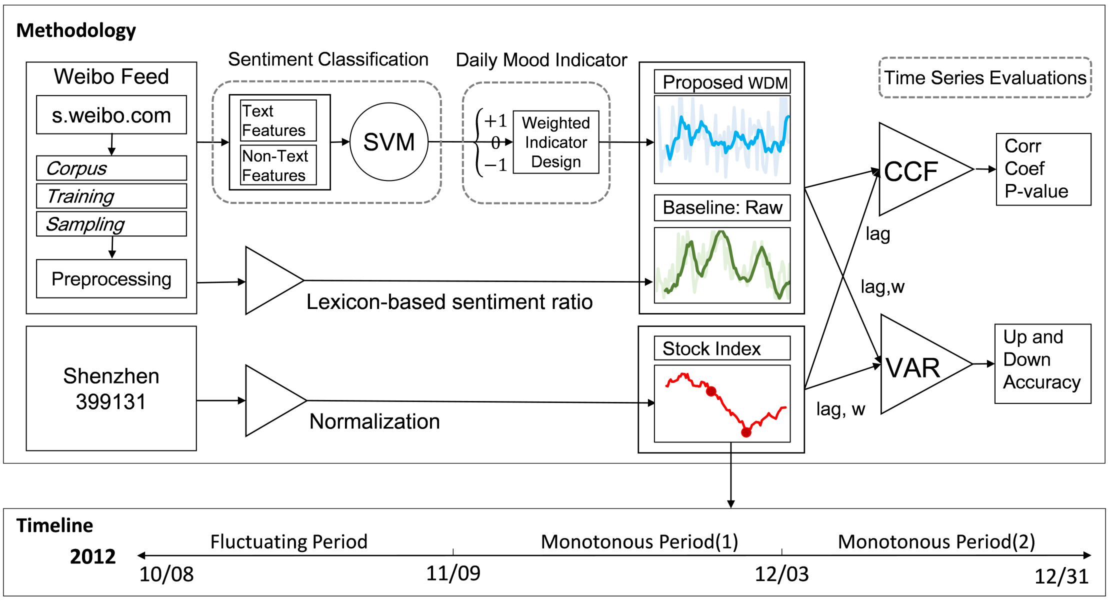


Figure 12: Overview of the Stock Prediction Research

of mood curves—CCF and VAR. Moreover, as shown at the bottom of Figure 12, the stock index is divided into one fluctuating period and two monotonous periods according to the degree of volatility. We will compare how differently mood curves perform during the two kinds of time periods.

## 6.4 Customized Sentiment Classification

Both Pang et al. [57] and Go et al. [20] reported that SVM outperformed other classifiers using n-gram and POS features and that unigram feature worked the best for both traditional and Twitter sentiment analysis. Therefore, we choose SVM as our classier. Given the limited length of microblogging (no more than 140 characters), n-gram and POS features lead to severe data sparseness problem, so we design our microblog-specific features for the SVM classifier. In this chapter, our SVM classifier is trained with LibSVM toolkit using RBF kernel [11].

### 6.4.1 Text Features

Besides traditional text features such as n-gram, POS tags and the number of polarity words, there are many microblog-specific features in the text part of a tweet[2].

**Entity Number** Entities are special elements in tweets. We exploit four kinds of high-frequency entities: hashtag, @ tag, URL and seed. The former three are the same as Twitter tweets, while the last one is a Sina Weibo tweet-specific entity which allows users to subscribe RSS news about certain tagged words. The number of the four kinds of entities are used as features. These features were also used in previous work.

**Set-count Neutral Signals** Based on observation of many tweets, we collected neutral signals for identifying objective tweets. The more neutral signals a tweet contains, the more possible it is objective. The neutral signals consist of two subsets. One subset includes: bracket pair ( 【】 ), book title mark ( 《》 ), time patterns (e.g. *月*日) and numbers symbols (e.g. 35%), and the other contains 5 types of words: news words (e.g. 宣传日), Q&A words (e.g. 科普), stock terms (e.g. 沪指), sharing words (e.g. 下载), and irrelevance words (e.g. 抽奖). Neutral signals are set-count features[3], so there are two of them.

**Sentence Number** Unlike English tweets, Chinese tweets can easily have 3 or more sentences, so sentence information is important for Chinese tweets. We count the number of sentences, the number of exclamatory sentence indicated by exclamation marks, and the number of questions indicated by question marks.

The sum of the polarities of polarity words is the basic way to measure the sentiment of a sentence or a tweet, so we compute sentiment scores at both the sentence and tweet level. They are defined as:

$$Score(U) = \sum_{i=1}^{|U|} polarity(i) \tag{6–1}$$

where $U$ denotes a unit of text and $i$ denotes a word or an emoticon whose polarity is in $\{1, 0, -1\}$.

**Sentence Sentiment Score** The first sentence and the last sentence are always more important than others. Thus, we compute sentiment scores of them respectively. First, we clear up tags (entities, emoticon etc.) and normalize abnormal full stops in a raw tweet, then tokenize the cleaned tweet using NLPIR and

[2] For brevity, “tweet” means Sina Weibo tweet unless otherwise specified in this section ,

[3] A set-count feature counts the occurrence number of the elements in a set.

segment it into sentences by punctuations (period, semicolon, exclamation mark, question mark, and suspension points). Second, we turn sentences into word polarity vectors and compute the sentence scores by summing up all the values in the vectors. For example, "各**种** | 食品安全 | **问题** | 集中 | 爆**发** |， | 有些 | 是 | **问题** |， | 有些 | 是 | **误**解 |。" is transformed to $[0, 0, -1, 1, -1, 0, 0, 0, -1, 0, 0, 0, -1, 0]$. This calculation relies heavily on the quality of polarity dictionaries. There are three open-source polarity dictionaries for Chinese: Hownet dictionary[4], DTU ontology dictionary[5], and NTU dictionary[6]. By comparing the effectiveness of these lexicons and their combinations on a small test set, we use the union set of all of them.

**Tweet Sentiment Score** We compute two global sentiment scores by emoticons and polarity words, respectively. Emoticon is such a special reference for noisy labeling [55] and a strong indicator of global polarity [32] that we consider it separately. Unlike the emoticons in English that consist of ASCII characters, Sina Weibo emoticons are icons. Thus, we first classified 72 high-frequency emoticons in Sina Weibo into 3 categories (i.e., positive, negative and neutral), then sum up their polarities as one of the global sentiment scores. For example, there are two emoticons at the end of the example tweet (see Figure 1). The other global sentiment score by polarity words is computed in the same way as the sentence sentiment score.

### 6.4.2 Non-Text Features

Apart from text features, there are many metadata about a tweet (surrounding information) and the author of it (user information). Previous studies have not made full use of these metadata. Since raw tweets are stored in HTML pages, metadata enclosed by HTML tags can be extracted by an HTML parser. We extract tweet ID, user ID, user badge, user nickname, sending date, sending source, the number of retweet, the number of replies, and the state of embedded picture and video. Some of the metadata are just identifiers with little meaning such as

[4] http://www.keenage.com/html/c_bulletin_2007.htm

[5] http://ir.dlut.edu.cn/EmotionOntologyDownload

[6] http://www.datatang.com/data/11837

tweet ID and user ID, while others can potentially be useful features.

**Surrounding Information** Surrounding information refers to the fields below the text part of a tweet (see Figure 11). In our study, user badge, the number of retweet, the number of replies, and the state of embedded picture and video are selected as features.

**User Information** We can access user information using Sina Weibo user interface by user ID. Many attributes such as gender, city, badge, and brief introduction about the user can be fetched. We only use the three numeric fields: the number of follower, following and posted tweets.

## 6.5 Daily Mood Indicator Design

Bollen et al. (2011) has shown that daily WPN ratio time series can represent public mood and emotionally responded to hot social events. Different from Bollen's curves based on word-level collective sentiment, our time series are built on tweet-level collective sentiment (i.e., GPN ratio).

Considering the global polarity distribution of the tweets in our experiment is skewed at the tweet level (due to there are very few positive tweets on food safety problem), we use Eq. 6–2 as our basic daily mood indicator instead. It also means the degree of happiness and is monotonically decreasing (the more there are negative tweets, the less it will be). The public mood of day $t$ (denoted as Daily Mood or DM) is defined as:

$$DM(t) = \frac{\#_t(tweet)}{\#_t(tweet_-)} \tag{6–2}$$

where $\#_t(tweet)$ denotes the number of tweets in date t and $\#_t(tweet_-)$ denotes the number of negative tweets in date t.

Different tweets have different weights. A tweet that has many retweets or posted by famous people is considered to have a stronger impact on public mood and then on stock market. Therefore, we need to take these useful non-text information into account. The weighted daily mood $WDM(t)$ and $Weight(t)$ are represented as:

$$WDM(t) = DM(t) * Weight(t) \tag{6–3}$$

$$Weight(t) = log_2(\frac{\sum_t(retweet)}{\sum_{t,-}(retweet)}) * \frac{lg(\sum_t(follower))}{lg(\sum_{t,-}(follower))} \tag{6–4}$$

where $retweet$ means the number of the retweets of a tweet and $follower$ means the number of the followers of the author of a tweet. We compute the total number of them in day $t$. Besides, $follower$ is log-transformed since $follower$ is much greater than $retweet$, and so is the product in Eq. (6–4) for order reduction.

## 6.6 Experiment on Sentiment Classification

### 6.6.1 Text Data

In the same way as described in Section , we scraped tweets discussing food safety with the keyword “食品安全” (food safety in Chinese) from its search service platform[7]. The collection period is the fourth quarter of 2012 (Oct. 1st, 2012–Dec. 31st, 2012) when food safety problem was the most concerned problem for Chinese people. In total, we fetched 51,611 pieces of tweets (denoted as Corpus).

A training dataset was annotated for the SVM classifier (denoted as Training). We tagged the global polarity of each tweet in Training in the same way as described in Section , so all the tweets in Training were tagged with one of $+1, 0, -1$. Training consists of 901 pieces of tweets, coming from a randomly selected date.

### 6.6.2 Sector Index

In order to evaluate our public mood time series, a sector stock index for food industry is needed. We select SZSE Food & Beverage Index 399131 (denoted as Index) as our stock index. Index consists of 56 main companies in the food sector of China. The period of Index corresponds to Corpus's collection period (Oct. 1st, 2012–Dec. 31st, 2012)[8]. To make it continuous, the values at weekends is generated by linear interpolation[9].

Figure 13 shows the Index curve (in order to compare with mood curves, the curve is z-score normalized). As we can see, there are continuous decline or in-

[7] http://s.weibo.com/

[8] Unfortunately, 399131 Index has been delisted from Mar. 1st, 2013.

[9] Since Oct. 1st–Oct. 7th are national holidays of China, we ignore these days.

crease periods in the curve. On one hand, these long-term monotonous (stable, in a sense) movement will render prediction more difficult since public mood usually changes drastically. On the other hand, prediction in long-term monotonous periods is less meaningful than it in fluctuating periods for stock investors. Therefore, it is necessary to discuss prediction in two types of periods: fluctuating period (e.g., Oct. 8th–Nov. 9th) and monotonous period (e.g., Nov. 10th–Dec. 3rd, and Dec. 4th–Dec. 31st).

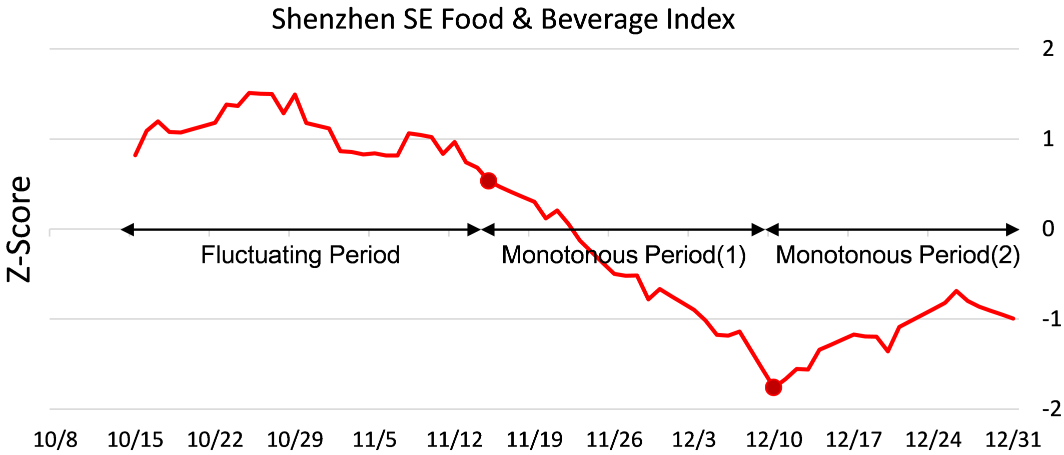


Figure 13: SZSE Food & Beverage Closing Values (Oct. 8th–Dec. 31st)

### 6.6.3 Classification Results

The SVM classifier with unigram for sentiment classification described in [20] was used as a baseline. We employed WEKA[10] to construct the unigram model, and classified tweets by its embedded LibSVM. We tried three combinations of our features described in Section 6.4. The validation method is 5-fold cross-validation. Table 33 show the accuracy of each model.

From Table 33, we find:

1. C* classifiers perform better than the baseline by 10.1% on average. In addition, the numbers of the dimension of C* classifiers are much less than the baseline, which saves learning time. The result also implies that the classification methods based on BoW feature have limitation for Twitter sentiment analysis,

[10] http://www.cs.waikato.ac.nz/ml/weka/

Table 33: Results of Different Models

| Features | #Dim | Precision |
|---|---|---|
| Baseline: unigram | 2517 | 79.69% |
| C1: text features only | 13 | 89.79% |
| C2: C1 + surrounding info | 17 | **92.23%** |
| C3: C2 + user info | 20 | 87.35% |

because a word is not necessarily an emotional signal. Hence, although the feature dimension is very high, each of the features does not contribute much. On the contrary, each of our customized features has its underlying influence on the global polarity.

2. C2 is higher than C1 by 2.44%, and C3 decreased by 4.88%. This suggests that surrounding information improves the classification, while user information does not. This makes sense because we know that controversial tweets having many retweets or replies are more likely to be emotional. On the contrary, user information is not only different from other features in magnitude, but also incompatible with them in quality, resulting in disturbing the learning.

As a result, we utilized C2 as the final model. Let us look at the accuracies of different categories. The accuracy for neutral class reaches an impressive 98%, and 72.3% for negative class, both of which are higher than them of [92][11]. Besides, public mood on social events goes to extremes easily. The majority subjective class in Corpus is negative, because public mood for food safety in China was irritated at the data collection period. There are only 8 positive tweets in Training and only 1 of them is classified correctly. Consequently, the prediction for positive tweets is unreliable. In fact, according to a manual check, the positive tweets account for less than 1% of Corpus. This is why we changed the definition of daily mood in Section 6.5.

[11] Note that this is a loose comparison because the datasets are different.

### 6.6.4 SVM Mood Curve and Sample Mood Curve

We are actually simulating real mood curves based on the results of sentiment analysis, but what if the real mood curves themselves have no power to predict sector index at the first place? In order to answer this question, we annotated another larger dataset (denoted as Sample). Tweets in Sample is randomly selected from Corpus during the fluctuating period (i.e., Oct. 8th–Nov. 9th) at the rate of 20% (i.e., 4106 tweets)[12]. Each tweet has been tagged by two independent annotators, and the agreement rate between the annotators is 88%. The supervisor (i.e., the author of this thesis) double-checked the left inconsistent tweets, and decided the final polarities of them.

Figure 14 shows the C2-based curve and the Sample-based curve. The vertical axis is WDM value. Figure 14 suggests that the two curves are correlated significantly ($p$-value of correlation analysis $< 0.01$), which means that C2 is reliable for building WDM time series. The prediction performance of Sample-based curve is shown in the next section.

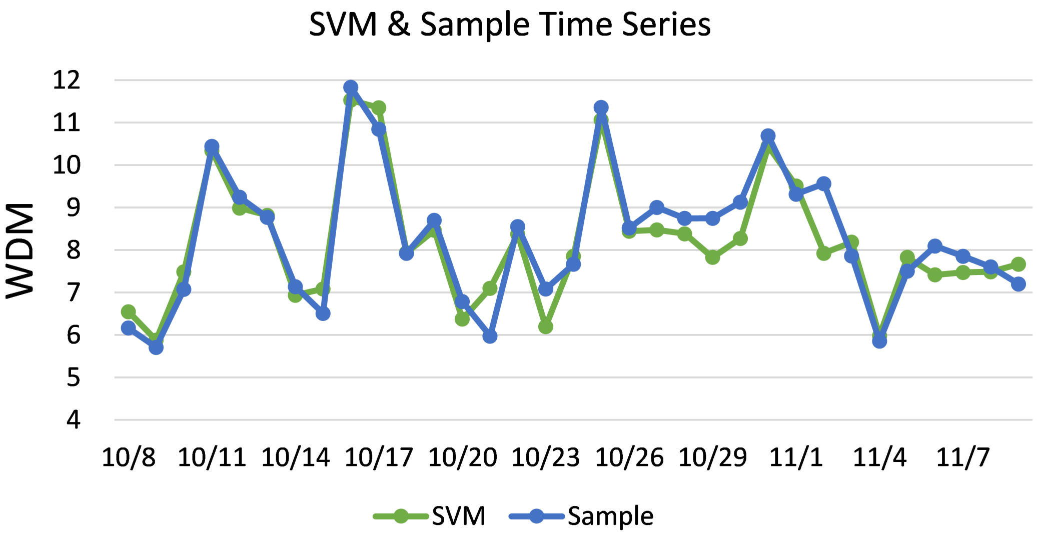


Figure 14: Comparison between SVM (C2) and Sample WDM Time Series (Oct. 8th–Nov. 9th)

[12] The best way to obtain real mood curves is to tag all the tweets in Corpus, but that is too many for manual annotation.

## 6.7 Experiments on Mood Time Series

Stock prediction is an extremely complex process. To better verify the prediction effect of the proposed mood time series, we evaluate it in two ways (i.e., CCF and VAR). CCF observes the static similarity between a mood time series and a stock index, while VAR assesses the dynamic one-day-ahead prediction ability of a mood time series. Moreover, we evaluate the WDM time series during fluctuating periods and monotonous periods, separately.

### 6.7.1 Public Mood Time Series

The C2 model is applied to predict the polarity for each tweet in Corpus. Since there is no similar work on tweet-level mood time series, we use Bollen's word-level method as our baseline (denoted as Raw).

Using WDM indicator, we can draw our mood time series[13]. For comparison, we also draw the DM time series. What's more, concerning that the original public mood is highly vibrant [54], we smooth the mood curves by moving average in a window of the past 7 days. Smoothed time series of Raw, DM, and WDM are shown in Figure 15 (z-score normalized).

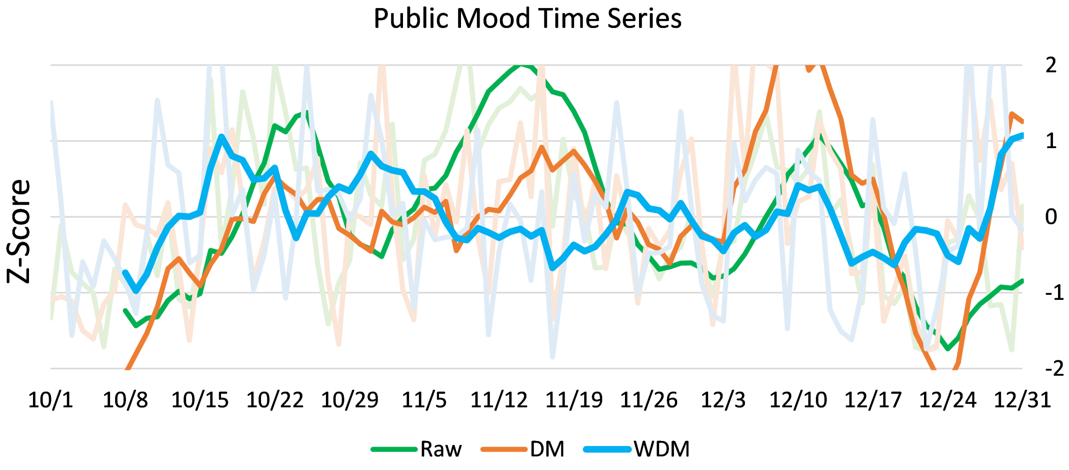


Figure 15: Public Mood Time Series (Oct. 8th–Dec. 31st)

[13] To compare with Sample-based WDM time series, the first 6 days of Index are cut off for smoothing.

### 6.7.2 Cross-Correlation Coefficient

Cross-correlation coefficient shifts one curve back and forth to estimate correlation between two series at different time lag [64]. We shift Index curve, so the right part where lag is greater than 0 means the ability to predict.

Figure 16 shows the correlation coefficients between mood curves[t] and Index [t + lag]. We can see that the WDM curve has the best similarity with Index in the predicting part of all kinds of period. The average correlation coefficient of WDM is 0.31 at the predicting part of the entire period. Moreover, WDM has a similar trend with Sample. What surprises us is that WDM is even higher than Sample. This may be because that Sample only contains 20% of Corpus, while WDM observes the whole Corpus. Furthermore, it is obvious that WDM works better than simple DM[14], which verifies our idea that non-text information helps. Lastly, we can see that WDM works much better in fluctuating periods than in monotonous periods. It achieves the best value when lag is 2 in the fluctuating period, whereas both DM and Raw have little predictive ability in this period.

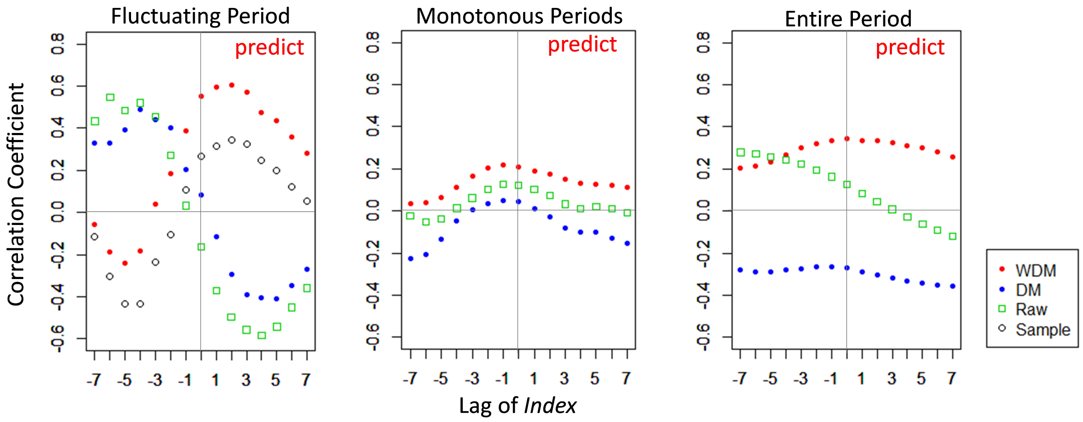


Figure 16: Correlation Coefficient for Different Lags in Different Periods

[14] For reference, [64] reported a 0.1 on average using DM.

### 6.7.3 Vector Auto Regression

To assess dynamic prediction ability, we use the VAR evaluation proposed in [71]. The first order (lag = 1) VAR model is defined as:

$$x_t = \theta_{11} x_{t-1} + \theta_{12} y_{t-1} + \epsilon_{x,t} \tag{6–5}$$

$$y_t = \theta_{21} x_{t-1} + \theta_{22} y_{t-1} + \epsilon_{y,t} \tag{6–6}$$

The training data are generated by sliding a window of the past $w$ days over mood curves and Index. The VAR uses the training data to predict the one-day-ahead up and down of Index. In our study, lag is in $\{1, 2, 3\}$ and $w$ is in $\{5, 10, 15\}$. Apart from mood curves, we also test Index itself by univariate VAR. All curves are normalized to $[0, 1]$.

Table 34 shows the average accuracy of the predictions in different lags. First, we can see that WDM performs best on average in the fluctuating period. It achieves the highest accuracy 72.9% at lag = 2, which is in accordance with the CCF results. Since the curve fluctuates greatly in this period, the average accuracy of Index itself is only 51.4%, which is nearly guess. Second, if we look at the monotonous periods, all the three mood curves are worse than the Index itself. This is because that the tendency in monotonous periods is very clear, Index itself can be a very strong predictor. In this kind of period, DM performs the best among the mood curves. Therefore, we combine a W&D curve using WDM in fluctuating periods and DM in monotonous periods for the entire period. W&D achieves an accuracy of 65.3% averagely, performing better than DM or WDM alone. Unfortunately, since the monotonous periods are nearly twice the length as the fluctuating period, the overall accuracy does not win Index.

## 6.8 Conclusion and Future Work

In this chapter, we presented a framework using public mood on social issues to predict sector index movement. We developed a low-dimensional customized sentiment classifier and designed a weighted daily mood indicator.

Table 34: Average Accuracies over All Training Windows Size and Different Lags in Different Periods

| Fluctuating Period | | | | | |
|---|---|---|---|---|---|
| Lag | Index | Raw | DM | WDM | Hand |
| 1 | 0.579 | 0.592 | 0.592 | 0.601 | 0.592 |
| 2 | 0.454 | 0.617 | 0.626 | 0.729 | 0.647 |
| 3 | 0.510 | 0.626 | 0.550 | 0.610 | 0.626 |
| avg | 0.514 | 0.612 | 0.589 | **0.647** | 0.622 |
| Monotonous Periods | | | | | |
| Lag | Index | Raw | DM | WDM | – |
| 1 | 0.755 | 0.735 | 0.769 | 0.683 | – |
| 2 | 0.757 | 0.688 | 0.717 | 0.738 | – |
| 3 | 0.797 | 0.695 | 0.683 | 0.667 | – |
| avg | 0.769 | 0.706 | **0.723** | 0.696 | – |
| Entire Period | | | | | |
| Lag | Index | Raw | DM | WDM | W&D |
| 1 | 0.694 | 0.653 | 0.658 | 0.636 | 0.673 |
| 2 | 0.677 | 0.668 | 0.659 | 0.683 | 0.678 |
| 3 | 0.685 | 0.600 | 0.634 | 0.591 | 0.606 |
| avg | 0.685 | 0.640 | 0.650 | 0.637 | **0.653** |

We found that non-text information of tweets was useful for both sentiment classification and daily mood design. Experiment results showed that our proposed method worked the best when evaluated by the static CCF. For predicting one-day-ahead up and down by dynamic VAR, mood curves performed better during fluctuating periods.

Although we only presented an experiment of the topic “food safety”, the described technique can be extended to any other topics. In the future, we plan to experiment with controversial topics, such as “genetically modified food” and “addictive online games”. In addition, since the prediction power depends on the period type, it’s meaningful to identify where the boundary of the period types lies. This will also be part of our future work.

# Chapter 7

# Conclusion and Future Work

In this paper, we constructed an annotated multilingual corpus for deeper sentiment understanding that encompassed three languages (English, Japanese, and Chinese) and four international topics (iPhone 6, Windows 8, Vladimir Putin, and Scottish Independence). We proposed a novel annotation scheme that embodied the idea of separating emotional signals and rhetorical context, which, in addition to global polarity, identifies rhetoric devices, emotional signals, degree modifiers, and subtopics. To address low inter-annotator agreement in previous corpora, we proposed a pivot dataset comparison method to effectively improve the agreement rate.

As discussed in Section 3.6.2, there is still much room for improvement in our annotation, we will continue to refine our corpus as part of our future work.

Based on observations and our analysis of our corpus, we find that 1) languages differ in terms of emotional signals and rhetoric devices, and the idea that cultures have different opinions regarding the same objects is reconfirmed; 2) each rhetoric device maintains its own characteristics, influences global polarity in its own way, and has an inherent structure that helps to model the sentiment that it represents; 3) the models of the expression of feelings in different languages are rather similar, suggesting the possibility of unifying multilingual opinion mining at the sentiment level.

We paid much attention to the agreement of global polarity in Section 3.5.3; given that the agreement of fine-grained components (i.e., emotional signals, degree modifiers, rhetorical context, and subtopics) involves so many situations (e.g., tag presence/absence, tag overlap, and tag category), we leave a detailed discussion about it for future work.

Besides, we proposed a new deep learning paradigm to assimilate language difference for MSA. We first pre-trained monolingual word embeddings separately, then mapped word embeddings in different spaces into a shared embedding space, and finally trained a parameter-sharing deep neural network for MSA. The experimental results showed that our paradigm was effective. Especially, our convolutional neural network model using transformed word embeddings outperforms a strong baseline by around 2.3% in term of classification accuracy.

The downsides of our work is that some components of our paradigm are still simple. In the future, we will try more complex transformation methods and network structures. Besides, pre-trained monolingual word embeddings can be further tuned using word-level polarities of words in the context that provided in the MDSU corpus. Finally, unsupervised text tokenizers, such as SentencePiece, may liberate us from using any language-specific tokenizers, which makes the proposed paradigm for MSA totally language-independent.

As an application, we applied monolingual sentiment analysis to unfolding public mood on social issues for sector index prediction. We first trained a low-dimensional SVM classifier using surrounding information for Twitter sentiment classification. Then, we generated public mood time series by aggregating tweet-level weighted daily mood (WDM) based on the sentiment classification results. Further, we evaluated our WDM series against the real stock index in two kinds of time period (i.e., fluctuating and monotonous period) by both static CCF and dynamic VAR. The experiments on "food safety" issue showed that the proposed WDM method outperformed the word-level baseline in predicting stock movement, especially during fluctuating periods.

Although we only presented an experiment of the topic "food safety", the described technique can be extended to any other social topics. In the future, we plan to experiment controversial topics, such as "genetically modified food" and "addictive online games". In addition, since the prediction power depends on period type, it's meaningful to judge where the boundary of the period types lies. These will also be part of our future work.

We hope that our work contributed some new knowledge to the research of social media sentiment analysis, especially its multilingual adaptability. We know that there are many issues left, such as how to make social media sentiment analysis rhetoric-tolerant. We will continue our work in the future.

# List of Published Papers

## Peer-Reviewed Paper in Journal

(1) Yujie Lu, Kotaro Sakamoto, Hideyuki Shibuki, and Tatsunori Mori. Construction of a Multilingual Annotated Corpus for Deeper Sentiment Understanding in Social Media. Journal of Natural Language Processing. 2017. 24(2), pages 205–266.

## Peer-Reviewed Paper in International Conference Proceedings

(1) Yujie Lu, Jinlong Guo, Kotaro Sakamoto, Hideyuki Shibuki, and Tatsunori Mori. Predicting Sector Index Movement with Microblogging Public Mood Time Series on Social Issues. In Proceedings of the 29th Pacific Asia Conference on Language, Information and Computation (PACLIC 29). 2015. pages 563–571.

## Non Peer-Reviewed Conference Papers

(1) Yujie Lu, Kotaro Sakamoto, Hideyuki Shibuki, and Tatsunori Mori. Are Deep Learning Methods Better for Twitter Sentiment Analysis? In Proceedings of The 23rd Annual Meeting of Natural Language Processing (Japan). 2017. pages 787–790.

(2) Yujie Lu, Kotaro Sakamoto, Hideyuki Shibuki, and Tatsunori Mori. Public Mood and the Collective Sentiment of Tweets. In Proceedings of The 23rd

Annual Meeting of Natural Language Processing (Japan). 2017. pages 651–654.

(3) Yujie Lu, Kotaro Sakamoto, Hideyuki Shibuki, and Tatsunori Mori. Construction of a Multilingual Annotated Corpus for Deep Sentiment Understanding in Social Media. Natural Language Processing Workshop 2015-NL-222 (Information Processing Society of Japan). 2015. pages 1–12.

(4) Yujie Lu, Kotaro Sakamoto, Hideyuki Shibuki, and Tatsunori Mori. A Method of Topic-Specific Multilingual Data Collecting and Annotation Preparation for Social Media. In Proceedings of The 21st Annual Meeting of Natural Language Processing (Japan). 2015. pages 557–560.

## Other Related Papers

(1) Jinlong Guo, Yujie Lu, Tatsunori Mori, and Catherine Blake. Expert-Guided Contrastive Opinion Summarization for Controversial Issues. In Proceedings of The 3rd International Workshop on Natural Language Processing for Social Media (SocialNLP 2015). 2015. pages 1105–1110.

(2) 阪本浩太郎, 陸宇傑, 福原優太, 渋木英潔, 石下円香, 森辰則, 神門典子. 2017. 大学入試世界史論述問題における非指定重要語句生成に関する検討. 言語処理学会第23回年次大会発表論文集. 2017. pages 166–169

# References

[1] Apoorv Agarwal, Boyi Xie, Ilia Vovsha, Owen Rambow, and Rebecca Passonneau. Sentiment analysis of twitter data. *Proceedings of the Workshop on Language in Social Media (LSM 2011)*, pages 30–38, 2011.

[2] Dongfang Wang ang Fang Li. Sentiment analysis of chinese microblogs based on layered features. In *Neural Information Processing*, pages 361–368. Springer International Publishing, 2014.

[3] Alexandra Balahur and Marco Turchi. Improving sentiment analysis in twitter using multilingual machine translated data. *In Proceedings of Recent Advances in Natural Language Processing*, pages 49–55, 2013.

[4] Carmen Banea, Rada Mihalcea, and Janyce Wiebe. Multilingual subjectivity: Are more languages better? *In Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010)*, pages 28–36, 2010.

[5] Roy Bar-Haim, Elad Dinur, Ronen Feldman, Moshe Fresko, and Guy Goldstein. Identifying and following expert investors in stock microblogs. *In Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing (EMNLP'11)*, pages 1310–1319, 2011.

[6] Valerio Basile and Malvina Nissim. Sentiment analysis on italian tweets. *Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 100–107, 2013.

[7] Yoshua Bengio, Rejean Ducharme, Pascal Vincent, and Christian Jauvin. A neural probabilistic language model. *Journal of Machine Learning Research*, 3:1137–1155, 2003.

[8] Shohini Bhattasali, Jeremy Cytryn, Elana Feldman, and Joonsuk Park. Automatic identification of rhetorical questions. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pages 743–749, 2015.

[9] Johan Bollen, Huina Mao, and Xiaojun Zeng. Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1):1–8, 2011.

[10] Pierre Luc Carrier and Kyunghyun Cho. Lstm networks for sentiment analysis. http://deeplearning.net/tutorial/lstm.html, 2017. [Online; accessed May 10, 2017].

[11] Chih-Chung Chang and Chih-Jen Lin. Libsvm: A library for support vector machine. *ACM Transactions on Intelligent Systems and Technology*, 2(3), 2011. Article No. 27.

[12] Marc Cheong and Vincent C. S. Lee. A microblogging-based approach to terrorism informatics: Exploration and chronicling civilian sentiment and response to terrorism events via twitter. *Information Systems Frontiers*, 13(1):45–59, 2011.

[13] Dmitry Davidov, Oren Tsur, and Ari Rappoport. Semi-supervised recognition of sarcastic sentences in twitter and amazon. *Proceedings of the Fourteenth Conference on Computational Natural Language Learning*, pages 107–116, 2010.

[14] Kerstin Denecke. Using sentiwordnet for multilingual sentiment analysis. *In Proceedings of the 24th International Conference on Data Engineering Workshop (ICDE 2008)*, pages 507–512, 2008.

[15] Cícero Nogueira dos Santos. Think positive: Towards twitter sentiment analysis from scratch. *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 647–651, 2014.

[16] Cícero Nogueira dos Santos and Maíra Gatti. Deep convolutional neural networks for sentiment analysis of short texts. *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 69–78, 2014.

[17] Jeffrey L Elman. Finding structure in time. *Cognitive Science*, 14(2):179–211, 1990.

[18] Murthy Ganapathibhotla and Bing Liu. Mining opinions in comparative sentences. *Proceedings of the 22nd International Conference on Computational Linguistics (COLING 2008)*, pages 241–248, 2008.

[19] Aniruddha Ghosh, Tony Veale, Ekaterina Shutova, John Barnden, Guofu Li, Paolo Rosso, and Antonio Reyes. Semeval-2015 task 11: Sentiment analysis of figurative language in twitter. *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 470–478, 2015.

[20] Alec Go, Richa Bhayani, and Lei Huang. Twitter sentiment classification using distant supervision. *CS224N Project Report (Stanford)*, pages 1–6, 2009.

[21] Roberto González-Ibáñez, Smaranda Muresan, and Nina Wacholder. Identifying sarcasm in twitter: A closer look. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: short papers (ACL'11)*, pages 581–586, 2011.

[22] Masatsugu Hangyo, Daisuke Kawahara, and Sadao Kurohashi. Building and analyzing a diverse document leads corpus annotated with semantic relations. *Journal of Natural Language Processing*, 21(2):213–247, 2014.

[23] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv*, pages 1–18, 2012.

[24] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computatio*, 9(8):1735–1780, 1997.

[25] Eduard H. Hovy. What are sentiment, affect, and emotion? applying the methodology of michael zock to sentiment analysis. In *Language Production, Cognition, and the Lexicon*, pages 13–24. Springer International Publishing, 2015.

[26] Bernard J. Jansen, Mimi Zhang, Kate Sobel, and Abdur Chowdury. Twitter power: Tweets as electronic word of mouth. *Journal of the American Society for Information Science and Technology*, 60(11):2169–2188, 2009.

[27] Fei Jiang, Yiqun Liu, Huanbo Luan, Min Zhang, and Shaoping Ma. Microblog sentiment analysis with emoticon space model. In *Neural Information Processing*, pages 76–87. Springer International Publishing, 2014.

[28] Vineet John. A survey of neural network techniques for feature extraction from text. *arXiv*, pages 1–12, 2017.

[29] Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. A convolutional neural network for modelling sentences. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 655–665, 2014.

[30] Yoon Kim. Convolutional neural networks for sentence classification. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, 2014.

[31] Svetlana Kiritchenko and Saif M. Mohammad. Sentiment composition of words with opposing polarities. *In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2016)*, pages 1102–1108, 2016.

[32] Efthymios Kouloumpis, Theresa Wilson, and Johanna Moore. Twitter sentiment analysis: The good the bad and the omg! *Proceedings of the Fifth International Conference on Weblogs and Social Media (ICWSM'11)*, pages 538–541, 2011.

[33] Zornitsa Kozareva. Multilingual affect polarity and valence prediction in

metaphor-rich texts. *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL'13)*, pages 682–691, 2013.

[34] George Lakoff and Mark Johnson. *Metaphors We Live By.* University of Chicago Press, 1980.

[35] Bing Liu. *Sentiment Analysis and Opinion Mining.* Morgan & Claypool Publishers, 2012.

[36] Bing Liu, Minqing Hu, and Junsheng Cheng. Opinion observer: Analyzing and comparing opinions on the web. *Proceedings of the 14th international conference on World Wide Web (WWW '05)*, pages 342–351, 2005.

[37] Yujie Lu, Jinlong Guo, Kotaro Sakamoto, Hideyuki Shibuki, and Tatsunori Mori. Predicting sector index movement with microblogging public mood time series on social issues. *Proceeding of 29th Pacific Asia Conference on Language, Information and Computation (PACLIC 2015)*, pages 563–571, 2015.

[38] Yujie Lu, Kotaro Sakamoto, Hideyuki Shibuki, and Tatsunori Mori. Construction of a multilingual annotated corpus for deep sentiment understanding in social media. *IPSJ SIG Technical Reports*, 2015-NL-222(1):1–12, 2015.

[39] Yujie Lu, Kotaro Sakamoto, Hideyuki Shibuki, and Tatsunori Mori. Are deep learning methods better for twitter sentiment analysis? *In Proceedings of The 23rd Annual Meeting of Natural Language Processing (Japan)*, pages 787–790, 2017.

[40] Yujie Lu, Kotaro Sakamoto, Hideyuki Shibuki, and Tatsunori Mori. Construction of a multilingual annotated corpus for deeper sentiment understanding in social media. *Journal of Natural Language Processing*, 24(2):205–266, 2017.

[41] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval Chapter Scoring, Term Weighting and the Vector Space Model.* Cambridge University Press, 2008.

[42] James H. Martin. Representing regularities in the metaphoric lexicon. *Proceedings of the 12th conference on Computational Linguistics (COLING'88)*, pages 396–401, 1988.

[43] Rada Mihalcea, Carmen Banea, and Janyce Wiebe. Learning multilingual subjective language via cross-lingual projections. *In Proceedings of the 45th Annual Meeting of the As-sociation of Computational Linguistics (ACL 2007)*, pages 976–983, 2007.

[44] Tomas Mikolov, Martin Karafiat, Lukas Burget, JanCernocky, and Sanjeev Khudanpur. Recurrent neural network based language model. *INTERSPEECH*, pages 1045–1048, 2010.

[45] Tomas Mikolov, Quoc V. Le, and Ilya Sutskever. Exploiting similarities among languages for machine translation. *arXiv:1309.4168*, 2013.

[46] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrasesand their compositionality. *In Proceedings of NIPS 2013*, pages 1–9, 2013.

[47] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv*, pages 1–12, 2013.

[48] Rintaro Miyazaki and Tatsunori Mori. Creation of sentiment corpus by multiple annotators with an annotation tool that has a function of referring example annotations. *Journal of Natural Language Processing*, 17(5):3–50, 2010.

[49] Karo Moilanen and Stephen Pulman. Sentiment composition. *Computational Linguistics Group Course Report(Oxford)*, 2007.

[50] Subhabrata Mukherjee and Pushpak Bhattacharyya. Sentiment analysis in twitter with lightweight discourse analysis. *Proceedings of the 23th International Conference on Computational Linguistics: Technical Papers (COLING 2012)*, pages 1847–1864, 2012.

[51] Preslav Nakov, Sara Rosenthal, Zornitsa Kozareva, Veselin Stoyanov, Alan Ritter, and Theresa Wilson. Semeval-2013 task 2: Sentiment analysis in twitter. *Proceedings of the 2nd Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 312–320, 2013.

[52] Graham Neubig and Kevin Duh. How much is said in a tweet? a multilingual, information-theoretic perspective. *AAAI Spring Symposium*, pages 32–39, 2013.

[53] John R. Nofsinger. Twitter mood predicts the stock market. *Journal of Behavioral Finance*, 6(3):144160, 2005.

[54] Brendan O'Connor, Ramnath Balasubramanyan, Bryan R. Routledge, and Noah A. Smith. From tweets to polls: Linking text sentiment to public opinion time series. *Proceedings of the International AAAI Conference on Weblogs and Social Media (ICWSM'10)*, pages 122–129, 2010.

[55] Alexander Pak and Patrick Paroubek. Twitter as a corpus for sentiment analysis and opinion mining. *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, pages 1320–1326, 2010.

[56] Bo Pang and Lillian Lee. *Opinion mining and sentiment analysis.* Now Publishers, 2008.

[57] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up? sentiment classification using machine learning techniques. *Proceedings of Empirical Methods on Natural Language Processing (EMNLP 2002)*, pages 79–86, 2002.

[58] Jonathon Read. Using emoticons to reduce dependency in machine learning techniques for sentiment classification. *Proceedings of the ACL Student Research Workshop*, pages 43–48, 2005.

[59] Antonio Reyes, Paolo Rosso, and Davide Buscaldi. From humor recognition to irony detection: The figurative language of social media. *Data and Knowledge*

*Engineering*, 74:1–12, 2012.

[60] Antonio Reyes, Paolo Rosso, and Tony Veal. A multidimensional approach for detecting irony in twitter. *Language Resources and Evaluation*, 47(1):239–268, 2013.

[61] Sara Rosenthal, Alan Ritter, Preslav Nakov, and Veselin Stoyanov. Semeval-2014 task 9: Sentiment analysis in twitter. *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 73–80, 2014.

[62] Sebastian Ruder. A survey of cross-lingual embedding models. *arXiv:1706.04902*, 2017.

[63] Sebastian Ruder, Parsa Ghaffari, and John G. Breslin. Insight-1 at semeval-2016 task 5: Deep learning for multilingual aspect-based sentiment analysis. *In Proceedings of Proceedings of SemEval-2016*, pages 330–336, 2016.

[64] Eduardo J. Ruiz, Vagelis Hristidis, Carlos Castillo, Aris tides Gionis, and Alejandro Jaimes. Correlating financial time series with micro-blogging activity. *In Proceedings of the fifth ACM international confer-ence on Web search and data mining (WSDM' 12)*, pages 513–522, 2012.

[65] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533536, 1986.

[66] Hassan Saif, Miriam Fernandez, Yulan He, and Harith Alani. Evaluation datasets for twitter sentiment analysis: A survey and a new dataset, the sts-gold. *Proceedings of the 1st International Workshop on Emotion and Sentiment in Social and Expressive Media (ESSEM 2013)*, 2013.

[67] Hassan Saif, Yulan He, and Harith Alani. Alleviating data sparsity for twitter sentiment analysis. *Proceedings of the 2nd Workshop on Making Sense of Microposts*, pages 2–9, 2012.

[68] Erik Tjong Kim Sang and Johan Bos. Predicting the 2011 dutch senate election results with twitter. *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 53–60, 2012.

[69] Aliaksei Severyn and Alessandro Moschitti. Twitter sentiment analysis with deep convolutional neural networks. *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'15)*, pages 959–962, 2015.

[70] Imran Sheikh, Irina Illina, Dominique Fohr, and Georges Linarés. Learning word importance with the neural bag-of-words model. *Proceedings of the 1st Workshop on Representation Learning for NLP*, pages 222–229, 2016.

[71] Jianfeng Si, Arjun Mukherjee, Bing Liu, Qing Li, Huayi Li, and Xiaotie Deng. Exploiting topic based twitter sentiment for stock prediction. *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL'13)*, pages 24–29, 2013.

[72] Wladimir Sidorenko, Jonathan Sonntag, Manfred Stede, Nina Krüger, and Stefan Stieglitz. From newspaper to microblogging: What does it take to find opinions? *Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 81–86, 2013.

[73] Richard Socher, Alex Perelygin, Jean Y. Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1631–1642, 2013.

[74] Josef Steinberger, Polina Lenkova, Mijail Kabadjov, Ralf Steinberger, and Erik van der Goot. Multilingual entity-centered sentiment analysis evaluated by parallel corpora. *Proceedings of Recent Advances in Natural Language Processing*, pages 770–775, 2011.

[75] Duyu Tang, Furu Wei, Bing Qin, Ting Liu, and Ming Zhou. Coooolll: A deep learning system for twitter sentiment classification. *Proceedings of the*

*8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 208–212, 2014.

[76] Huifeng Tang, Songbo Tan, and Xueqi Cheng. A survey on sentiment detection of reviews. *Expert Systems with Applications*, 36(7):10760–10773, 2009.

[77] Yi-Jie Tang and Hsin-Hsi Chen. Chinese irony corpus construction and ironic structure analysis. *Proceedings of the 25th International Conference on Computational Linguistics: Technical Papers (COLING 2014)*, pages 81–86, 2014.

[78] TensorFlow. Vector representations of words. https://www.tensorflow.org/tutorials/word2vec, 2017. [Online; accessed May 10, 2017].

[79] Andranik Tumasjan, Timm O. Sprenger, Philipp G. Sandner, and Isabell M. Welpe. Election forecasts with twitter: How 140 characters reflect the political landscape. *Social Science Computer Review*, 29(4):402–418, 2011.

[80] Piyoros Tungthamthiti, Enrico Santus, Hongzhi Xu, Chu-Ren Huang, and Kiyoaki Shirai. Sentiment analyzer with rich features for ironic and sarcastic tweets. *Proceedings of the 29th Pacific Asia Conference on Language, Information and Computation (PACLIC 2015)*, pages 178–187, 2015.

[81] Peter D. Turney. Thumbs up or thumbs down?: Semantic orientation applied to unsupervised classification of reviews. *ACL '02 Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 417–424, 2002.

[82] Andrea Vanzo, Danilo Croce, and Roberto Basili. A context-based model for sentiment analysis in twitter. *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers)*, pages 2345–2354, 2014.

[83] Julio Villena-Román, Janine García-Morera, Sara Lana-Serrano, and José Carlos González-Cristóba. Tass 2013 - a second step in reputation analysis in spanish. *Procesamiento del Lenguaje Natural*, 52:37–44, 2014.

[84] Svitlana Volkova, Theresa Wilson, and David Yarowsky. Exploring demographic language variations to improve multilingual sentiment analysis in social media. *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP 2013)*, pages 1815–1827, 2013.

[85] Tien Thanh Vu, Shu Chang, Quang Thuy Ha, and Ni gel Collier. An experiment in integrating sentiment features for tech stock prediction in twitter. *In Proceedings of the Workshop on Information Ex-traction and Entity Analytics on Social Media Data (COLING'12)*, pages 23–38, 2012.

[86] Xiaojun Wan. Co-training for cross-lingual sentiment classification. *In Proceedings of the Joint Conference of the 47th An-nual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 235–243, 2009.

[87] Haohan Wang and Bhiksha Raj. On the origin of deep learning. *arXiv:1702.07800v4*, 2017.

[88] Xin Wang, Yuanchao Liu, Chengjie Sun, Baoxun Wang, and Xiaolong Wang. Predicting polarities of tweets by composing word embeddings with long short-term memory. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pages 1343–1353, 2015.

[89] Janyce Wiebe, Theresa Wilson, and Claire Cardie. Annotating expressions of opinions and emotions in language. *Language Resources and Evaluation*, 39(2-3):165–210, 2005.

[90] Michael Wiegand, Alexandra Balahur, Benjamin Roth, Dietrich Klakow, and Andres Montoyo. A survey on the role of negation in sentiment analysis. *Proceedings of the Workshop on Negation and Speculation in Natural Language Processing*, pages 60–68, 2010.

[91] Bing Xiang and Liang Zhou. Improving twitter sentiment analysis with topic-based mixture modeling and semi-supervised training. *Proceedings of the*

*52nd Annual Meeting of the Association for Computational Linguistics (Short Papers)(ACL'14)*, pages 434–439, 2014.

[92] Lixing Xie, Ming Zhou, and Maosong Sun. Hierarchical structure based hybrid approach to sentiment analysis of chinese micro blog and its feature extraction. *Journal of Chinese Information Processing*, 26(1):73–83, 2012.

[93] Wenpeng Yin, Katharina Kann, Mo Yu, and Hinrich Schütze. Comparative study of cnn and rnn for natural language processing. *arXiv:1702.01923*, 2017.

[94] Ye Yuan and You Zhou. Twitter sentiment analysis with recursive neural networks. *CS224d Project Report (Stanford)*, pages 1–8, 2015.

# Appendix A

# Detailed Statistics of Data Selection

Data collection spanned one year, but data selection was required before we made the data available to the annotators. Hence, the original dataset for data selection was only the first part of the collected tweets described in Table 2. However, we did not record data regarding the actual data selection process during the first run; thus, we obtained the data presented in Table 35 from a reimplementation of data selection after annotation was completed; as such, there may be minor differences between what we present here and the first run.

Here, we describe the meaning of the terms in Table 35. Original is the number of tweets for selection. RT is the number of tweets after the exclusion of retweets. Veto is the number of tweets after the exclusion of tweets containing any other pattern in Table 3. Length is the number of tweets selected based on the length threshold (i.e., byte count). Count is the number of tweets selected based on the count threshold. Final is the number of tweets selected from the Top k tweets (if the size of the remaining set was large, not all of them needed to be checked during manual selection). Pct. (%) indicates the selection percentage of each stage.

Example tweets that were removed during manual selection are listed below; note that this stage was primarily applied to Japanese and Chinese tweets.

- TheTIMESALE(ザ・タイムセール) は、お店の余剰商品・時間をムダなく有効活用できるサービスです。カフェやレストラン、サロンや癒し店を経営の

Table 35: Detail Statistics for Data Selection

| **English** | **Stage/Step** | **I6** | **Pct.(%)** | **W8** | **Pct.(%)** | **PU** | **Pct.(%)** | **SI** | **Pct.(%)** |
|---|---|---|---|---|---|---|---|---|---|
| | Original | 693396 | 100.0 | 31164 | 100.0 | 137836 | 100.0 | 29016 | 100.0 |
| Exclusive Filtering | RT | 580945 | 3.7 | 28906 | 4.7 | 53241 | 9.1 | 11142 | 12.6 |
| | Veto | 25692 | | 1467 | | 12573 | | 3664 | |
| Inclusive Filtering | Length | 9558 | 2.4 | 801 | 54.6 | 9557 | 17.9 | 2692 | 37.3 |
| | (threshold) | (>=100) | | (>=100) | | (>=100) | | (>=100) | |
| | Count | 608 | | 801 | | 2254 | | 1367 | |
| | (threshold) | (<3) | | (none) | | (<3) | | (<2) | |
| Manual Selection | Top k | 446 | 96.6 | 489 | 93.0 | 459 | 98.0 | 452 | 99.6 |
| | Final | 431 | | 455 | | 450 | | 450 | |
| **Japanese** | **Stage/Step** | **I6** | **Pct.(%)** | **W8** | **Pct.(%)** | **PU** | **Pct.(%)** | **SI** | **Pct.(%)** |
| | Original | 312027 | 100.0 | 12650 | 100.0 | 285166 | 100.0 | 7430 | 100.0 |
| Exclusive Filtering | RT | 288983 | 0.2 | 11834 | 0.7 | 135931 | 7.8 | 3709 | 9.7 |
| | Veto | 774 | | 90 | | 22327 | | 721 | |
| Inclusive Filtering | Length | 705 | 91.1 | 87 | 96.7 | 11968 | 39.8 | 618 | 85.7 |
| | (threshold) | (>=60) | | (>=60) | | (>=100) | | (>=100) | |
| | Count | 705 | | 87 | | 8878 | | 618 | |
| | (threshold) | (none) | | (none) | | (<1) | | (none) | |
| Manual Selection | Top k | 705 | 42.0 | 64 | 73.4 | 1283 | 37.3 | 586 | 65.9 |
| | Final | 296 | | 47 | | 478 | | 386 | |
| **Chinese** | **Stage/Step** | **I6** | **Pct.(%)** | **W8** | **Pct.(%)** | **PU** | **Pct.(%)** | **SI** | **Pct.(%)** |
| | Original | 28278 | 100.0 | 6661 | 100.0 | 42856 | 100.0 | 6262 | 100.0 |
| Exclusive Filtering | Veto | 16852 | 59.6 | 3144 | 47.2 | 10187 | 23.8 | 1943 | 31.0 |
| Inclusive Filtering | Length | 8264 | 22.6 | 2456 | 70.4 | 4992 | 38.6 | 1801 | 83.9 |
| | (threshold) | (>150) | | (>100) | | (>150) | | (>60) | |
| | Count | 3814 | | 2212 | | 3933 | | 1631 | |
| | (threshold) | (<=1) | | (<=3) | | (<=2) | | (<=4) | |
| Manual Selection | Top k | 2010 | 23.4 | 1209 | 37.6 | 1489 | 30.2 | 1631 | 28.8 |
| | Final | 470 | | 455 | | 450 | | 450 | |

みなさま、ぜひご活用ください！ #アプリ #サービス #iPhone6 #android #集客 #飲食店 #マッサージ #サロン

[Ref: TheTIMESALE is a service to make full use of the redundant goods and time of your shops. People who are running cafes, restaurants, salons and healing shops, please use this freely. #app #service #iPhone6 #android #attractingcustomers #restaurant #massage #salon]

Type (2): This is a commercial tweet containing no opinion regarding iPhone 6.

- 不会**吧**，**这**才几天**啊**，**这帮**明星都用上 iphone6 了，**唉**，**这**世界，富的真富，而那些乞丐……明星就不**应该**有**这么**多的工**资**，他**们**有什**么贡**献**啊**，**对**世界**对**人民？

  [Ref: No way, it's just been a few days. These stars are all using iPhone 6. Alas, what a world. The rich become richer, whereas the beggars... Those stars shouldn't get paid so much, what have they contributed to the world and the people?]

  Type (3): This is a tweet ridiculing a social phenomenon instead of evaluating iPhone 6.

- イリーナ・ウラジーミロヴナ・プチナ（プーチン）「私をあまり怒らせない方がいいわよ……？」（幼なじみは大統領）

  [Ref: Irina Vladimirovna Putina (Putin) "you'd better not make me angry…?" (My Girlfriend Is the President)]

  Type (3): Note that this is a tweet that should have been removed, since it does not discuss the Russian President; however, it was neglected because the supervisor was not familiar with the given novel at the moment.

# Appendix B

# Example of Annotation Result in XML

Figure 17 is an example of annotation result in XML from the gold standard. Note that in this example, the sentence polarity of the sarcasm frame is tagged on the full stop of the locating sentence in that we did not perform sentence segmentation to reduce the burden on annotators.

```
<Tweet Version="2.0" Language="English" Topic="I6" TID="209"
Polarity="Negative" Rhetoric="Comparison(s); Sarcasm(s)"
Subtopics="voice texting; rotary dial">
  <Span SID="1" Category="positive">Wow</Span>, with #iPhone6, you
  <Span SID="2" Category="positive">can</Span> send a message <Span
  SID="3" Category="intensifier">just</Span> by talking!  In <Span
  SID="4" Category="intensifier">any</Span> voice you <Span SID="5"
  Category="positive">like</Span>.  So can my mom’s <Span SID="6"
  Category="negative">old</Span> <Span SID="7" Category="neutral"
  RhetoricType="Comparison(s)">rotary dial</Span><Span SID="8"
  Category="negative" RhetoricType="Sarcasm(s)">.</Span>
</Tweet>
```

Figure 17: Example of Annotation Results in XML

# Appendix C

# Detail of Coding Manual

The following coding manual is included here to explain why and how to perform the annotation work to the annotators. As a guidebook, it is intended to standardize the procedure of annotation using the annotation support tool shown in Figure 2. Note that the special cases discussed in Section 7.2 were discovered while performing the analysis; thus, their corresponding annotation methods are not yet included in this manual.

**Background & Purpose**

"What other people think" has always been an important piece of information for most of us during the daily decision-making process. The emerging of social network, such as Twitter and Facebook, has now provided people enough material. Collecting opinions and mining out sentiment distribution or tendency from those social media automatically is an indispensable research field, which is generally called Sentiment Analysis. In this project, we will expand such analysis to a multilingual setting, including English, Japanese and Chinese.

To have a good understanding of natural language texts and to construct a gold-standard corpus for system evaluation, manual annotation is carried out by most of the researchers. For these reasons, we are taking effort for building up such a multilingual tweet corpus.

**Points of Attention**

(1) Annotation is a process that needs carefulness. Please be concentrated and patient while working.

(2) Please work at a proper pace, and start/finish a working section (2 or 3 h) on time.

(3) Download the latest tool and datasets before you start a new working section, and upload your result files when you finish.

(4) Some of your annotation results will be checked afterward, and you may get feedback before the next working section.

**Task Description**

You need to annotate the following items of the given tweets regarding an evaluation object. **Attention: All the judgment and annotation are done for the evaluation objects. Please don't lose the focus.*

**(1) Tag the words/phrases containing emotion and their degree modifiers**

1 Look from the start to the end of a tweet word by word (do not skip any word).

2 Recognize and tag the words containing positive, negative or neutral emotions.

3 The positive/negative words are especially important. After you tag them, they will also be displayed at the right of the tweet text editor. A positive/negative word can be any part of speech.

4 Recognize and tag the degree words (including intensifier, diminisher and negation) that modify those emotional words.

5 Intensifier, diminisher, and negation are also good indicators to distinguish the boundaries of emotional words.

6 Note that the emotions of positive/negative/neutral words are their emotions in general use (i.e., in a dictionary) or in social-network use

(e.g., net slang). Particularly, non-emotional words (i.e., words containing no emotion in a dictionary) can become emotional according to the contexts. For example, the word “joined” in the following tweet can be tagged as “positive”, because “join” expresses a preference to topic iPhone 6.

☑ Just joined**(positive)** the #iphone6 family. Gonna miss my HTC One I think but the camera is gonna be way better with iPhone. Time to take some photos!

7 When should we annotate an emotional word?

- We only annotate the words when they have an influence on global polarity (called as “signal”). This means that the annotated words should be the clues or hints to global polarity (i.e., having a relation with global polarity). We attach importance to emotional signals rather than pure words.
- If a word in a polarity lexicon is only a statement or narrative that has nothing to do with its evaluation object, there is no need to tag it. For example, “want” in the following tweet should not be tagged, since it has no influence on global polarity.

  ☒ Biggest complaint with my #iPhone6 is the gyroscope. Constantly shaking & rotating to recognize I want**(positive)** my phone sideways. Text app specific

- This judgment sometimes depends heavily on the your interpretation of the context.

8 Emotional words and their degree modifiers often appear in pairs (sometimes they are away from each other). The emotional words are at the center of the pair. For example, “very well” in the following tweet is such a pair.

☑ The #iphone6 is a very**(intensifier)** well**(positive)** made phone..beautiful interface, fast processing sleek design, iv enjoyed the experience

- Please avoid tagging degree modifiers alone in a sentence as follows.

  ☒ I'm so not**(negation)** used to this big ass phone but finally I upgraded & most importantly I got a new cellular device #iPhone6

9 If there is a phrase or an expression containing an emotion, such as "brand new" and "get rid of," tag all its words together. If the phrase/expression is separated by other words, you can group them as follows.

☑ I've been using #Windows8 and 8.1 since May 2014 and have had nothing but trouble. #Microsoft were insane to force this upon**(negative)** their users.

10 Note that the smallest unit to tag is word for English and character for Japanese and Chinese. When you tag a word, it becomes a signal. The attributes of signals are shown over the tweet text editor.

11 Note that each word only belongs to one category. For most cases, this rule works well. If there is a violation, please make a choice by the content (i.e., meaning, not the shape) of the word.

12 There is no need to tag the evaluation objects (they are already in bold).

13 Net slang and smileys need to be tagged. If you are not sure about their meaning, please google them.

14 There is no need to tag the full stops of the sentences unless they are sarcastic sentences or rhetorical questions (see more on this in (2)).

15 Some other language-specific points:

- Chinese

  ◇ Texts like [心] are substitutes for graphic emoticons. Please regard them as words.

  ◇ Pay attention to the segmentation of some words, such as "不**满**意 → 不 + **满**意" and "不好 → 不 + 好."

- Japanese

◇ Tag verbs with their conjugation (変格活用), such as [なくなりました][なくなった]。

◇ Tag *suru* verbs (サ変動詞) as a whole, such as [肥大する].

**(2) Choose rhetoric devices in the tweets (if there is any)**

Rhetoric devices that tweets contain can cause difficulties for the proposed systems to classify the global polarities of tweets. This is what researchers are paying attention to. In linguistics, rhetoric has strict definitions and many genres, while in our task, we only focus on highly frequent rhetoric devices, including metaphor, comparison, sarcasm, and rhetorical question.

Please choose one or more rhetoric devices, if any exists. The supplementary information related to the selected rhetoric devices also needs to be tagged.

This is a multiple-choice question, whit five choices:

- Non-rhetoric
  Straightforward tweet. No rhetoric in the tweet.

If the tweet contains metaphor or comparison, please supplement the emotional information of their counterparts.

- Metaphor

☑ As a #Mac user, #Windows8 is figuratively **[the bane of my existence] (Metaphorically negative)**. Trying to do anything is nigh on impossible.

- Comparison

☑ Can now definitively say that #Windows8 IS indeed faster and more stable than **[#Windows7](Comparatively inferior)** used both for a while now. Don't be afraid of 8 #fb

If the tweet contains sarcasm or rhetorical question, please supplement the sentence polarity at the full stops of the locating sentences (there is no full stops, a "$" symbol can be added).

- Sarcasm

☑ [Every time I use #Windows8, I become more impressed with how profoundly bad a UX it is.]**(Sarcastically negative)**[Its an almost perfect #antidesign]**(Sarcastically negative)**

- Rhetorical Question

☑ last #windows8 update took more time than loading 20 #c64 games with #datasette ...[what went wrong in 30 years?]**(Rhetorically negative)**

After you choose the rhetoric devices, please supplement the necessary information for each chosen rhetoric device.

If a word has already been tagged as subtopic in the previous task, you need to change the previous tag to a rhetoric element, because rhetoric takes priority at any time.

### (3) Determine the Global Polarity of Tweets

Global polarity is a vital information of a tweet, which is important for evaluating proposed systems. Please judge whether the author of the tweet is for (supportive) or against (non-supportive) the evaluation object, and select the global polarity toward the evaluation object.

Before you make the judgment, remember to clear up the old impression of the evaluation object in your brain first. The answer should be decided based on the tweet text and should not be affected by the annotator's own preference.

This is a single choice, and there are the four choices.

- Positive (supportive)
- Negative (non-supportive)
- Neutral

    - ◇ Mixed or undecided tweets
    - ◇ Non-comment tweets
    - ◇ Objective tweets, such as news, commercial
    - ◇ Irrelevant tweets

- None (unreadable)

  Choose this option only if the tweet is unable to be understood for reasons such as encoding problem, dialects.

**(4) Write Down the Subtopics of the Tweets**

Subtopic information is not ignorable in order to observe the components of people's opinions. For each tweet, there is an evaluation object, which is also the main topic. Regarding the main topic, there could be many subtopics that people are talking about in tweets, such as aspects and attributes.

You need to write down a couple of subtopics (at least 1) for each tweet. The form of subtopic can be word or phrase. Subtopics should be noun or gerund.

- Please recognize and tag the subtopics from the tweet text (they are automatically added to the subtopic text editor, and their positions in the tweet are recorded). Pay close attention to nouns in the tweet.
- If there is no any apparent subtopic text in the tweet, please summarize from the tweet. You can freely edit the picked-up subtopics in the subtopic text editor into good shape.

### Data Distribution

(1) Each annotator only takes charge of one language (which is your native language).

(2) Each annotator takes charge of two evaluation objects (one of the following combinations).

- iPhone 6 (product) + Putin (figure)
- Windows 8 (product) + Scottish Independence (event)

(3) The tweets for annotation will be distributed in order. A new set of tweets will be distributed after you have finished the previous one.

# Appendix D

# Lexicons Used for WPN Computation

The lexicons we used for computing WPN are as follows:

- Liu Bing's English Opinion Lexicon (2006 positive/4783 negative words)
  URL: http://www.cs.uic.edu/∼liub/FBS/opinion-lexicon-English.rar

- Chinese Emotion Ontology Lexicon (11229 positive/10783 negative words)
  URL: http://ir.dlut.edu.cn/EmotionOntologyDownload

- Japanese Sentiment Polarity Lexicon (5462 positive/ 8129 negative words)
  URL: http://www.cl.ecei.tohoku.ac.jp/index.php?Open%20Resources%2FJapanese%20Sentiment%20Polarity%20Dictionary

- SentiStrength Emoticon Lookup Table (46 positive /58 negative emoticons)
  URL: http://Sentistrength.wlv.ac.uk/SentStrength_Data_Sept2011.zip