The Sentimental Value of Chinese Sub-Character Components

Yassine Benajiba **Or Biran** **Zhiliang Weng** **Yong Zhang** **Jin Sun**

Mainiway AI Lab

{yassine,or.biran,zhiliang.weng,yong.zhang,jin.sun}@mainiway.com

Abstract

Sub-character components of Chinese characters carry important semantic information, and recent studies have shown that utilizing this information can improve performance on core semantic tasks. In this paper, we hypothesize that in addition to semantic information, sub-character components may also carry emotional information, and that utilizing it should improve performance on sentiment analysis tasks. We conduct a series of experiments on four Chinese sentiment data sets and show that we can significantly improve the performance in various tasks over that of a character-level embeddings baseline. We then focus on qualitatively assessing multiple examples and trying to explain how the sub-character components affect the results in each case.

1 Introduction

Chinese characters are composed of one or more *components*, which may have a phonetic or semantic meaning. A special type of component is a *radical*, which is the component under which a character is traditionally listed in the dictionary. Radicals, in particular, often carry a semantic meaning. For example, the character 媽 (*mā*, "mother") is composed of the semantic component, which is also the radical, 女 (*nǔ*, "female") and the phonetic component 馬 (*mǎ*, "horse").

Recently, there has been growing focus on utilizing sub-character components, such as radicals, in natural language processing. These components can carry intrinsic semantic information that complements the contextual information that is utilized, e.g., in building word embeddings. It has been shown that embeddings which are constructed with a combination of radical, character and word level granulairty outperform those that lack the radical information on classical semantic tasks such as analogy and paraphrasing (Sun et al., 2014; Li et al., 2015; Yin et al., 2016; Yu et al., 2017).

In this paper, we explore the hypothesis that in addition to the sort of hard semantic tasks that they have so far been applied to, sub-character components can also carry *sentiment*-related or *emotional* information, and therefore should be useful in sentiment analysis as well. In particular, we have in mind three types of sentiment-related information in semantic components:

1. Components that have a specific polarity, such as 疒 ("disease") which is generally found in negative characters, or 子 ("child") which is somewhat more common in positive characters

2. Components that do not specify a polarity, but specify subjectivity or emotional content, such as 心 ("heart") or 忄 ("heart" in vertical form)

3. Components that are objective, but because of human tendencies are more likely to appear in characters that tend to appear in subjective context and may tend towards a particular polarity or intensity, such as 虫 ("insect") or 贝 ("treasure")

To test our hypothesis, we conduct experiments on multiple Chinese datasets annotated for sentiment or emotion, both at the word level and the phrase level, and show that using various forms of sub-character information significantly helps with correctly determining the sentiment of the text, and that combining them achieves the best results.

2 Related Work

Work on sentiment analysis started in the mid 1990's (Wiebe and Bruce, 1995; Hatzivassiloglou and McKeown, 1997), and initially relied heavily on lexicon-based methods and applied mostly to newswire data. Later on, statistical and distributional methods (Pang and Lee, 2005; Wilson et al., 2005; Socher et al., 2011) became prevalent, most recently with Deep Neural Nets (Tang et al., 2015; Poria et al., 2015; Qian et al., 2017). The domain of interest has also shifted, from newswire to social media, in particular blogs (Mei et al., 2007; Yu and Kübler, 2011) and microblogs (Go et al., 2009; Agarwal et al., 2011; Kiritchenko et al., 2014).

Although the availability of sentiment annotated Chinese corpora is limited, Chinese language sentiment analysis has also become an active research area in recent years. Most work in this area fits into three broad categories. One approach relies on bilingual knowledge to first translate the Chinese text into English text, and then leverage the abundance of English resources for sentiment analysis (Wan, 2008). The second focuses on lexical-based or rule-based sentiment scoring. For example, Xianghua et al. (2013) classify the polarity of the text using the HowNet lexicon, while Zhang et al. (2009) use word dependency rules to determine the sentiment of a sentence. The third approach employs supervised learning on a manually tagged dataset using specialized features (Tan and Zhang, 2008) or on automatically labeled data, e.g. Chinese tweets containing unambiguous emoticons (Zhao et al., 2012). Shared tasks relevant to Chinese sentiment analysis have become prevalent in recent years, and include the SIGHAN 2015 task on Topic-Based Chinese Message Polarity Classification (Liao et al., 2015), the IALP 2016 task on Dimensional Sentiment Analysis for Chinese Words (Yu et al., 2016b), and the upcoming IJCNLP 2017 task on Dimensional Sentiment Analysis for Chinese Phrases.

Work utilizing radicals and other sub-character components is fairly uncommon. One line of research which has become increasingly popular is focused on augmenting word- and character-level embeddings with sub-character information. Sun et al. (2014) and Li et al. (2015) used radicals to enhance the C&W model (Collobert and Weston, 2008) and the word2vec model (Mikolov et al., 2013), respectively. Yin et al. (2016) and later Yu et al. (2017) had shown that word embeddings of the CWE variety (Chen et al., 2015) created from a combination of word-level, character-level, and sub-character-level information outperformed those coming from a single granularity level on semantic tasks. Yu et al. (2017), in particular, show that in addition to radicals, other sub-character components are useful as well.

Ke and Hagiwara (2017) used embeddings created from the radicals of characters and used them in sentiment classification. They showed that their model performs as well on this task with these embeddings as with character-level embeddings, which require a higher-dimensional model and many more parameters. This is the only work, to our knowledge, which uses sub-character components for a sentiment task. Their work differs from ours in several ways, the most important being that they aim to use the radical-level embeddings *instead* of the character-level ones, showing that they can replicate the performance with fewer parameters; in contrast, our work investigates whether or not sub-character components contain useful sentiment information *beyond* that of contextual embeddings, and shows that they complement one another. In addition, we explore the use of non-radical components, in addition to radicals.

The only work, to our knowledge, which makes use not of a list of components but of the order of strokes (Bishun), which are the atomic units of Chinese characters, is by Mi et al. (2016) who used the stroke order predict the correct pronounciation of a character.

3 Approach

Since we are interested mostly in showing the value of the sub-character information, our focus is on performing experiments with various tasks, data sets and representations, and less on the model used in classification. We therefore perform all experiments with a single, straightforward Neural Network (NN) architecture, described below. In addition to using the radicals from a provided list, we devised a second representation of sub-character components, derived directly from the stroke order of the character.

3.1 Character level Embedding

Word embeddings have been very popular in recent years because of the significant improvement they brought about in almost all the subfields of NLP. Across these subfields, this meant not only a good way of dealing with the dimensionality problem, which is often encountered with one-hot encoding, but also a completely unsupervised, i.e. cheap, solution to create semantic spaces that encode most of the relationships among words in the vocabulary of a language.

The idea of encoding each word as a D-dimensional vector is not new (Levy et al., 2015); however, since the publication of the *word2vec* (Mikolov et al., 2013) paper we finally have a method that encompasses the algorithm together with the right negative sampling approach and hyper-parameters. In the paper, the authors explain that in order to compute the vectors representing the words w_i of a certain vocabulary V (of dimension $|V|$), it suffices to use a one hidden layer NN that tries to predict the current word given the neighboring words (CBOW) or the other way around (Skip-Gram).

The optimization function that aims at maximizing the probability between a word w and a context c is thus expressed as follows:

$$p(w|c) = \frac{e^s(w,c)}{\sum_{i=1}^{|V|} e^s(w_i,c)} \tag{1}$$

By making the hidden layer of a much lower dimensionality than $|V|$ we end up with word representations that are much lighter (we can now represent each word with only D dimensions) and bear semantic value (words that appear in similar contexts have vectors that are closer to each other in the semantic space).

In the work we present in this paper, we wanted to use Chinese word embeddings instead of a one-hot representation to take advantage of these properties. However, since our goal was also to investigate an approach that does not rely on heavy preprocessing (such as word segmentation) and that could work equally well on words, phrases and sentences, we found it challenging to use *word2vec*. A more convenient approach, which we employ here, is *fastText* (Bojanowski et al., 2017). This approach relies on the same intuition as *word2vec*, but has the advantage that it builds embeddings for the character n-grams that compose a word. By taking morphology into consideration, *fastText* is able to build embeddings for unseen words (including words with typos) which *word2vec* cannot. From a Chinese morphology perspective, however, this allows to build embeddings for a word, phrase or sentences using its constituent characters without the need of any preprocessing. In a sense, this is similar to computing the vector representing a sentence as the average of the *word2vec* vectors of its constituent words. Despite the simplicity of this approach and its undermining of syntax, it has proved to work very well in combination with deep dense networks yielding results that surpass those obtained with LSTMs (Iyyer et al., 2015). Our choice of learning model, which we describe in Section 3.2, is based on this idea.

3.2 Our Learning Machine

As we previously mentioned, Deep Averaging Networks (DANs) (Iyyer et al., 2015) is one of the most successful approaches to classifying embedded representations. As the authors describe in the paper, the results show that through applying N layers of non-linearity, the network is capable of boosting/shrinking the values of the dimensions that most/least contribute to the classification task. In their work, the authors have a first layer that computes the pointwise average embedding of the words in a sentence as follows:

$$av = \frac{\sum_{i=1}^{W} w_i}{W} \tag{2}$$

In our architecture, this layer is removed and the averaging operation is delegated to *fastText* as we want it to be performed at

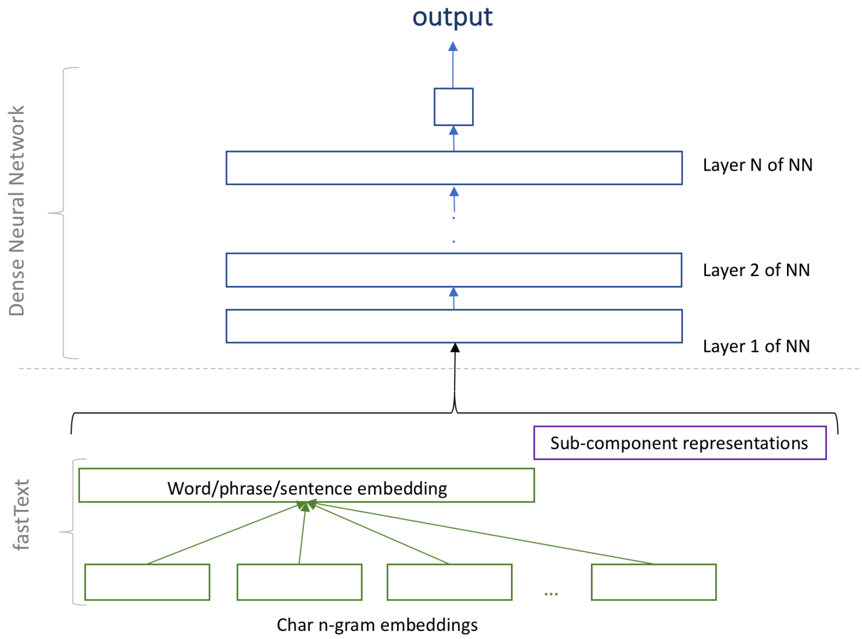

Figure 1: Architecture of a dense NN using fastText embeddings as an input. Output is a one dimension layer in case of regression and softmax for classification.

the character n-gram level. The sub component representations are subsequently concatenated (see Figure 1).

At each hidden layer h_i we apply a non-linear function to its input that can be described as:

$$h_i = f(W_i + b_i) \tag{3}$$

Where W_i and b_i are the parameters of the hidden layer. When performing classification, we apply the softmax function to the last layer. The softmax function ensures that our output is a probability distribution over our set of classes.

We experiment with one and three hidden layers and report the results accordingly. We keep the optimization function (adam) and the activation function (ReLU) fixed in all of the reported results. The dimensionality of the embeddings is 300.

3.3 Sub-Component Representations

We use the code made available by Yu et al. (2017) to collect the list of the components (one of which is the radical) for 20,879 characters. In our experiments, we use a one-hot representation for the 214 radicals and concatenate this representation with *fastText* embeddings (Bojanowski et al., 2017).

In addition, we employ a bottom-up approach using the stroke order for each character[1]. From this data, we collect all stroke n-grams for $n = 1 \dots 7$ and sort them by frequency. In our experiments, we use a one-hot representation of the k most frequent n-grams (trying a range of values for k) and concatenate these with the *fastText* embeddings. Unlike the radicals representation above, this approach has the potential of using non-radical sub-character information, and even information coming from combinations of components; it also has the advantage that it comes directly from the order of strokes, of which there are just over 20 types, instead of representing each component as a unique unit.

4 Data Sets

In order to investigate the usefulness of our approach on a variety of tasks, domains and text characteristics (e.g., length and style) we perform experiments on four datasets.

The first data set is the widely used **NTUSD** (Ku and Chen, 2007) - a sentiment dictionary containing binary polarity an-

[1]We scraped the stroke order for 25,723 Chinese characters from `https://bihua.51240.com/`

Data set	Total size	Entry length	# of labels	# of categories
NTUSD	$11,088$	Single word	1	2
CVAW	$3,552$	Single word	2	Continuous
CVAP	$3,000$	Short phrase	2	Continuous
Weibo	$333,044$	Microblog entry	1	4

Table 1: The four data sets and their properties.

notations (positive/negative) for over $11,000$ words.

The next two data sets come from this year's IJCNLP shared task on Dimensional Sentiment Analysis for Chinese Phrases (DSAP). In this task, terms are labeled with two numeric values, one for the *valence* of the term and one for the *arousal*, together comprising the term's location in the valence-arousal affect space (Russell, 1980). The task is evaluated on two data sets: **CVAW**, which contains $2,802$ and 750 annotated single words in its training and test set, respectively (Yu et al., 2016a); and **CVAP**, which similarly contains $2,250$ and 750 short phrases.

Finally, we include the **Weibo** emotion data set, collected by Fan et al. (2014) from Weibo, a Chinese microblogging service, and automatically annotated with emotional content. The data set contains over $333,000$ entries, each labeled with one of four emotions: joy, anger, sadness or disgust. In comparison with the words of NTUSD and CVAW, and even the short phrases of CVAP, the Weibo entries are significantly longer (the longest entries contain over 400 characters) and like most social media, exhibit unusual linguistic style.

In the cases of NTUSD and Weibo, since there is no pre-determined separation into training and test sets, we randomized the data and set apart 10% of the instances as a test set.

Table 1 summarizes the differences between the four data sets.

5 Experiments

We conduct experiments on all four data sets with the following representation combinations. The baseline is the fastText embeddings, without any sub-character information; we then try the embeddings plus our radicals representation, and the embeddings plus the top k n-grams representation for $k \in \{100, 250, 500, 700\}$. Finally, we use the embeddings, radicals, and n-gram representation together.

For each combination, we try both a single-layer NN and a 3-layer NN, to see whether or not depth has a significant impact on the results.

Note that because of the different tasks (and label types), the four data sets require different evaluation metrics. In particular, CVAW and CVAP are evaluated using the mean absolute error (MAE) and the Pearson correlation coefficient (PCC) for valence and arousal separately, while NTUSD and Weibo are evaluated with Micro-F1.

5.1 Results

The results for the single-layer architecture are shown in Table 2, and the results for the three-layer architecture in Table 3.

Across the board, adding the subcomponents representations to the $fastText$ embeddings always outperforms the approach that resorts only to the latter. The only exception observed is when we predict valence for phrases, i.e. CVAP1, in a three layer NN.

For valence, adding sub-component representations reduced the MAE by up to 0.07 points (from 0.91 to 0.839) in a one layer NN, and 0.03 points when using a three layer network; whereas for arousal, the MAE was reduced by 0.18 in (from 1.12 to 0.94) in the one layer NN and 0.104 in a three layer NN. PCC was also improved accordingly.

Similarly, for the NTUSD data set, we obtained an improvement of 3.4 f-score points in the one layer NN (from 61.2 to 64.6) and 0.4 points in a three layer NN.

When classifying long sentences, i.e. Weibo data, we obtained an improvement of 2 points of f-measure in the one-layer NN and up to 6 (0.54 vs 0.60) points of improvements in the 3 layer NN. This result is interesting because it shows how the sub component representa-

	NTUSD	Weibo	CVAW				CVAP			
			Valence		Arousal		Valence		Arousal	
Combination	F1	F1	MAE	PCC	MAE	PCC	MAE	PCC	MAE	PCC
FastText	61.2	59.2	0.91	0.694	1.125	0.436	0.827	0.781	0.658	0.727
FT+radicals	63.5	60.1	0.882	0.71	1.024	0.488	0.803	0.78	0.609	0.755
FT+ngrams(100)	63.4	59.1	0.855	0.724	1.055	0.479	0.832	0.765	0.603	0.756
FT+ngrams(250)	61.9	56.6	0.871	0.735	1.0	0.523	0.861	0.737	0.576	0.758
FT+ngrams(500)	62.6	57.4	0.896	0.726	0.979	0.52	0.825	0.755	0.586	0.758
FT+ngrams(700)	**64.6**	56.7	0.907	0.728	0.949	0.532	0.813	0.755	0.589	0.754
FT+rad.+ng(100)	62.1	60.8	**0.839**	**0.739**	0.982	0.554	0.793	0.764	0.677	0.766
FT+rad.+ng(250)	62.2	59.7	0.861	0.74	0.966	0.557	0.794	0.772	**0.567**	**0.772**
FT+rad.+ng(500)	57.8	**61.6**	0.867	0.742	0.969	0.533	**0.777**	**0.772**	0.586	0.773
FT+rad.+ng(700)	64.3	60.1	0.859	0.739	**0.945**	**0.553**	0.787	0.763	1.869	0.708

Table 2: The experimental results with one layer.

	NTUSD	Weibo	CVAW				CVAP			
			Valence		Arousal		Valence		Arousal	
Combination	F1	F1	MAE	PCC	MAE	PCC	MAE	PCC	MAE	PCC
FastText	63.7	54.1	0.827	0.738	1.029	0.497	**0.652**	**0.847**	0.587	0.765
FT+radicals	62.3	59.3	0.81	0.756	0.975	0.518	0.69	0.821	0.559	0.786
FT+ngrams(100)	63.6	58.3	0.824	0.752	0.972	0.534	0.71	0.803	0.596	0.758
FT+ngrams(250)	62.7	59.9	0.834	0.754	0.953	0.547	0.742	0.79	0.639	0.722
FT+ngrams(500)	61.8	**60.8**	0.798	0.762	0.94	0.549	0.719	0.795	**0.551**	**0.776**
FT+ngrams(700)	63.6	57.2	0.838	0.753	0.93	0.556	0.772	0.773	0.572	0.767
FT+rad.+ng(100)	61.4	60.2	**0.796**	**0.764**	0.948	0.557	0.706	0.821	0.568	0.773
FT+rad.+ng(250)	63.6	60.1	0.809	0.763	0.939	0.554	0.698	0.807	0.624	0.741
FT+rad.+ng(500)	**64.1**	55.7	0.799	0.756	**0.925**	**0.574**	0.769	0.765	0.588	0.757
FT+rad.+ng(700)	63.5	55.5	0.833	0.765	0.948	0.556	0.777	0.766	1.341	0.347

Table 3: The experimental results with three layers.

tions can help maintain a high performance even when the text is long. Using $fastText$ only, however, yields poor results even if we increase the number of hidden layers.

Overall, the sub-component representations consistently improve results although the improvement is bigger for shallower networks.

5.2 Analysis

In this section, we go through a number of examples where the sub-character features were helpful and a few where they introduced errors. In all cases, we try to explain why the difference might have emerged.

In NTUSD, there are many cases where one or both of the variants made the correct prediction while the embeddings-only baseline did not. For example, the baseline predicts that 好学 ("studious") is negative, which is wrong; the two radicals, 女 ("woman") and 子 ("child") are both somewhat more likely to appear in positive characters, which in this case pushes the classifier in the right direction. Other words which are classified correctly by all of our variants, but not the baseline, include 严酷的 ("cruel") and 勇敢的 ("brave").

In the case of 败俗 ("ruined"), the baseline as well as the radicals representation made an error. The ngrams representation, however, got it right. We believe this is because of the radical 贝 ("treasure"), which usually appears in positive characters. In this case, the ngram representation has multiple variants of this radical and some subsequent strokes, which may explain how it can more accurately separate between sets of characters. Other cases like this include 狼心狗肺 ("ungrateful and cold-blooded") and 法西斯党员 ("fascist party members"). In contrast, in the case of 犯过错 ("made a mistake"), the radicals representation made the correct prediction, possibly because of the radical 犭 ("dog"), which despite seeming objective often appears in characters having to do with animals or animal characteristics, which in Chinese tend to appear in negative contexts. The baseline made an incorrect prediction here, and so did the n-gram variants, for reasons that are not entirely clear to us. In general, we expect the ngram representations to be wrong more often for words with rare radicals that may not make the threshold, or with radicals that are composed of many strokes and cannot be represented well by 7-grams.

In some cases, the baseline gets it right while all of our variants fail. In some of these cases, it is not immediately intuitive that these really are subjective words: 命运注定的 (“predestined”) and 有贵族气派的 (“aristocratic”), for example. This semantic ambiguity may make it a task more suitable for embeddings, and the sub-character components could simply be adding noise. Another example where our variants fail is 雄辩 (“eloquent”); in this case, we have two fairly rare radicals - 隹 (“short-tailed bird”) and 辛 (“bitter”), which we likely have sparse data for. In addition, the second radical is more often seen in negative characters, which may in this case push the classifier in the wrong direction.

In CVAW, instead of binary labels, we have continuous dimensions which provides a more granular view. One interesting example from this data set is 异常死亡 (”abnormal death”), which has a valence of 1.42, very negative. With embeddings alone, the classifier ends up with a very bad prediction: 6.38 - far into the positive side. This is likely because the first two characters of 异常死亡 are not negative, while the last two (both having to do with death) appear in a diverse context which is not always (perhaps not often) negative. The radical 歹 (”death”) of the third word, however, is a clearly negative radical which pushes our variants towards the negative end, arriving at a prediction of 4.97 - still not great, but on the negative side of valence. Similar examples include 极为优秀 (“very good”) and 本来有点同情 (“originally a bit sympathetic”).

The sub-character components add much more to arousal prediction, however. It may be because arousal is less likely to be modeled well in embeddings (since the context for similar words with different arousal levels can be very similar), while some radicals model it directly. The word 极为震怒 (“extremely angry”) has a gold arousal value of 8.56, very high. the embeddings alone predict 4.21, which is far from it and on the low arousal side. With radicals, we arrive at 5.46, much closer and on the high arousal side. This is likely because of two radicals associated with higher arousal, on average: 心 (”heart”) and 雨 (”rain”). The stroke ngrams, in this case, do better than the baseline but not as well as the radicals: 4.91. In other cases, such as 很担心 (“very worried”), the ngrams perform significantly higher than the radicals.

Although interesting, examples from the longer texts in CVAP and Weibo are very difficult to analyze. We leave it to future work to explore these data sets beyond our quantitative evaluation.

6 Conclusion

We showed through experiments on multiple data sets that sub-character components, represented either as a set of radicals or as stroke n-grams, contain information that is useful in sentiment classification beyond the semantic information encoded in character-level embeddings. We showed that with a few exceptions, this effect can be seen with a variety of text lengths and linguistic styles, as well as with varying model depths.

One problem that is inherent to both the $word2vec$ and $fastText$ approaches is that the embeddings of negative and positive sentiment words, e.g. *good* and *bad*, tend to be very similar because they occur in similar contexts; similar behavior exists for emotional dimensions other than polarity (e.g., arousal). In ideographic languages such as Chinese, we can leverage the fact that the characters themselves contain sentiment cues which cannot easily be found with a distributional approach.

We illustrated with specific examples the advantages and disadvantages of the two representations, and showed experimentally that they are in fact complementary, and we can generally achieve the best performance by using both. We also show that using sub-character components yield much more improvement when dealing with long text. We leave the exploration of additional useful representations, as well as the best model to use them with, to future work.

References

Apoorv Agarwal, Boyi Xie, Ilia Vovsha, Owen Rambow, and Rebecca Passonneau. 2011. Sentiment analysis of twitter data. In *Proceedings* *of the Workshop on Languages in Social Media*. LSM '11, pages 30–38.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics* 5:135–146.

Xinxiong Chen, Lei Xu, Zhiyuan Liu, Maosong Sun, and Huan-Bo Luan. 2015. Joint learning of character and word embeddings. In *IJCAI*. pages 1236–1242.

Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th International Conference on Machine Learning*. ICML '08, pages 160–167.

Rui Fan, Jichang Zhao, Yan Chen, and Ke Xu. 2014. Anger is more influential than joy: Sentiment correlation in weibo. *PloS one* 9(10):e110184.

Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford* 1(2009):12.

Vasileios Hatzivassiloglou and Kathleen McKeown. 1997. Predicting the semantic orientation of adjectives. In *Proceedings of the Joint ACL/EACL Conference*. pages 174–181.

M. Iyyer, V. Manjunatha, J. Boyd-Graber, and H Daume III. 2015. Deep unordered composition rivals syntactic methods for text classification. In *ACL*. pages 1681–1689.

Y. Ke and M. Hagiwara. 2017. Radical-level Ideograph Encoder for RNN-based Sentiment Analysis of Chinese and Japanese. *ArXiv e-prints* .

Svetlana Kiritchenko, Xiaodan Zhu, and Saif M Mohammad. 2014. Sentiment analysis of short informal texts. *Journal of Artificial Intelligence Research* 50:723–762.

Lun-Wei Ku and Hsin-Hsi Chen. 2007. Mining opinions from the web: Beyond relevance retrieval. *Journal of the American Society for Information Science and Technology* 58(12):1838–1850.

O. Levy, Y. Goldberg, and I Dagan. 2015. Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics* 3:211–225.

Yanran Li, Wenjie Li, Fei Sun, and Sujian Li. 2015. Component-enhanced chinese character embeddings. In *EMNLP*. pages 829–834.

Xiangwen Liao, Binyang Li, and Liheng Xu. 2015. Overview of topic-based chinese message polarity classification in sighan 2015. In *Proceedings of the Eighth SIGHAN Workshop on Chinese Language Processing*. Beijing, China, pages 56–60.

Qiaozhu Mei, Xu Ling, Matthew Wondra, Hang Su, and ChengXiang Zhai. 2007. Topic sentiment mixture: Modeling facets and opinions in weblogs. In *Proceedings of the 16th International Conference on World Wide Web*. WWW '07, pages 171–180.

Chenggang Mi, Yating Yang, Xi Zhou, Lei Wang, Xiao Li, and Tonghai Jiang. 2016. Exploiting bishun to predict the pronunciation of chinese. *Computación y Sistemas* 20(3):541–549.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*. ACL '05, pages 115–124.

Soujanya Poria, Erik Cambria, and Alexander Gelbukh. 2015. Deep convolutional neural network textual features and multiple kernel learning for utterance-level multimodal sentiment analysis. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. pages 2539–2544.

Qiao Qian, Minlie Huang, Jinhao Lei, and Xiaoyan Zhu. 2017. Linguistically regularized lstm for sentiment classification. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vancouver, Canada, pages 1679–1689.

J.A. Russell. 1980. A circumplex model of affect. *Journal of personality and social psychology* 39(6):1161–1178.

Richard Socher, Jeffrey Pennington, Eric H. Huang, Andrew Y. Ng, and Christopher D. Manning. 2011. Semi-supervised recursive autoencoders for predicting sentiment distributions. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. EMNLP '11, pages 151–161.

Yaming Sun, Lei Lin, Duyu Tang, Nan Yang, Zhenzhou Ji, and Xiaolong Wang. 2014. Radical-enhanced chinese character embedding. *CoRR* abs/1404.4714.

Songbo Tan and Jin Zhang. 2008. An empirical study of sentiment analysis for chinese documents. *Expert Syst. Appl.* 34(4):2622–2629. https://doi.org/10.1016/j.eswa.2007.05.028.

Duyu Tang, Bing Qin, and Ting Liu. 2015. Document modeling with gated recurrent neural network for sentiment classification. In *EMNLP*. pages 1422–1432.

Xiaojun Wan. 2008. Using bilingual knowledge and ensemble techniques for unsupervised chinese sentiment analysis. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. EMNLP '08, pages 553–561.

Janyce Wiebe and Rebecca Bruce. 1995. Probabilistic classifiers for tracking point of view. In *Proceedings of the AAAI Spring Symposium on Empirical Methods in Discourse Interpretation and Generation*. pages 181–187.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*. HLT '05, pages 347–354.

Fu Xianghua, Liu Guo, Guo Yanyan, and Wang Zhiqiang. 2013. Multi-aspect sentiment analysis for chinese online social reviews based on topic modeling and hownet lexicon. *Knowledge-Based Systems* 37:186–195.

Rongchao Yin, Quan Wang, Peng Li, Rui Li, and Bin Wang. 2016. Multi-granularity chinese word embedding. In *EMNLP*. pages 981–986.

Jinxing Yu, Xun Jian, Hao Xin, and Yangqiu Song. 2017. Joint embeddings of chinese words, characters, and fine-grained subcharacter components. In *EMNLP*.

Liang-Chih Yu, Lung-Hao Lee, Shuai Hao, Jin Wang, Yunchao He, Jun Hu, K. Robert Lai, and Xuejie Zhang. 2016a. Building chinese affective resources in valence-arousal dimensions. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. San Diego, California, pages 540–545.

Liang-Chih Yu, Lung-Hao Lee, and Kam-Fai Wong. 2016b. Overview of the IALP 2016 shared task on dimensional sentiment analysis for chinese words. In *2016 International Conference on Asian Language Processing, IALP*. Tainan, Taiwan, pages 156–160.

Ning Yu and Sandra Kübler. 2011. Filling the gap: Semi-supervised learning for opinion detection across domains. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning*. CoNLL '11, pages 200–209.

Changli Zhang, Daniel Zeng, Jiexun Li, Fei-Yue Wang, and Wanli Zuo. 2009. Sentiment analysis of chinese documents: From sentence to document level. *J. Am. Soc. Inf. Sci. Technol.* 60(12):2474–2487.

Jichang Zhao, Li Dong, Junjie Wu, and Ke Xu. 2012. Moodlens: An emoticon-based sentiment analysis system for chinese tweets. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, New York, NY, USA, KDD '12, pages 1528–1531.