Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 5714–5720
Marseille, 11–16 May 2020

Ciron: a New Benchmark Dataset for Chinese Irony Detection

Rong Xiang[1], Xuefeng Gao[2], Yunfei Long[34], Anran Li[2], Emmanuele Chersoni[2], Qin Lu[1], Chu-Ren Huang[2]
The Hong Kong Polytechnic University[12], University of Nottingham[3], University of Essex[4]
Department of Computing, 11 Yuk Choi Road, Hung Hom, Hong Kong (China)[1]
Chinese and Bilingual Studies, 11 Yuk Choi Road, Hung Hom, Hong Kong (China)[2]
NIHR Nottingham Biomedical Research Centre, Nottingham (UK) [3]
School of Computer Science and Electronic Engineering, University of Essex, Colchester (UK)[4]
xiangrong0302@gmail.com[1], csluqin@comp.polyu.edu.hk[1],
{xue-feng.gao,an-ran.li}@connect.polyu.hk[2], yunfei.long@nottingham.ac.uk[3],
{emmanuele.chersoni,churen.huang}@polyu.edu.hk[2]

Abstract

Automatic Chinese irony detection is a challenging task, and it has a strong impact on linguistic research. However, Chinese irony detection often lacks labeled benchmark datasets. In this paper, we introduce Ciron, the first Chinese benchmark dataset available for irony detection for machine learning models. Ciron includes more than 8.7K posts, collected from Weibo, a micro blogging platform. Most importantly, Ciron is collected with no pre-conditions to ensure a much wider coverage. Evaluation on seven different machine learning classifiers proves the usefulness of Ciron as an important resource for Chinese irony detection.

Keywords: Irony detection, Chinese benchmark dataset, social media text, text processing

1. Introduction

The development of the social web has stimulated the use of figurative and creative language in public including the use of irony. As a special kind of figurative device, the most striking feature of irony is the incongruity between the literal meaning and the contextual meaning of an ironic sentence (Farías et al., 2016). Although a unanimous definition of irony is still lacking in the literature, it is often identified as a trope whose actual meaning differs from what is literally enunciated.

Due to its nature, irony detection is very important for natural language processing (NLP) tasks, especially if tasks aim at automatically understanding human languages. Indeed, automatic irony detection has a large potential for various applications in the domain of text mining, especially those that require semantic analysis, such as author profiling, online harassment and hate speech detection, and perhaps, the most well-known task of affective analysis.

Compared to other text analysis tasks, irony detection has received limited computational treatment (Barbieri and Saggion, 2014; Ghosh et al., 2019). Affective analysis works started to analyze and summarize linguistic features of irony from a sentiment shifting perspective in order to allow for its computational formalization (Ebert et al., 2015; Long et al., 2019). While theories based on English have provided a relatively comprehensive map of irony, there is still a lack of literature dealing with non-Indo-European languages. Even though irony is a pervasive linguistic phenomenon, some of its features vary in different cultures and in structural properties of a specific language (Xing and Xu, 2015). For example, Karoui and colleagues (2019) suggested capitalized words as a strong hint of irony in English. Yet, this hint does not work for Chinese as there is no capitalization or other obvious lexical variations in surface forms in Chinese text. Studies on irony detection should take into account the specific ways in which irony is expressed in a given culture and a given language. Otherwise, the capacity of automatic systems in modeling the notion of "context" will always be limited (Van Hee, 2017).

Chinese irony detection in social networks is more challenging because the language of social networks is mostly composed of short statements (Li and Huang, 2019). Few works to date have tried to investigate Chinese irony detection in social networks (Tang and Chen, 2014). However, existing resources are limited to linguistic studies of Chinese irony detection to describe certain lexical patterns as well as syntactic patterns which cannot be readily formulated for machine learning algorithms. The lack of training data for Chinese irony is still a bottleneck for developing computationally-intensive, broad-coverage Chinese irony detection models.

To solve this problem, this work aims to first explore the characteristic features of irony of the Chinese language and also to provide a benchmark dataset that can be used for automatic irony detection. In this paper, we present the new dataset which includes 8.7K short statements labeled for their degree of irony by native speakers, referred as Ciron https://github.com/Christainx/Ciron. Ciron is the first Chinese resource for Chinese irony detection with such a volume of data and fine-grained annotation. In contrast to many NLP datasets that are crowdsourced, instances in Ciron are collected from Chinese

microblogs in a grounded and more natural context. The annotation process ensures consistency and quality. The dataset is applied to several popular machine learning-based methods to demonstrate its effectiveness including Naive Bayes (NB), logistic regression (LR), support vector machine (SVM), convolutional neural networks(CNN) (Kim, 2014), long short-term memory networks(LSTM) (Tang et al., 2015), bidirectional LSTM with attention mechanism (BiLSTM-AT) (Zhang et al., 2018) and Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2019).

The rest of this paper is organized as follows. Section 2 introduces related works on irony theories and illustrates typical examples of ironic sentences in Chinese. Section 3 provides an in-depth description of the data that we extracted from Weibo forums, together with the formalization of the irony features. The performance of the baseline models is evaluated in Section 4, highlighting its validity. Section 5 concludes this paper.

2. Related Works

Based on several linguistic studies (Huang et al., 2017), we define ironic text as those expressions showing discrepancy/incongruity between the literal meaning and the actual/contextual meaning. The representation of irony-related figurative methods attracted a lot of attention based on social media corpora because the polarity reversal from the literal to the contextual meaning of the words poses a serious challenge to text mining tasks such as affective analysis (Reyes and Rosso, 2014).

In recent years, irony detection in NLP has become one of the most arduous and attractive research topics. There are two main approaches in irony detection: rule-based approaches and machine learning approaches (Joshi et al., 2017). A rule-based approach in irony identification mainly relies on lexicons and syntactic patterns, while a machine learning approach combines different types of features and knowledge bases to detect irony (Maynard and Greenwood, 2014; Khattri et al., 2015). In addition to lexical features, some other types of features can contribute to detecting irony, depending on the text genre. These features included but not limited to sentiment words, punctuation (Carvalho et al., 2009; Buschmeier et al., 2014), emoticon (Buschmeier et al., 2014), emotional scenarios (Reyes and Rosso, 2014), as well as reversals (Li et al., 2019). Meanwhile, Support Vector Machine (SVM) and Logistic Regression (LR) remain the most popular models in classical machine learning approaches for irony detection (Ghosh et al., 2015; Li et al., 2019).

Some recent works also tried to tackle irony identification using deep learning methods. Deep learning methods no longer need feature engineering and have shown superior abilities to complex word composition in text (Ghosh and Veale, 2016; Huang et al., 2017). A system based on Long Short-Term Memory (LSTM) model with a multi-task learning strategy recently performed very well in the irony task (Wu et al., 2018).

A large number of studies on irony detection have been conducted for English text. But, attempts on Chinese irony detection are still quite limited (Van Hee et al., 2018). Tang and Chen used a number of linguistic patterns to extract posts from Yahoo Blog as potential irony instances and then manually checked to identify irony instances for Traditional Chinese in Taiwan (2014). The linguistic patterns used in this work include emoticons, linguistic forms and sentiment hashtags. Potential candidates include sentences with negative emoticons and positive words, a condition assumed as a cue for irony reversal. After manual check, 1,005 posts were confirmed to contain irony and now form the National Taiwan University (NTU) Irony Corpus. The selection as candidates for this corpus is based mainly on five kinds of patterns: (1) degree adverbs+positive adjectives; (2) the use of positive adjective with high intensity; (3) the use of positive nouns with high intensity; (4) The use of ‘很好 (very good)’ ; and (5) “可以再…一点 (It’ s okay to be worse)” . Hyperbole is another rhetorical figure frequently found in irony text, e.g. ” 你真是全宇宙最有权力的人！ ” (You are the most powerful person in the universe!). Results showed that these patterns were limited, and thus, extracted irony text are limited to the identified patterns. Another issue with this corpus is that it only contains positive examples, and thus not quite suited for machine learning purpose when used alone.

An irony identification task was reported for simplified Chinese using Weibo data (Deng et al., 2015). They built a feature-based system including six features inspired by the typical characteristics of ironic sentences in Chinese social media Weibo: (1) the basic emotion keywords; (2) Chinese homo-phonic words; (3) the repetition of punctuation; (4) the length of text; (5) “BEI + V” (bei “被” is a pseudo passive form in Chinese); (6) affective imbalance. In the evaluation, the authors implemented five traditional machine learning based classifiers and results showed that the logistic regression model has the highest precision rate and the decision tree Model has the highest recall rate. Furthermore, Jia et al. (2019) experimented on a three-way decisions based feature fusion method, which yielded an improvement over traditional one-step classification in irony detection.

On the theoretical side, Huang recently claimed that the crucial nature of irony is ”reversal” (Huang, 2019). All features related to the notions of *incongruity* and *opposition* can be seen as tools to identify ironic intent, which lead language users to the correct interpretation of the ironic utterance. Following Huang and Apter’s Reversal Theory (Apter, 2007), Li et al. proposed their own framework to detect Chinese ironic expressions (Li et al., 2019). Firstly, they introduced an Irony Identification Procedure (IIP) to guide the detection of irony and identified seven linguistic devices that can be used for irony reversal in Chinese (Li and Huang, 2019). They used the proposed constructions to query

and retrieve ironic sentences from different corpora. These patterns are helpful to find irony samples in a given corpus. However, they cannot be used as irony detection algorithms due to their limited coverage.

3. Ciron: A New Benchmark Dataset

Most studies in Chinese irony focus on finding irony related language features and patterns that can be used for irony identification (Deng et al., 2015; Tang and Chen, 2014; Jia et al., 2019). These patterns are useful for studying irony specific expressions. However, lexicon-based and syntax-based rules can be too coarse or too specific, resulting in low recall due to limited coverage of ironic instances. Moreover, identified language patterns alone are not suitable for collecting a benchmark dataset because they are often too simple for machine learning algorithms (Li et al., 2019). The lack of sufficient training data becomes the bottleneck for automatic Chinese irony detection. Therefore, it is of big importance to introduce a reasonably large and more diverse dataset suited for training machine learning algorithms as well as serving as a more reliable gold standard for Chinese irony detection.

Since irony is often used in informal writing occasions, Ciron data is collected from Weibo for irony annotation. After removing private user information, we produced a benchmark dataset for Chinese irony detection, referred to as Ciron, a short hand for Chinese Irony. The Ciron dataset contains 8.7K Weibo posts and each post has been labeled by five annotators. All the annotators are postgraduate students, aged between 24 and 30, and all of them are Chinese native speakers.

We follow the definition and view of irony being the expression that makes people experience a reversal during the understanding process. The annotation process follows the Irony Identification Procedure (IIP) (An example is given in Appendix) (Li et al., 2019). Since the understanding of reversal can be subjective, we defined five fine-grained classes for ironic ratings: 1 (not ironic), 2 (unlikely ironic), 3 (insufficient evidence), 4 (weakly ironic), 5 (strongly ironic). Class 2 and 4 are introduced to allow for fine-grained extent in the judgement. Before annotating the corpus, annotators are required to read the annotation instructions and a number of rating samples. Due to the intrinsic bias of subjectivity of different annotators, the inter-rater agreement Fleiss' Kappa results is 0.470, indicating a moderate level of agreement. Three examples rated 5, 3, and 1 are given below.

- E1: **Label: 5 (*strongly ironic*)**

  Sentence: 你顾不到她的时候呢？等她吃了大亏你就开心了。 *(When you cannot take care of her? You should be happy when you wait to see her get hurt.)*

  Judgement: Although 开心 originally means happiness, the speaker does not really mean the listener would be happy seeing *her* being hurt. Instead, the speaker attempts to give a warning for a possible bad situation. The "reversal" between the seemingly positive expression and underlying negative intention indicates that this sentence is an instance of *strongly ironic*.

- E2: **Label: 3 (*insufficient evidence*)**

  Sentence: 亲，幸亏你的生日不是 6 号，要不然送你一盒子铅笔 *(My dearest, fortunately your birthday is not on the 6th. otherwise I would give you a box of pencils.)*

  Judgement: 幸亏 (fortunately) is often used in ironic sentence. But the contextual information for identifying the speaker's intention in this case is missing. Therefore, *insufficient evidence* is given in this case.

- E3: **Label: 1 (*not ironic*)**

  Sentence: 最后一句笑死我了，好形象生动。*(The last sentence made me burst in laughter as it is so vivid with realistic imagery.)*

  Judgement: This sentence describes a funny reading experience, and positive affect is appropriately mentioned with positive lexicon *vivid*. Thus, *not ironic* is marked for this instance.

Figure 1 shows the statistics of the annotation result with respect to the distribution of different classes in a pie chart。the actual number of instances are given in Table 1. Ironic instances (class 4 and 5) only appears in approximately 11.1% of all instances. 88.2% of data falls into class 1 and class 2. class 3 is the unclear group which means, annotators cannot make clear judgement because there is no clear evidence of reversal. However, this group appears in only 0.7% of the collection. Since these data are randomly picked without any precondition, it also reflects the use of irony in real Weibo data.

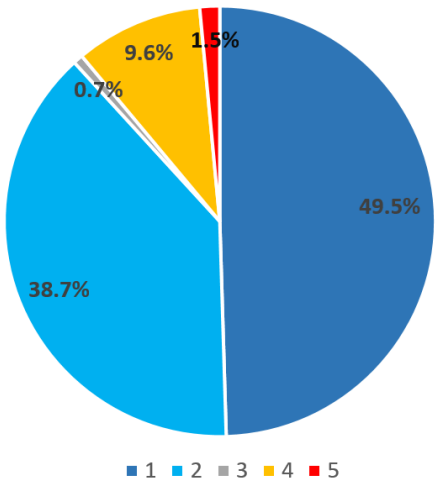

Figure 1: Class Proportion.

From the perspective of computational linguistics, it is important to investigate the use of commonly used lexical and syntactic patterns. We take the five commonly used lexical patterns proposed in two reference

Label	1	2	3	4	5
Count	4,343	3,391	64	838	130

Table 1: Label Frequency.

Class	**1**	**2**	**3**	**4**	**5**
真的	159	146	6	27	3
(really)	46.6	42.8	1.8	7.9	0.9
很好	49	25	0	1	0
(very good)	65.3	33.3	0	1.3	0
幸亏	32	21	4	4	5
(fortunately)	48.5	31.8	6.1	6.1	7.6
可以再	9	10	6	5	1
(It is okay to repeat that)	29.0	32.3	19.4	16.1	3.2
要不是	5	1	0	0	1
(if it were not for)	71.4	14.3	0	0	14.3
Overall Proportion	49.5	38.7	0.7	9.6	1.5

Table 2: Distribution of the five most commonly used lexical patterns in Ciron. Their frequencies and proportion in each class are shown in the first and second row respectively. The overall class proportion is given in the bottom row as a reference.

works (Tang and Chen, 2014; Li et al., 2019) as examples to see how they are distributed in Ciron. The five patterns are listed in Table 2. Table 2 also shows the distribution of these patterns in Ciron both in terms of the total number of occurrence and percentage. Take the most frequently used pattern " 真的" (really) as an example. Supposedly, it should provide a cue for irony. But a detailed look shows that this pattern appears in different classes in a distribution quite consistent to the general distribution of the five different classes given in Table 2. In fact, its occurrence in class 4 and 5 is only about 9%, even less than 11%. When we look at the second frequent pattern " 很好", we can see that it only appears in class 4 and the total percentage is only 1.3%. At least we do see that the last three patterns have higher percentages of distribution in class 4 and class 5 compared to the whole collection, yet the change in distribution is not very large. These observations indicate that machine learning algorithms cannot simply rely on lexical patterns. as they do not have high distributions in the irony samples.

Table 3 shows the general statistics of Ciron. As a benchmark data for machine learning models, the dataset is split using Stratified sampling the ratio 8:1:1 to get the training set, the validation set, and the testing set. With the average length of the posts being 41 characters, each post is likely to be either a long sentence or a number of short sentences.

4. Performance Evaluation

To see how Ciron can be used in irony detection, we used four well known deep learning models.

Dataset Statistics	
Training set size	7,014.0
Validation set size	876.0
Testing set size	876.0
Average sentence number	2.8
Average post length (characters)	41.8
Standard deviation of length (characters)	20.7
Vocabulary (characters)	4,645.0

Table 3: Statistics of data split in Ciron.

4.1. Experimental Settings

We evaluate the performance of three traditional machine learning methods and four deep learning methods on Ciron. Traditional methods include: Naive Bayes (NB), Logistic Regression (LR), and Support Vector Machine (SVM). Deep learning methods include: Convolutional Neural Network (CNN) (LeCun et al., 1998), Long Short-term Memory network (LSTM) (Hochreiter and Schmidhuber, 1997), Bidirectional Long Short-term Memory network with attention mechanism (BiLSTM-AT) (Zhang et al., 2018) and the context aware Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2019).

For the first three deep learning methods, pre-trained GloVe vectors (Pennington et al., 2014) are used as word embedding features. BERT, with the Chinese pre-trained model, is fine-tuned for Ciron. All models are tuned with the datasets. Detailed settings for the models are provided in Table 4. Accuracy and weighted F1 score are adopted as the metrics of performance.

4.2. Analysis of Result

Empirical results are listed in Table 5. In general, traditional methods are outperformed by deep learning methods. NB is the the worst performer as expected. The performance of LR is worse than SVM with a narrow margin. SVM results in a competitive accuracy compared to deep learning methods. Among the deep learning methods, CNNs, generally considered an efficient algorithm for text classification (Kim, 2014), is the worst performer. A potential limit of CNNs for this task is the lack of ability to capture sequential information which is essential for irony detection where polarity reversal can take place in any part of text. Compared to CNNs, LSTM models are better, thanks to their ability to track long dependencies. We did not observe significant improvements with the LSTM model with attention, which performs slightly worse than the basic LSTM in terms of F1-score. The recently-introduced BERT, which is based on a Transformer architecture, has the best performance outperforming all other methods in both accuracy and recall with significant differences. The multi-head attention mechanism contribute to the better representation capability with l

Model	Input	Settings
NB	bag of words vector	
LR	bag of words vector	
SVM	bag of words vector	kernel='rbf', iteration=10
CNN	150 dim word embedding	Optimizer=Adam, Learning Rate=0.0005 Dropout=0.1, epoch=3, conv_window=3
LSTM	150 dim word embedding	Optimizer=Adam, Learning Rate=0.0005 Dropout=0.1, epoch=3
BiLSTM-AT	150 dim word embedding	Optimizer=Adam, Learning Rate=0.0005 Dropout=0.1, epoch=3
BERT	768 dim BERT embedding	Optimizer=Adam, Learning Rate=0.00001 Dropout=0.1, epoch=3, bert-base-chinese

Table 4: Algorithms Settings.

Classifier	Accuracy	F1-score
NB	46.0%	0.464
LR	52.2%	0.502
SVM	53.1%	0.504
CNN	52.5%	0.488
LSTM	54.8%	<u>0.518</u>
BiLSTM-AT	<u>55.3</u>%	0.512
BERT	**60.3**%	**0.572**

Table 5: Accuracy of the models: the best is in **bold** and the second-best is <u>underlined</u>.

4.3. Error Analysis

Based on the performance of BERT, the best algorithm in our evaluation, we present a few incorrect prediction cases below.

- E4: **True Label: 5 (strongly ironic), Predict Label: 1 (not ironic)**

  Sentence: 郭德刚的相声，除了不好笑，其他都很棒。 *(Other than not being funny at all, Guo Degang's cross-talks (Xiangsheng) are perfectly fine.) Note: Guo Degang is a Chinese comedian.*

  Judgement: By using *...are perfectly fine.*, the original sentence seems to compliment Guo Degang's cross-talks. However, being funny is the most crucial property for a cross-talk. The lack of commonsense could lead to a failure to detect this irony. BERT, a transformer-based model, fails to identify the real intention of the speaker and wrongly classifies the instance with the label 1. However, this is more likely to be due to the lack of background knowledge that Guo is a comedian.

- E5: **True Label: 3 (insufficient evidence), Predict Label: 1 (not ironic)**

  Sentence: 意大利的”富二代“一下场就发微薄 ???? *(The Italian players of the "second generation rich" will be micro-blogging once they get off the field????*

  Judgement: The repetition of question marks indicates the speaker is not satisfied with the performance of Italian players. The quotation marks of 富二代, *"second generation rich"* also implies that there could be some special meaning for this term. However, there is no more information to refer to in this sentence. The label is thus 3 in this case, whereas BERT incorrectly assigns it to label 1.

Analyzing the cases above, it is clear that Chinese irony detection is not an easy task. The main issue is the difficulty to infer the underlying intention of a sentence. Comprehension based on commonsense, stance etc. may differ from person to person. Additionally, some patterns mentioned in previous works, e.g. repetition of symbols, may be misleading in some cases. In summary, further research about Chinese irony detection is still in demand.

5. Conclusion

In this work, we introduce the Ciron benchmark data, a new dataset for Chinese irony detection for machine learning. Ciron provides a reasonably large dataset which can be helpful for the development of computational approaches to Chinese irony detection. Ciron, collected from Micro-blog posts, also makes it possible to investigate broad-coverage real-world ironic patterns for Chinese. We show that identifying Chinese irony is a difficult work both in terms of annotation and automatic classification. Also, it is interesting to see that irony is indeed not uncommon as there are in about 11.1% of all posts has some form of irony. As the first benchmark dataset, it provides more opportunity for irony detection for Chinese.

One issue we faced in this study is that irony is still a relatively less used in natural text. This means that a dataset of this size in Ciron still has insufficient examples for machine learning algorithms. Possible future direction is to use Ciron as bootstrapping data to acquire more irony samples so that more training data can be obtained in a semi-automatic method. Other possibility is to consider the fusion of Ciron with other irony corpus available to enlarge training sample size to help machine learning algorithms.

6. Acknowledgements

The work is partially supported by the research grants from Hong Kong Polytechnic University (PolyU RTVU) and GRF grant (CERG PolyU 15211/14E, PolyU 152006/16E).

Yunfei Long acknowledges the financial support of the NIHR Nottingham Biomedical Research Centre and NIHR MindTech Healthcare Technology Co-operative.

7. Bibliographical References

Apter, M. J. (2007). *Reversal Theory: The Dynamics of Motivation, Emotion, and Personality*. Oneworld Publications Limited.

Barbieri, F. and Saggion, H. (2014). Automatic detection of irony and humour in twitter. In *ICCC*, pages 155–162.

Buschmeier, K., Cimiano, P., and Klinger, R. (2014). An Impact Analysis of Features in a Classification Approach to Irony Detection in Product Reviews. In *Proceedings of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 42–49.

Carvalho, P., Sarmento, L., Silva, M. J., and De Oliveira, E. (2009). Clues for Detecting Irony in User-Generated Contents: Oh...!! It's so Easy;-. In *Proceedings of the 1st International CIKM Workshop on Topic-sentiment Analysis for Mass Opinion*, pages 53–56. ACM.

Deng, Z., Jia, X., and Chen, J. (2015). Research on Chinese Irony Detection in Microblog. In *Computer Engineering Science*, volume 37(12), pages 2312–2317.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-Training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of NAACL*.

Ebert, S., Vu, N. T., and Schütze, H. (2015). A Linguistically-Informed Convolutional Neural Network. In *Proceedings of the 6th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 109–114.

Farías, D. I. H., Patti, V., and Rosso, P. (2016). Irony Detection in Twitter: The Role of Affective Content. *ACM Transactions on Internet Technology (TOIT)*, 16(3):19.

Ghosh, A. and Veale, T. (2016). Fracking Sarcasm Using Neural Network. In *Proceedings of the 7th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 161–169.

Ghosh, A., Li, G., Veale, T., Rosso, P., Shutova, E., Barnden, J., and Reyes, A. (2015). Semeval-2015 Task 11: Sentiment Analysis of Figurative Language in Twitter. In *Proceedings of Semeval*, pages 470–478.

Ghosh, D., Musi, E., Upasani, K., and Muresan, S. (2019). Interpreting verbal irony: Linguistic strategies and the connection to thetype of semantic incongruity. *arXiv preprint arXiv:1911.00891*.

Hochreiter, S. and Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.

Huang, Y.-H., Huang, H.-H., and Chen, H.-H. (2017). Irony Detection with Attentive Recurrent Neural Networks. In *Proceedings of ECIR*, pages 534–540. Springer.

Huang, C.-R. (2019). Double Meaning and Reversal: Toward an Empirical Linguistic Account of Irony. In *In 2019 Joint Conference of Linguistic Societies in Korea The 26th Joint Workshop on Linguistics and Language Processing*.

Jia, X., Deng, Z., Min, F., and Liu, D. (2019). Three-Way Decisions Based Feature Fusion for Chinese Irony Detection. *International Journal of Approximate Reasoning*, 113:324–335.

Joshi, A., Bhattacharyya, P., and Carman, M. J. (2017). Automatic Sarcasm Detection: A Survey. *ACM Computing Surveys (CSUR)*, 50(5):73.

Karoui, J., Benamara, F., and Moriceau, V. (2019). *Automatic Detection of Irony: Opinion Mining in Microblogs and Social Media*. John Wiley & Sons.

Khattri, A., Joshi, A., Bhattacharyya, P., and Carman, M. (2015). Your Sentiment Precedes You: Using an Author' s Historical Tweets to Predict Sarcasm. In *Proceedings of the 6th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 25–30.

Kim, Y. (2014). Convolutional Neural Networks for Sentence Classification. *arXiv preprint arXiv:1408.5882*.

LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., et al. (1998). Gradient-Based Learning Applied to Document Recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Li, A.-r. and Huang, C.-R. (2019). A Method of Modern Chinese Irony Detection. In *The 19th Chinese Lexical Semantics Workshop*, pages 273–288.

Li, A.-R., Chersoni, E., Xiang, R., Huang, C.-R., and Lu, Q. (2019). On the "Easy" Task of Evaluating Chinese Irony Detection. In *Proceedings of PACLIC*.

Long, Y., Xiang, R., Lu, Q., Huang, C.-R., and Li, M. (2019). Improving Attention Model Based on Cognition Grounded Data for Sentiment Analysis. *IEEE Transactions on Affective Computing*.

Maynard, D. and Greenwood, M. A. (2014). Who Cares About Sarcastic Tweets? Investigating the Impact of Sarcasm on Sentiment Analysis. In *Proceedings of LREC*. ELRA.

Pennington, J., Socher, R., and Manning, C. D. (2014). GloVe: Global Vectors for Word Representation. In *Proceedings of EMNLP*, pages 1532–1543.

Reyes, A. and Rosso, P. (2014). On the Difficulty of Automatically Detecting Irony: Beyond a Simple Case of Negation. *Knowledge and Information Systems*, 40(3):595–614.

Tang, Y.-j. and Chen, H.-H. (2014). Chinese Irony Corpus Construction and Ironic Structure Analysis. In *Proceedings of COLING*, pages 1269–1278.

Tang, D., Qin, B., and Liu, T. (2015). Learning Semantic Representations of Users and Products for Document Level Sentiment Classification. In *Proceedings of ACL*.

Van Hee, C., Lefever, E., and Hoste, V. (2018). Semeval-2018 Task 3: Irony Detection in English Tweets. In *Proceedings of Semeval*, pages 39–50. Association for Computational Linguistics.

Van Hee, C. (2017). *Can Machines Sense Irony? Exploring Automatic Irony Detection on Social Media*. Ph.D. thesis, Ghent University.

Wu, C., Wu, F., Wu, S., Liu, J., Yuan, Z., and Huang, Y. (2018). Thu_ngn at Semeval-2018 Task 3: Tweet Irony Detection with Densely Connected Lstm and Multi-Task Learning. In *Proceedings of Semeval*, pages 51–56.

Xing, F. Z. and Xu, Y. (2015). A Logistic Regression Model of Irony Detection in Chinese Internet Texts. *Research in Computing Science*, 90:239–249.

Zhang, Y., Wang, J., and Zhang, X. (2018). YNU-HPCC at SemEval-2018 Task 1: BiLSTM with Attention based Sentiment Analysis for Affect in Tweets. In *Proceedings of Semeval*, pages 273–278.

Appendix: Annotation Example

The following sentence is exemplified to elaborate the annotation (IIP) steps.

Label: 5 (strongly ironic)

Sentence: 你顾不到她的时候呢？等她吃了大亏你就开心了。*(What if you cannot take care of her? You should be happy when you wait to see her get hurt.)*

- The first step is to read the entire sentence to sketch a holistic understanding of the meaning. The first sentence in this example is a special question. The complete form of this question could be " 你顾不到她的时候会怎样呢? (What will happen when you cannot take care of her?)". The second sentence is a declarative sentence. The speaker present a consequence (she get hurt) and assume the listener's reaction to this consequence(you should be happy).

- The second step is to determine the contextual meaning of the sentences, especially the core constructions of them. In a given context, the first sentence is an ordinary "special question". The speaker tries to draw listeners' attention to let them imagine the situation when he/she cannot take care of "her(the people who may be mentioned in the context)" and consider the accompanying consequence. The second sentence contains a construction " 等 (wait) + X(replaceable event) + noun/pronoun + 就 (should be) + 开心/高兴/满意 (happy/glad/satisfied) + 了 (past tense)", In this construction, the clause X has to be an event which the subject does not want it happens. If the clause X meet this condition, the constructed meaning will emerge. At the semantic level, the constructed meaning is "noun/pronoun shall be sad/grieved/regret when X happen". At the pragmatic level, the construction expresses a negative evaluation (to current situation) and negative emotion (to expectation) with the word " 等"(wait). From the first sentence we know that the listener always (at least, often) take care of "her", so it is obvious that he/she does not want "her to get hurt". Therefore, the clause X meet the condition. Then the contextual meaning turns to be "you should be regretful when she gets hurt" and the evaluation and sentiment of the sentence are negative.

- The third step is to determine the literal meaning of the sentences. As what are mentioned by Li et al. (2019), literal meanings have to be direct, formal and common. Hence, for the first sentence, the literal meaning is asking the listener to imagine "what will happen when you cannot take care of her". And the literal meaning of the second one is "you will be happy when she gets hurt".

- The last step is to compare the contextual meanings and the literal meanings of the sentences. Apparently, compare with its literal meaning, the contextual meaning of the second sentence experience a reversal. Although the first sentence is in its original meaning, it provides a context to the second one. This helps us to confirm the clause X meet the condition of the construction. Finally, the annotators can judge that this item is likely to be (or contain) and irony.