Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying, pages 6–12
Language Resources and Evaluation Conference (LREC 2020), Marseille, 11–16 May 2020

TOCP: A Dataset for Chinese Profanity Processing

Hsu Yang and Chuan-Jie Lin
Department of Computer Science and Engineering
National Taiwan Ocean University
Keelung, Taiwan ROC

{10757013, cjlin}@ntou.edu.tw

Abstract

This paper introduced TOCP, a larger dataset of Chinese profanity. This dataset contains natural sentences collected from social media sites, the profane expressions appearing in the sentences, and their rephrasing suggestions which preserve their meanings in a less offensive way. We proposed several baseline systems using neural network models to test this benchmark. We trained embedding models on a profanity-related dataset and proposed several profanity-related features. Our baseline systems achieved an F1-score of 86.37% in profanity detection and an accuracy of 77.32% in profanity rephrasing.

Keywords: profanity detection, profanity rephrasing, Chinese profanity processing

1. Introduction

Abusive language is an important issue in the Internet. One of its major subclasses is profanity, which uses explicit profane words to express feelings or to insult other users (Ross *et al*., 2016; Waseem, 2016; Wulczyn, *et al*., 2017). Although profanity is not always abusive (Chen *et al*., 2012; Clarke and Grieve, 2017; Davidson *et al*., 2017) which can appear in positive expressions such as compliments ("This is fxxking awesome"), some readers might still feel uncomfortable thus it is not recommended.

Most of the available datasets nowadays are about abusive language. This issue is highly language-dependent, hence there have been many datasets built in different languages including English (Wassem and Hovy, 2016), German (Wiegand *et al*., 2018; Davidson *et al*., 2017), Dutch (Tulkens *et al*., 2016), Greek (Pavlopoulos *et al*., 2017), Arabic (Mubarak *et al*., 2017), Slovene (Fišer *et al*., 2016), and Indonesian (Alfina *et al*., 2017). It is essential for us native speakers to build datasets in Chinese by ourselves.

In our previous work (Su *et al*., 2017), we have built a small Chinese profanity dataset, which contains 2,044 sentences classified into 29 groups with profanity tagging and rephrasing information. As there are less than 100 sentences in each group, the amount of data is too few for machine learning or deep learning. This is the reason why we want to build a larger dataset.

Many proposed abusive detection systems were built by machine learning (Montani and Schüller, 2018; Tarasova, 2016) or deep learning (Park and Fung, 2017; Gambäck and Sikdar, 2017; Pavlopoulos et al., 2017; Badjatiya et al., 2017; Wiedemann et al., 2018). Besides word embeddings, two major types of features are often adopted.

The content-based features include keywords (Xiang *et al*., 2012), words (Warner and Hirschberg, 2012), character n-grams (Mehdad and Tetreault, 2016), word n-grams (Yin *et al*., 2009; Chen *et al*., 2012), POS n-grams (Davidson *et al*., 2017), and syntactic information (Burnap and Williams, 2014). We would like to see which features is useful for processing Chinese profanity, because Chinese text needs to be segmented but it is hard for a word segmentation system to recognized newly invented profane words.

Because Twitter is the most popular source for building abusive language datasets, another major class of features relates to user profiles or social media, such as gender (Waseem and Hovy, 2016), living place (Waseem and Hovy, 2016), user activities (Dadvar *et al*., 2013; Balci and Salah, 2015), and neighboring posts (Yin *et al*., 2009). We did not use these features because we could not have such information in the dataset or from the source websites.

According to our observations, we think that the main challenges of Chinese profanity processing are as follows:

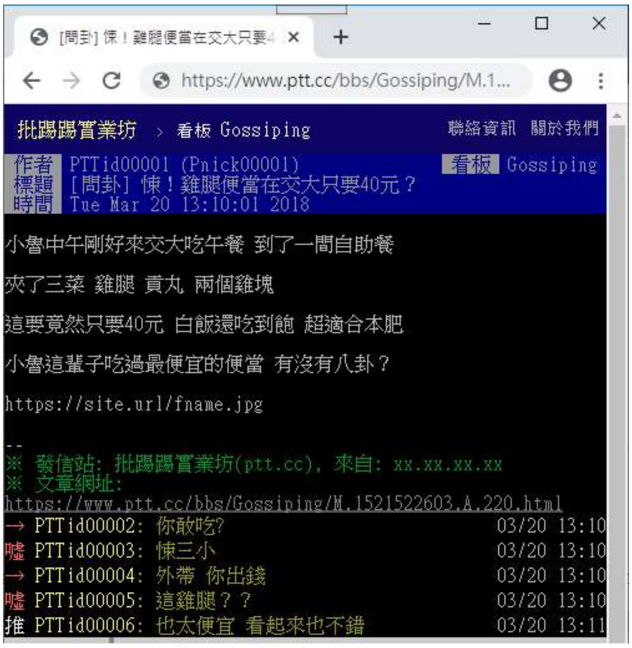
(1) Insufficient training data: larger datasets are beneficial to machine learning and deep learning.
(2) High variety of Chinese profanity: nowadays the Internet users often invent new profane words with different characters with the same or similar soundings to bypass anti-harassment policy.
(3) Profane words in Taiwanese (a dialect commonly spoken in Taiwan): they do not have formal surface forms yet and are often transliterated in many different ways.
(4) Context-based rephrasing: a profane word may have more than one part-of-speech or meaning. Its rephrasing should take its contextual information into consideration.

The TOCP (N**TO**U **C**hinese **P**rofanity) dataset was built for developing Chinese profanity processing techniques. As stated in our previous work (Su *et al.*, 2017), detecting and rephrasing profanity not only reduce the abusive language in the Internet, but also make the text more comprehensible than the simple masking method. Moreover, the users will be educated and more aware of what kinds of expressions are offensive to the others. These reasons make this work important.

This paper is organized as follows. Section 2 describes the construction of TOCP dataset. Sections 3 and 4 propose several baseline systems for profanity detection and rephrasing. Section 5 delivers the evaluation results, and Section 6 concludes the paper.

2. Description of TOCP

We built the TOCP dataset in the similar way as our previous work (Su *et al.*, 2017) but in a larger scale from different websites. As a result, more types of profanity and rephrasing were discovered in this dataset. Details are given in the following subsections.


[問卦] 悚！雞腿便當在交大只要4
https://www.ptt.cc/bbs/Gossiping/M.1...
批踢踢實業坊 › 看板 Gossiping
聯絡資訊 關於我們

```
作者 PTTid00001 (Pnick00001)                    看板 Gossiping
標題 [問卦] 悚！雞腿便當在交大只要40元？
時間 Tue Mar 20 13:10:01 2018

小魯中午剛好來交大吃午餐 到了一間自助餐

夾了三菜 雞腿 貢丸 兩個雞塊

這要竟然只要40元 白飯還吃到飽 超適合本肥

小魯這輩子吃過最便宜的便當 有沒有八卦？

https://site.url/fname.jpg

--
※ 發信站: 批踢踢實業坊(ptt.cc), 來自: xx.xx.xx.xx
※ 文章網址:
https://www.ptt.cc/bbs/Gossiping/M.1521522603.A.220.html
→ PTTid00002: 你敢吃?                                03/20 13:10
噓 PTTid00003: 悚三小                                 03/20 13:10
→ PTTid00004: 外帶 你出錢                             03/20 13:10
噓 PTTid00005: 這雞腿？？                             03/20 13:10
推 PTTid00006: 也太便宜 看起來也不錯                  03/20 13:11
```

Figure 1. An Example of PTT Posts

2.1 Collecting Profanity Data

Most of the teams used crowdsourcing to prepare data annotation (Kolhatkar and Taboada, 2017; Wulczyn *et al.*, 2017). Unfortunately, our main sources of Chinese profanity were text written by Taiwanese users, and we cannot not find a popular crowdsourcing site where we could recruit enough annotators who were native speakers from Taiwan.

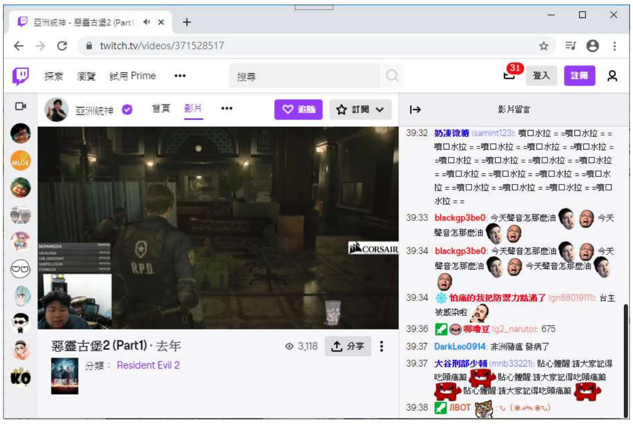
We considered PTT and Twitch as the source websites to collect profanity data. We recruited 10 undergraduate students to annotate profane expressions and provide rephrased expressions.

PTT Bulletin Board System

PTT[1] is a famous BBS site in Taiwan. According to its report[2] on Jan 2020, it has 251 boards related to diverse topics. A top-10 board can be visited by more than 1,000 or even 10,000 users at the same time.

Figure 1 shows an example of the webpage of a PTT post. The leading section shows some metadata about this post, followed by the content of the post, basically in text mode but with some styles of highlights or URLs linking to images in other websites. The title of the post in Figure 1 starts with "悚！" (*Terrifying*!) to express the author's surprise about the low price of a lunchbox in a university. But someone in the comment section replied "悚三小" (*Why the hxll is it terrifying*) which was quite offensive.

Below each post, other users can vote and give comments with a label '推' for like, '噓' for dislike, or '→' for a neutral opinion. Due to the restriction of the length of a line in the comment area, a long comment will be separated into several comment lines, but only the first line will show 'like' or 'dislike' while the other lines will be neutral. Note that a segmenting point is not necessary at the word boundary, which means the characters inside a Chinese word may be separated and appear in two different lines.



39:32 奶凍微糖 (samint123): 噴口水拉 = =噴口水拉 = =噴口水拉 = =噴口水拉 = =噴口水拉 = =噴口水拉 = =噴口水拉 = =噴口水拉 = =噴口水拉 = =噴口水拉 = =噴口水拉 = =噴口水拉 = =噴口水拉 = =噴口水拉 = =噴口水拉 = =噴口水拉 = =噴口水拉 = =噴口水拉 = =

39:33 blackgp3be0: 今天聲音怎那麼油 今天聲音怎那麼油

39:34 blackgp3be0: 今天聲音怎那麼油 今天聲音怎那麼油 今天聲音怎那麼油

39:34 怕痛的我把防禦力點滿了 (gn88019111): 台主被感染啦

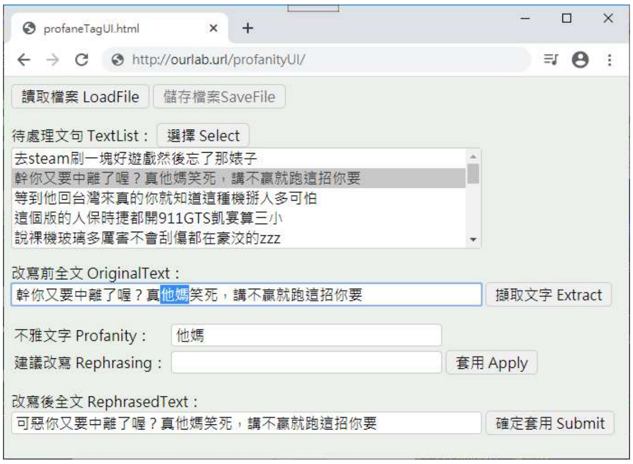
39:36 哪嚕豆 (g2_naruto): 675

39:37 DarkLeo0914: 非洲豬瘟 發病了

39:37 大谷刑部少輔 (mnb33221): 貼心體醒:請大家記得吃頭痛藥 貼心體醒:請大家記得吃頭痛藥 貼心體醒:請大家記得吃頭痛藥

39:38 JIBOT : ଲ（◉ᴥ◉ଲ）

Figure 2. An Example of Twitch Live Streaming



待處理文句 TextList：

去steam刷一塊好遊戲然後忘了那婊子
幹你又要中離了喔？真他媽笑死，講不贏就跑這招你要
等到他回台灣來真的你就知道這種機掰人多可怕
這個版的人保時捷都開911GTS凱宴算三小
說裸機玻璃多厲害不會刮傷都在豪洨的zzz

改寫前全文 OriginalText：幹你又要中離了喔？真他媽笑死，講不贏就跑這招你要

不雅文字 Profanity：他媽

建議改寫 Rephrasing：

改寫後全文 RephrasedText：可惡你又要中離了喔？真他媽笑死，講不贏就跑這招你要

Figure 3. Profanity Annotation Tool

Moreover, if two or more users post comments at the same time, their lines may appear in an interleaving way.

Therefore, the comment lines should be preprocessed to restore the original sentences. Two comment lines were concatenated when (1) they were post by the same user and (2) the latter line was not labeled as 'like' or 'dislike'.

The same as our previous work, we used Google Search to retrieve PTT posts by submitting the profane keywords with the option "site:ptt.cc" for several weeks. We set the searching option for the newest posts in recent one week in order to avoid duplication. Finally, 7,250 posts with 1,043,231 sentences were collected. Only 39,937 of the sentences contain profane keywords.

Twitch Live Streaming

Twitch[3] is a live streaming platform mostly for video game playing. We considered the chatrooms of Twitch channels as a source of profanity, because haters often come to insult or harass the live streamers or the other users. Figure 2 shows an example of a Twitch live streaming channel. The main frame in the middle is the screen showing scenes of game playing, and the area in the right is the chatroom displaying real-time conversations among the viewers.

We monitored 17 streamers by a crawler for two weeks and collected 1,006,434 utterances in their chatrooms, where 14,950 of them contain profane keywords.

[1] https://www.ptt.cc/bbs/index.html

[2] https://www.ptt.cc/bbs/PttHistory/M.1581255677.A.F56.html

[3] https://www.twitch.tv/

```
[
  {
    "ID": "03166_63",
    "orginal_sentence": "幹你又要中離了喔？真他媽笑死，
          講不贏就跑這招你要",
    "source_website": "PTT",
    "profane_expression": [
      {
        "start": 0,
        "end": 1,
        "orginal_expression": "幹",
        "rephrased_expression": "可惡"
      },
      {
        "start": 10,
        "end": 12,
        "orginal_expression": "他媽",
        "rephrased_expression": ""
      }
    ]
  }, ...
]
```

Figure 4. An Example of TOCP data

2.2 Data Annotation

Now we have collected 2,049,665 sentences from social media and 54,887 of them contain profane keywords. It is time-consuming to annotate all the 54 thousand sentences, not to mention checking the other 1.5 million sentences to see if there is any new type of profanity being missed.

As an alternative, we clustered the 54,887 sentences into groups according to the profane keywords and randomly selected sentences to a certain amount in each group. Totally 16,450 sentences were selected

Ten undergraduate students were asked to annotate the real profane expressions in these sentences and provided one possible way to rephrase these expressions into less offensive ones. An annotation tool as shown in Figure 3 was developed for this purpose. If two annotators had different opinions, we would choose the more correct ones.

Finally, there were 17,578 profane expressions being identified in 14,285 sentences. As shown in Figure 4, each of the TOCP data contains an ID, an original sentence, its source web site, and a set of profane expressions appearing in this sentence. Each profane expression is represented by its starting and ending positions, the text of this profane expression, and a rephrasing suggestion.

Please note that the types of Chinese profanity targeted in this paper belong to the following categories.

(1) Terms related to “sexual intercourse”
(2) Terms related to sexual organs or substances
(3) Terms related to “bxtch”
(4) Terms related to “hxll”
(5) Terms in the pattern of “someone's relative's”, a special pattern of profanity in Chinese

All 16,450 sentences are collected in the TOCP dataset provides, including those sentences not containing any profane expressions.

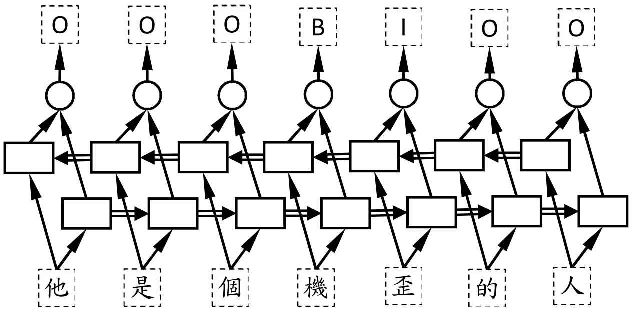


Figure 5. Sequetial-Labeling Profanity Detection

3. Profanity Detection

In this section, we proposed several baseline models to detect profane expressions with different embedding models and features.

3.1 Character-Based Sequence Labeling

The task of profanity detection is to identify profane expressions appearing in an input sentence. However, we think that word-based models may fail due to the limited ability of a Chinese word segmentation system to recognize profane words, especially when these words have many variants and a lot of them are out of vocabulary.

Therefore, we treated the profanity detection problem as a character-based sequence labeling task. Each Chinese character in an input sentence will be tagged with a label of BIO by the classifier to denote if this character is at the beginning (B), inside (I), or outside (O) of a profane expression. The final output of a profanity detection system are substrings in the input sentence tagged with consecutive BI labels.

Figure 5 shows an example of profanity detection by one layer of BiLSTM. The input “他是個機歪的人” (He is a bxtchy guy) is a Chinese sentence with 7 characters. Since the string “機歪” is a profane expression, the correct prediction should be a ‘B’ label for the character ‘機’, an ‘I’ label for ‘歪’, and ‘O’ labels for the other characters.

We tried 1 to 4 layers of BiLSTM, combining with 0 to 2 layers of ConvolutionalNN. Dropout rate was set at 0.5 to avoid overfitting. We also tried different sets of parameters.

3.2 Character Embedding

For embedding, one choice is to use pre-trained embedding models such as Google nnlm-zh-128 model[4] (Bengio *et al.*, 2003). It is a 128-dimension character embedding model trained on Chinese Google News 100B corpus.

However, these available Chinese character embeddings may not meet our needs. The main reason is that the training corpora for these models were general text which did not contain many profane expressions, not to mention those out-of-vocabulary profane words written in the same or similar sounding characters invented by Internet users to bypass anti-harassment policy.

For this reason, we proposed two methods to train profane-related embedding models. The first method was self-training which used one-hot encoding to learn embeddings

[4] https://tfhub.dev/google/nnlm-zh-dim128-with-normalization/2

from the training data directly. In order not to create high-dimensional vectors, we only took characters in the profane expressions and their context (up to 4 characters) into consideration, plus one dimension for “others”.

Our second approach was to train an embedding model based on a profane-related corpus. We used PTT sentences which were not selected into the TOCP dataset to train the profane-related embedding model with a dimension of 100. Training tools were Word2Vec (Mikolov *et al.*, 2013) and fastText[5] (Bojanowski *et al.*, 2016) developed by Facebook AI Research Lab.

3.3 Character Features

Besides embeddings, text itself also provides important features for profanity detection. We designed several features as follows.

Profanity Keywords

The profanity detection rules introduced in Sec 5.1 consist of several sets of profanity keywords. For example, the rule “**YOU** + **RL** + 的” (your relative's) represents a special pattern of profanity in Chinese, where **YOU** is the set of the word “you” (你, 您,…) and **RL** is the set of terms for relatives or acquaintances such as 媽 (mother) or 老師 (teacher). We use two sets of Boolean features. One represents if a character belongs to any of the 45 profanity keyword sets. The other represents if a character belongs to a keyword set in the 24 profanity groups (cf. Sec 5.1).

Dictionary Common Terms

A Chinese character appearing in a profane word may also appear in a common word. For example, the character ‘幹’ has many meanings other than “fxxk”, such as “幹活” (working) or “樹幹” (tree stem). In order to avoid false alarm, we use a Boolean feature to denote if the substring containing the target character is a dictionary common term.

Pronunciation (Pinyin)

Because Internet users often write profane words in different characters with the same or similar soundings to bypass anti-harassment policy, the pronunciation features (*Pinyin* hereafter) were designed to identify these variants. We use two sets of Boolean features, 21 for consonants and 63 for vowels, to represent a character’s pronunciation, and an additional integer feature for the tone of the target character (because Chinese is a tonal language).

These feature vectors would be concatenated with the word embedding vectors to form the input of a neural network.

4. Profanity Rephrasing

We treated the profanity rephrasing problem as a sequence-to-sequence problem. Figure 6 shows a common sequence-to-sequence model by using LSTM. The left part is an encoder which takes a sequence of characters as input, like “機歪” (bxtchy) in the figure. The right part is a decoder which generates a sequence of characters as output, like “機車” (a milder term for ‘bxtchy’) in the figure.

Commonly the input of a sequence-to-sequence model is the text to be rephrased. However, in our observation, contextual information is also important for rephrasing. For example, the character ‘屌’ has many meanings (where the original meaning is “pxnis”) as follows:

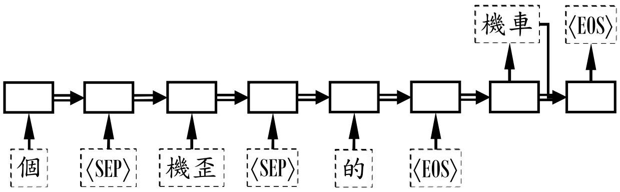


Figure 6. Sequence-to-Sequence Profanity Rephrasing

Original:	金融 好 屌 阿 ～～～
Rephrased:	金融 好 厲害 阿 ～～～
(English:	Finance is so cool~~)

Original:	沒人 屌 你
Rephrased:	沒人 理 你
(English:	No one cares about you.)

So we put context into the input sequences in the format of PREC ⟨SEP⟩ PRFN ⟨SEP⟩ FOLW, where PREC is the preceding context, PRFN is the target profane expression, FOLW is the following context, and ⟨SEP⟩ is a separating symbol. We presume that word-based context is better than character-based, because the meaning can be correctly represented. Note that the output is only the rephrased text.

5. Experiments

All 16,450 sentences in the TOCP dataset were used to do the evaluation. The evaluation method was 10-fold cross-validation.

When evaluating profanity detection systems, the input was a whole sentence and the output was a set of strings recognized as profane expressions. The evaluation metrics were recall and precision based on the number of expressions. Note that an expression should be exactly the same as the human annotation to be counted as correct.

When evaluating profanity rephrasing systems, the input was a profane expression with its context (in its original text) in TOCP and the output was a rephrased string. The evaluation metric was the accuracy for profanity rephrasing, i.e. the ratio of expressions being correctly rephrased.

5.1 Rule-Based Systems

In our previous work (Su *et al.*, 2017), we have designed 29 rules to detect and rephrase profane expressions. Our first effort was to revised these rules according to the cases observed in TOCP. Finally, 41 detection and rephrasing rules (categorized into 24 groups) were formulated.

The performance of rule-based profanity detection is shown in Table 1. The first column shows the ID of profanity groups. Those groups having IDs with the same leading number are related to the same profane keywords. The second column shows the number of profane expressions tagged in TOCP belonging to each group. The overall F1-score is 79.58%.

Please note that there are 1,087 profane expressions which cannot be detected by our rules (denoted as “Other” in Table 1), because there are too many variations but too few examples to deduce general rules. If excluding these outliers, the F1-score becomes 82.08% (denoted as “Apply” in Table 1).

[5] https://fasttext.cc/

Group	#Sents	R	P	F1
1.0	854	51.99	58.12	54.88
1.1	2214	90.61	87.87	89.22
1.2	264	96.59	97.70	97.14
2.0	2760	84.75	59.07	69.61
3.0	1040	83.94	72.87	78.02
4.0	582	95.70	87.72	91.54
4.1	75	74.67	83.58	78.87
5.0	660	90.61	86.04	88.27
5.1	46	65.22	63.83	64.52
6.0	3716	87.65	95.77	91.53
7.0	337	92.28	92.01	92.15
8.0	37	97.30	46.75	63.16
9.0	24	100.00	58.54	73.85
10.0	21	100.00	100.00	100.00
11.0	139	100.00	97.20	98.58
11.1	227	60.35	44.19	51.02
12.0	685	95.62	91.35	93.44
12.1	2	100.00	25.00	40.00
13.0	268	99.25	97.79	98.52
14.0	36	36.11	19.40	25.24
15.0	1016	78.64	62.08	69.39
15.1	836	99.88	96.64	98.24
15.2	576	86.46	85.42	85.94
15.3	76	50.00	86.36	63.33
Other	1087	0.00	0.00	0.00
Total	17578	80.72	78.47	79.58
Apply	16491	86.04	78.47	82.08

Table 1. Performance of Rule-Based Profanity Detection

Group	Acc	Group	Acc
1.0	50.00	9.0	100.00
1.1	85.14	10.0	85.71
1.2	89.77	11.0	17.27
2.0	78.99	11.1	32.16
3.0	66.06	12.0	93.72
4.0	75.60	12.1	100.00
4.1	24.00	13.0	96.27
5.0	88.03	14.0	11.11
5.1	0.00	15.0	54.72
6.0	87.03	15.1	85.41
7.0	89.02	15.2	27.60
8.0	86.49	15.3	10.53
		Other	0.00
Total	71.13	Apply	88.12

Table 2. Performance of Rule-Based Profanity Rephrasing

The performance of rule-based profanity rephrasing is shown in Table 2. The overall accuracy was 71.13%, or 88.12% if the expressions were applicable with the new rules.

5.2 NN-Based Profanity Detection

Several neural network models have been tested, including 1 to 4 layers of bidirectional LSTM combining with 0 to 2 layers of Convolutional NN. The CNN layers were added in front of the BiLSTM layers. Dropout rate was set at 0.5 to avoid overfitting.

Table 3 shows the evaluation results of different NN-based profanity detection systems. We can see that the best systems were a 2-layer BiLSTM with or without a preceding CNN layer.

Model	R	P	F1
BiLSTM	84.77	80.92	82.80
BiLSTM*2	**85.54**	82.17	83.82
BiLSTM*3	85.05	81.00	82.97
BiLSTM*4	84.02	80.98	82.47
CNN + BiLSTM	79.36	78.04	78.69
CNN + BiLSTM*2	85.43	**82.31**	**83.84**
CNN*2 + BiLSTM	74.71	69.90	72.22

Table 3. Performance of NN-Based Profanity Detection

Model	R	P	F1
One-Hot (Char)	85.54	82.17	83.82
One-Hot (Word)	59.56	76.28	66.89
Google nnlm-zh-128	76.45	74.54	75.49
Pinyin	82.52	79.18	80.81
Word2Vec	86.44	84.41	85.41
fastText	85.67	83.70	84.67
Word2Vec + Pinyin	86.05	84.08	85.05
Word2Vec + KW	86.85	85.12	85.98
Word2Vec + KW + Dict	86.38	84.52	85.44
fastText + Pinyin	86.56	84.64	85.59
fastText + KW	**87.50**	**85.26**	**86.37**
fastText + KW + Dict	87.47	84.72	86.07
fastText + KW + Dict + Pinyin	**87.50**	84.53	85.99

Table 4. Comparison of Combinations of Embeddings and Features in Profanity Detection

We also tested different combinations of embedding models and features described in Sections 3.2 and 3.3. Embedding models include one-hot encoding (character-based, word-based, and pinyin-based), Google nnlm-zh-128 model, and our character embedding models trained on PTT sentences by Word2Vec (CBOW model) or fastText (Skip-gram model). Features include pinyin, profanity keywords (KW), and dictionary common terms (Dict).

The experimental results were shown in Table 4 where all systems were built with 2 layers of BiLSTM. The best system was achieved an F1-score of 86.37% by the character embedding trained by fastText combining with the keyword and dictionary-term features. The performance shown in these tables were measured after parameter tuning.

Some conclusions can be drawn from the results in Table 4: (1) The fastText-trained embedding achieved better performance than one-hot encoding, Google nnlm-zh-128, and Word2Vec-trained embedding; (2) The keyword and dictionary-term features improved the performance more than the pinyin feature; (3) The performance of the word-based one-hot encoding was poor, which supported our assumption that incorrect word segmentation would decrease the ability of profanity detection.

Moreover, all NN-based systems outperformed the rule-based detection system either in recall or precision. In the future, we would like to propose hybrid systems which can take advantages from these two kinds of approaches.

5.3 NN-Based Profanity Rephrasing

Our baseline systems for profanity rephrasing mainly differ in the contextual information. Besides using no context, we also took one character or one word preceding or following the target profane expression as context. All systems were built with LSTM models.

Word-Based			Char-Based		
Left	Right	Acc	Left	Right	Acc
0	0	74.83	0	0	73.15
0	1	76.11	0	1	74.12
1	0	**77.32**	1	0	74.43
1	1	76.47	1	1	74.42

Table 5. Performance of NN-Based Profanity Rephrasing

Batch	One-Hot	Word2Vec	fastText
32	76.92	75.71	77.28
64	77.00	75.48	77.06
128	77.32	75.21	76.61
256	76.71	--	--
512	64.21	--	--

Table 6. Comparison of Embeddings and Batch Sizes in Profanity Rephrasing

The choices of embeddings of the target profane expression and the context were the same as the ones in the detection experiments, only that the word-based models were trained on machine word-segmented data.

Table 5 shows the performance of NN-based profanity rephrasing. The systems not using contextual information were the worse systems. Word-based context was better than character-based context. The best system only considered one preceding word and achieved an accuracy of 77.32%, better than the rule-based rephrasing system.

In fact, we also tried to use the whole sentence as context, but the performance was too bad so we did not show the result here.

Table 6 shows the comparison of different embeddings and batch sizes for the best system in Table 5. Because one-hot encoding slightly outperformed the pre-trained word embedding models, it seems that the surface information is as useful as the semantics in profanity rephrasing.

6. Conclusion

This paper introduced TOCP, a larger dataset of Chinese profanity for detection and rephrasing. This dataset contains 16,450 sentences collected from social media websites, where 14,285 of them contains totally 17,578 profane expressions. Rephrasing suggestions to make these expressions less offensive are also provided. This dataset has been released in the Internet[6].

This paper also proposed several baseline systems for profanity detection and rephrasing to evaluate the dataset. Rule-based systems become worse because the rules cannot cover the great variety of profane expressions.

The best profanity detection system consists of two layers of BiLSTM preceded by CNN. Character embeddings were trained by fastText on the PTT sentences, a profanity-related dataset, and concatenated with the profanity keyword feature and dictionary common term feature. The F1-score of detection was 86.37%.

The best profanity rephrasing system took the profane expression and its preceding word as input, where word embeddings came from word-based one-hot encoding and the batch size was set to 128. The accuracy was 77.32%.

We are now building another dataset for abusive language in Chinese. We will observe the similarity and difference between these two datasets.

7. Acknowledgements

This research was funded by the Taiwan Ministry of Science and Technology (grant: MOST 107-2221-E-019-038).

8. Bibliographical References

Alfina, I., Mulia, R., Fanany, M. I., and Ekanata, Y. (2017). Hate speech detection in the Indonesian language: a dataset and preliminary study. In *Proceedings of 2017 International Conference on Advanced Computer Science and Information Systems* (*ICACSIS*), pp. 233-238.

Badjatiya, P., Gupta, S., Gupta, M., and Varma, V. (2017). Deep learning for hate speech detection in Tweets. In *Proceedings of the 26th International Conference on World Wide Web Companion* (*WWW '17 Companion*), pp. 759-760.

Balci, K., and Salah, A. A. (2015). Automatic analysis and identification of verbal aggression and abusive behaviors for online social games. *Computers in Human Behavior*, 53:517-526.

Bengio, Y., Ducharme, R., Vincent, P., and Jauvin, C. (2003). A neural probabilistic language model. *Journal of Machine Learning Research*, 3:1137-1155.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2016). Enriching word vectors with subword information, arXiv:1607.04606.

Burnap, P., and Williams, M. L. (2014). Hate speech, machine classification and statistical modelling of information flows on Twitter: interpretation and communication for policy decision making. In *Proceedings of the Internet, Policy and Politics Conference*, pp. 1-18.

Chen, Y., Zhou, Y., Zhu, S., and Xu, H. (2012). Detecting offensive language in social media to protect adolescent online safety. In *Proceedings of the 2012 ASE/IEEE International Conference on Social Computing and 2012 ASE/IEEE International Conference on Privacy, Security, Risk and Trust*, (*SOCIALCOM-PASSAT '12*), pp. 71-80.

Clarke, I., and Grieve, J. (2017). Dimensions of abusive language on Twitter. In *Proceedings of the First Workshop on Abusive Language Online* (*ALW*-1), *the 55th Annual Meeting of the Association for Computational Linguistics* (*ACL 2017*), pp. 1-10.

Dadvar, M., Trieschnigg, D., and de Jong, F. (2013). Expert knowledge for automatic detection of bullies in social networks. In *Proceedings of the 25th Benelux Conference on Artificial Intelligence* (*BNAIC 2013*), pp. 57-64.

Davidson, T., Warmsley, D., Macy, M., and Weber, I. (2017). Automated hate speech detection and the problem of offensive language. In *Proceedings of the 11th International AAAI Conference on Web and Social Media* (*ICWSM17*), pp. 512-515.

Fišer, D., Erjavec, T., and Ljubešić, N. (2017). Legal framework, dataset and annotation schema for socially unacceptable on-line discourse practices in Slovene. In *Proceedings of the 1st Workshop on Abusive Language*

[6] http://nlp.cse.ntou.edu.tw/resources/TOCP/

Online (*ALW1*), *the Annual Meeting of the Association of Computational Linguistics* (*ACL 2017*), pp. 46-51.

Gambäck, B., and Sikdar, U. K. (2017). Using convolutional neural networks to classify hate-speech. In *Proceedings of the 1st Workshop on Abusive Language Online* (*ALW1*), *the Annual Meeting of the Association of Computational Linguistics* (*ACL 2017*), pp. 71-75.

Kolhatkar, V., and Taboada, M. (2017). Constructive language in news comments. In *Proceedings of the First Workshop on Abusive Language Online* (*ALW*-1), *the 55th Annual Meeting of the Association for Computational Linguistics* (*ACL 2017*), pp. 11-17.

Mehdad, Y., and Tetreault, J. (2016). Do characters abuse more than words? In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue, Association for Computational Linguistics*, pp. 299-303.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representation in vector space. In *the Workshop Track Proceedings of the 1st International Conference on Learning Representations*.

Montani, J. P., and Schüller, P. (2018). TUWienKBS at GermEval 2018: German abusive tweet detection. In *Proceedings of the GermEval 2018 Workshop*, *the 14th Conference on Natural Language Processing* (*KONVENS 2018*), pp. 45-50.

Mubarak, H., Darwish, K., and Magdy, W. (2017). Abusive language detection on Arabic social media. In *Proceedings of the 1st Workshop on Abusive Language Online* (*ALW1*), *the Annual Meeting of the Association of Computational Linguistics* (*ACL 2017*), pp. 52-56.

Park, J. H., and Fung, P. (2017). One-step and two-step classification for abusive language detection on Twitter," *Proceedings of the 1st Workshop on Abusive Language Online* (*ALW1*), *the Annual Meeting of the Association of Computational Linguistics* (*ACL 2017*), pp. 41-45.

Pavlopoulos, J., Malakasiotis, P., and Androutsopoulos, I. (2017). Deep learning for user comment moderation. In *Proceedings of the 1st Workshop on Abusive Language Online* (*ALW1*), *the Annual Meeting of the Association of Computational Linguistics* (*ACL 2017*), pp. 25-35.

Ross, B., Rist, M. Carbonell, G., Cabrera, B., Kurowsky, N., and Wojatzki, M. (2016). Measuring the reliability of hate speech annotations: The case of the European refugee crisis. In *Proceedings of NLP4CMC III: 3rd Workshop on Natural Language Processing for Computer-Mediated Communication. Bochumer Linguistische Arbeitsberichte*, 17:6-9.

Su, H.-P., Huang, Z.-J., Chang, H.-T., and Lin, C.-J. (2017). Rephrasing profanity in Chinese text. In *Proceedings of the First Workshop on Abusive Language Online* (*ALW*-1), *the 55th Annual Meeting of the Association for Computational Linguistics* (*ACL 2017*), pp. 18-24.

Tarasova, N. (2016). *Classification of Hate Tweets and Their Reasons using SVM*, master's thesis, Uppsala Universitet.

Tulkens, S., Hilte, L., Lodewyckx, E., Verhoeven, B., Daelemans, W. (2016). A dictionary-based approach to racism detection in Dutch social media. In *Proceedings of the Workshop Text Analytics for Cybersecurity and Online Safety* (*TA-COS*).

Warner, W., and Hirschberg, J. (2012). Detecting hate speech on the World Wide Web. In *Proceedings of the Workshop on Language and Social Media, The Association for Computational Linguistics*, pp. 19-26.

Waseem, Z. (2016). Are you a racist or am I seeing things? Annotator influence on hate speech detection on Twitter. In *Proceedings of the First Workshop on NLP and Computational Social Science. Association for Computational Linguistics*, pp. 138-142.

Waseem, Z. and Hovy, D. (2016). Hateful symbols or hateful people? Predictive features for hate speech detection on Twitter. In *Proceedings of the NAACL Student Research Workshop. Association for Computational Linguistics*, pp. 88-93.

Wiedemann, G., Ruppert, E., Jindal, R., and Biemann, C. (2018). Transfer learning from LDA to BiLSTM-CNN for offensive language detection in Twitter. In *Proceedings of the GermEval 2018 Workshop, the 14th Conference on Natural Language Processing* (*KONVENS 2018*), pp. 85-94.

Wiegand, M., Siegel, M., and Ruppenhofer, J. (2018). Overview of the GermEval 2018 Shared Task on the Identification of Offensive Language," *Proceedings of the GermEval 2018 Workshop, the 14th Conference on Natural Language Processing* (*KONVENS 2018*), pp. 1-10.

Wulczyn, E., Thain, N., and Dixon, L. (2017). Ex Machina: Personal attacks seen at scale. In *Proceedings of the 26th International Conference on World Wide Web* (*WWW 2017*), pp. 1391-1399.

Xiang, G., Fan, B., Wang, L., Hong, J., and Rose, C. (2012). Detecting offensive tweets via topical feature discovery over a large scale Twitter corpus. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management* (*ACM CIKM '12*), pp. 1980-1984.

Yin, D., Xue, Z., Hong, L., Davison, B. D., Kontostathis, A., and Edwards, L. (2009). Detection of harassment on Web 2.0. In *Proceedings of the Content Analysis in the WEB*, pp. 1-7.