sensors

MDPI

Article

Connecting Text Classification with Image Classification: A New Preprocessing Method for Implicit Sentiment Text Classification

Meikang Chen [1], Kurban Ubul [1,2,*], Xuebin Xu [1], Alimjan Aysa [2] and Mahpirat Muhammat [3]

[1] College of Information Science and Engineering, Xinjiang University, Urumqi 830046, China; xj_ckk@stu.xju.edu.cn (M.C.); xuxuebin@xju.edu.cn (X.X.)
[2] The Key Laboratory of Multilingual Information Technology, Xinjiang University, Urumqi 830046, China; alim@xju.edu.cn
[3] International Cultural Exchange College Xinjiang University, Xinjiang University, Urumqi 830046, China; xmahpu@xju.edu.cn
* Correspondence: kurbanu@xju.edu.cn; Tel.: +86-139-9926-9785

Abstract: As a research hotspot in the field of natural language processing (NLP), sentiment analysis can be roughly divided into explicit sentiment analysis and implicit sentiment analysis. However, due to the lack of obvious emotion words in the implicit sentiment analysis task and because the sentiment polarity contained in implicit sentiment words is not easily accurately identified by existing text-processing methods, the implicit sentiment analysis task is one of the most difficult tasks in sentiment analysis. This paper proposes a new preprocessing method for implicit sentiment text classification; this method is named Text To Picture (TTP) in this paper. TTP highlights the sentiment differences between different sentiment polarities in Chinese implicit sentiment text with the help of deep learning by converting original text data into word frequency maps. The differences between sentiment polarities are used as sentiment clues to improve the performance of the Chinese implicit sentiment text classification task. It does this by transforming the original text data into a word frequency map in order to highlight the differences between the sentiment polarities expressed in the implicit sentiment text. We conducted experimental tests on two common datasets (SMP2019, EWECT), and the results show that the accuracy of our method is significantly improved compared with that of the competitor's. On the SMP2019 dataset, the accuracy-improvement range was 4.55–7.06%. On the EWECT dataset, the accuracy was improved by 1.81–3.95%. In conclusion, the new preprocessing method for implicit sentiment text classification proposed in this paper can achieve better classification results.

Keywords: natural language processing; implicit sentiment analysis; text classification; image classification; data preprocessing

check for updates

Citation: Chen, M.; Ubul, K.; Xu, X.; Aysa, A.; Muhammat, M. Connecting Text Classification with Image Classification: A New Preprocessing Method for Implicit Sentiment Text Classification. *Sensors* **2022**, *22*, 1899. https://doi.org/10.3390/s22051899

Academic Editor: Sheryl Berlin Brahnam

Received: 13 December 2021
Accepted: 25 February 2022
Published: 28 February 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Sentiment analysis is a basic task in natural language processing. Because sentiment analysis can be widely used in specific practical tasks such as comment analysis [1], public opinion analysis [2], mental health analysis [3], recommender systems [4] and spam identification, it has great development prospects. Sentiment analysis tasks are generally divided into explicit sentiment analysis and implicit sentiment analysis. Generally speaking, there are clear sentiment words in explicit sentiment text sentences, which provide obvious sentiment clues for preliminary sentiment analysis. Therefore, it is relatively easy to distinguish the sentiment polarity of these sentences through classical machine learning methods or popular deep learning methods. However, people often express their feelings in an implicit and obscure way in their daily lives. Despite this, implicit sentiment texts, that is, language fragments (sentences, clauses or phrases) that express subjective emotions but do

not contain explicit sentiment words, have rarely attracted researchers' attention. Moreover, the research on implicit sentiment text analysis has also made slow progress. One of the important reasons for this is that the existing text-processing methods cannot extract the implicit sentiment clues contained in the implicit sentiment text. The traditional method based on sentiment words [5] cannot effectively process the text. Another important reason is that in the existing processing methods for implicit sentiment text, the elimination method of redundant words of implicit sentiment needs to be strengthened. According to a survey, the proportion of implicit sentiment in subjective sentences in various fields is as high as 15–20% [6,7]. At present, implicit sentiment analysis has become one of the core problems in the field of natural language processing [5,7].

Please refer to the following examples of explicit and implicit sentiment sentences:

E1: 我们在一起每天都是**开开心心**的。 ("We are **happy** every day together." Note: explicit sentiment sentences; explicit sentiment word: **happy**; sentiment label: **positive.**)

E2: 果然是传说中的奇女子，琴棋书画样样精通啊。 ("Sure enough, she is a legendary woman. She is proficient in piano, chess, calligraphy and painting." Note: implicit sentiment sentence; no sentiment words; sentiment label: **positive.**)

E3: 我的袜子洗了两遍还全是沙。 ("My socks have been washed twice and are still full of sand." Note: implicit sentiment sentence; no sentiment words; sentiment label: **negative.**)

Sentence E1 is an explicit sentiment sentence. The explicit sentiment word "开开心心 (happy)" can be used as an important clue to identify the sentiment polarity of this sentence. Sentence E2 and Sentence E3 are both implicit sentiment sentences. Sentence E2 show a kind of praise by expressing surprise at the talent displayed by the protagonist. Sentence E3 expresses an implicit negative sentiment by describing the fact that the sand in the shoes cannot be cleaned. It can be seen from these examples that explicit sentiment sentences contain clear sentiment words. Therefore, it is easier to judge the sentiment polarity of such sentences through classical machine learning methods or popular deep learning methods. However, in the face of implicit sentiment text, the existing methods based on sentiment words have difficulty obtaining the implicit sentiment polarity clues contained in the sentence.

Considering that it is difficult for implicit sentiment analysis to classify implicit sentiment text based on the explicit sentiment dictionary, a new preprocessing method for implicit sentiment text classification is described in this paper. With the help of a deep learning network, it can extract more implicit sentiment clues from the samples and judge whether two different samples belong to the same implicit sentiment category through some subtle features of the samples. At the same time, it has a new problem surrounding what method to use to extract the implicit sentiment clues in the samples, so as to quickly and accurately judge whether different samples belong to the same implicit sentiment category. In the process of research, we noticed some useful characteristics of Oriented Fast and Rotated Brief (ORB) [8,9] feature points in the field of 3D reconstruction. ORB feature points can judge whether the object in the picture is the same object by extracting similar feature points in different input samples. ORB feature points have a series of advantages, such as scale invariance, local invariance and so on. Additionally, ORB feature points have been widely proved to be effective feature points in the field of 3D reconstruction. Therefore, when designing the corresponding network, this paper refers to the idea of ORB feature point extraction to solve the problem we just mentioned. This paper assumes that implicit sentiment words with different sentiment polarities have different sentiment polarity weight ranges. When multiple texts with the same sentiment polarity are aggregated, due to the aggregation of the sentiment clues contained in multiple implied sentiment words with the same polarity, the weight value belonging to this domain of sentiment polarity can be greatly improved. This difference in weight values caused by the difference between different affective polarities can be regarded as an important feature of implicit sentiment polarity classification in implicit sentiment analysis tasks.

In this paper, we propose a new preprocessing method for implicit sentiment text classification. We named this method Text To Picture (TTP). TTP draws on the idea of

ORB extracting high-quality feature points from samples and applies this idea to the field of implicit sentiment classification; that is, after data enhancement processing of the text, the text is processed by pixel-value conversion, image multivaluing and other processing methods, so as to achieve an effect similar to ORB feature point extraction. Specifically, the samples after data enhancement will eliminate invalid pixels through the set frequency threshold in the process of TTP, converting word frequency value into pixel value, so as to obtain an image with obvious implicit sentiment polarity characteristics, and input it as a new sample to the image-processing network. In the process of selecting an image-processing network, considering the difficulty of extracting implicit sentiment clues in implicit sentiment sentences, fine-grained analysis of samples is needed in sentiment analysis. Therefore, this paper selects the relevant networks in the field of facial micro-expression recognition. To summarize, TTP aggregates the sentiment polarity words in the same sentiment polarity text by converting text into pictures and enhances the weight value of the same sentiment polarity in the input sample, so as to better highlight the sentiment polarity difference between different implicit sentiment texts in the input sample.

The main contributions of this paper are as follows:

1. A simple and effective implicit sentiment text data preprocessing method is proposed, which innovatively converts text features into picture features, so as to better extract the differences between different implicit sentiment polarities;
2. A new research direction of implicit sentiment analysis tasks is proposed: The innovative combination of the two fields of text analysis and image analysis.

The rest of this paper is arranged as follows: the second part introduces related work. The third part introduces the working principle of TTP in detail. The fourth part expresses the experiment and its analysis. Finally, the fifth part relays the conclusion of this paper

2. Related Works

In this part, we mainly introduce the research of implicit sentiment analysis, Synchronous Localization And Mapping (SLAM) and facial micro-expression recognition.

2.1. Implicit Sentiment Analysis

This section briefly describes the development of implicit-sentiment analysis. At the same time, some better processing methods and technologies in the field of implicit-sentiment analysis are introduced.

In the field of natural language processing, in order to encode the text input into the network, early experts and scholars either proposed to use one-hot coding to mark the word vector, proposed to use the word package to represent a piece of text, or proposed to use the Term Frequency Inverse Document Frequency (TF-IDF) algorithm to mine the information in the text before coding. With the deepening of research in the NLP field, Tomas Mikolov et al. [10] proposed a Word2Vector method with higher performance, which promoted the method of word vector processing in the NLP field to a new stage.

In the aspect of association rules, Mankar and Ingle [11] used association-rule mining technology to extract the implicit sentiment features in tourism reviews, constructed the co-occurrence frequency matrix between explicit features and opinion words, and extracted the implicit sentiment features in the dataset based on the co-occurrence frequency matrix. Jiang et al. [12], based on the association-rule mining technology, extended the basic rules extracted from the dataset by using the topic model. Then they removed the redundant part of the extracted feature indicator by using the pruning method. Finally, the method of combining basic rules with new rules was used to extract the implicit sentiment features from the dataset. Ilya et al. [13], based on association-rule mining technology, improved the original rules used to extract implicit sentiment features in English to a certain extent. By replacing the original algorithm mode in rule representation with the algorithm on a sentence syntax tree, the new rules could be better applied to the task of implicit sentiment analysis in Russian. Liu et al. [14] used the method based on association-rule mining to mark each opinion word without explicit features in the dataset as implicit features. They

also extracted a set of association rules from the annotated dataset. Finally, they translated the extracted rules into language patterns and used them for implicit feature extraction. Although these technologies are effective, the association rules that these methods need to design are more complex, and the cost of these methods is higher than the performance advantages they can offer.

2.2. Research on SLAM

This section briefly introduces the development history of and existing research on SLAM, as well as some technologies involved in SLAM. Then, it points out the role of ORB feature point extraction in SLAM.

SLAM originated in 1986, and most of the early visual SLAM programs were based on filtering methods. After continuous development and research, relevant staff found that SLAM's problem was the quick solving of a sparse matrix. The mathematical method for solving sparse matrixes greatly reduces the computational complexity of SLAM. With the continuous development of SLAM, Murray and Klein proposed a breakthrough method in visual SLAM, namely PTAM [15] (Parallel Tracking And Mapping). This method is a parallel algorithm that divides attitude tracking and mapping into two threads, which is very suitable for working in a small workspace. ORB-SLAM [16] and its extended version ORB-SLAM2 [17] were proposed by Mur-Artal et al. They divide tasks into three parallel threads: tracking, local mapping and closed-loop detection. Presently, they have become a classic processing flow in the field of SLAM.

In SLAM's keyframe-based method, only a few selected frames are generally used to estimate the map. The information from the intermediate frame is discarded. This method can perform more costly but more accurate BA (bundle adjustment) optimization on keyframes. The keyframe-based technique is more accurate than the filter-based method at the same computational cost [18]. Based on these ideas, ORB-SLAM [19,20] uses ORB features to solve the problem that the BRIEF descriptor does not have rotation invariance. By using descriptors in ORB to provide short-term and medium-term data association, the construction of common views is realized to limit the complexity of tracking and mapping. This method reduces the amount of calculation in SLAM and increases the accuracy of drawing.

2.3. Facial Micro-Expression Recognition

This part mainly introduces the development of facial micro-expression recognition in recent years and briefly describes some expression-recognition technologies.

In the early 21th century, facial micro-expression recognition began to be studied. With the continuous development of machine learning technology, many micro-expression detection and analysis algorithms are emerging. The most extensive micro-expression detection and analysis method first divides the face into several specific regions. Then the methods of feature extraction and existing classifiers are used for these specific areas, so as to further search and extract the specific relationship corresponding to the specific facial micro-expression features. At present, the method of deep learning is not commonly used in the field of facial micro-expression recognition. The main reasons for this are that the training time is too lengthy and a large number of datasets need to be used to train the system. Among several specific areas divided, because the micro-expression (ME) region is characterized by subtle and rapid change, the difficulty of the existing algorithms is in how to locate the onset frame, apex frame and offset frame accurately and quickly. Yante et al. [21] statistically analyzed the change rate of pixel value in the picture in the frequency domain, then used the information obtained in the frequency domain to locate the apex frame and analyzed the facial micro-expressions by using the local feature information and global feature information of facial micro-expression. Puneet [22] analyzed the spatial and temporal features of faces and used feature coding to encode subtle facial features and then input them to 2D convolutional neural network for recognition. Through the combination of feature coding and convolutional neural network Puneet achieved better ME recognition

performance. Li et al. [23] proposed a no-training method based on feature difference contrast and peak detection to identify ME regions. On the other hand, Ma et al. [24] further improved the performance of start frame positioning by using the histogram of directional optical flow.

In order to improve the performance of ME region recognition and analysis, the existing facial micro-expression recognition algorithms have increasingly tended to focus on pixel-level feature point extraction. At present, these pixel-level feature point extraction methods have achieved good results in analyzing subtle facial expressions.

3. Method

In implicit sentiment analysis tasks, because there is no explicit emotional word as the clue for implicit sentiment analysis, the traditional dictionary-based method is invalid. We must use other methods to analyze and process the implicit sentiment text. After extensive research, this paper proposes a novel and effective implicit sentiment text preprocessing method. The idea of this method comes from the method of ORB feature point extraction in SLAM. Through the extraction of sentiment features from the same implicit sentiment polarity text, we can increase the differences between different sentiment polarities.

The implicit sentiment text preprocessing method proposed in this paper consists of three parts, as shown in Figure 1.

- **Data Enhancement:** Data enhancement is aimed at the problem that there are few sentiment polarity clues contained in a single implicit sentiment text and there are no obvious differences in the weight of sentiment polarity between different implicit sentiment texts. Before extracting sentiment polarity cues from the input implicit sentiment text, TTP enhances the text data of the input samples, and the number of samples after data enhancement is 32 times the original. By aggregating 32 samples after data enhancement to a certain extent, this paper highlights the differences of total weight between different sentiment polarities. The data enhancement method used in this paper is the Natural Language Processing Chinese Data Augmentation (NLPCDA) module [25]. The NLPCDA module supports ten different data enhancement methods. The six data enhancement methods used in this paper are as follows:

  1. Random entity replacement;
  2. Random synonym substitution;
  3. Substitution of random near-synonyms and near-syllable characters;
  4. Random character deletion;
  5. Random permutation of the nearest neighbor characters;
  6. Equivalent substitution of Chinese characters.

  NLPCDA module parameter setting: create_num = 5, change_rate = 0.38.

  It should be noted that the implicit sentiment polarity in the original sentence will not be changed during data enhancement.

In order to better show and explain the effect of input samples after data enhancement, this paper takes Figure 2 as an example for corresponding explanation. The implicit sentiment text shown in Figure 2 was randomly selected from the SMP2019 dataset. From Figure 2, it can clearly be seen that through data enhancement, one sample becomes 32 samples. It is worth noting that Figure 2 illustrates that some samples will be significantly shorter after data enhancement. This situation is caused by the "random character deletion" method mentioned above. The advantage of this is the prevention of overfitting. As for the other five methods used in the process of data enhancement, they do not cause significant changes in sample length.

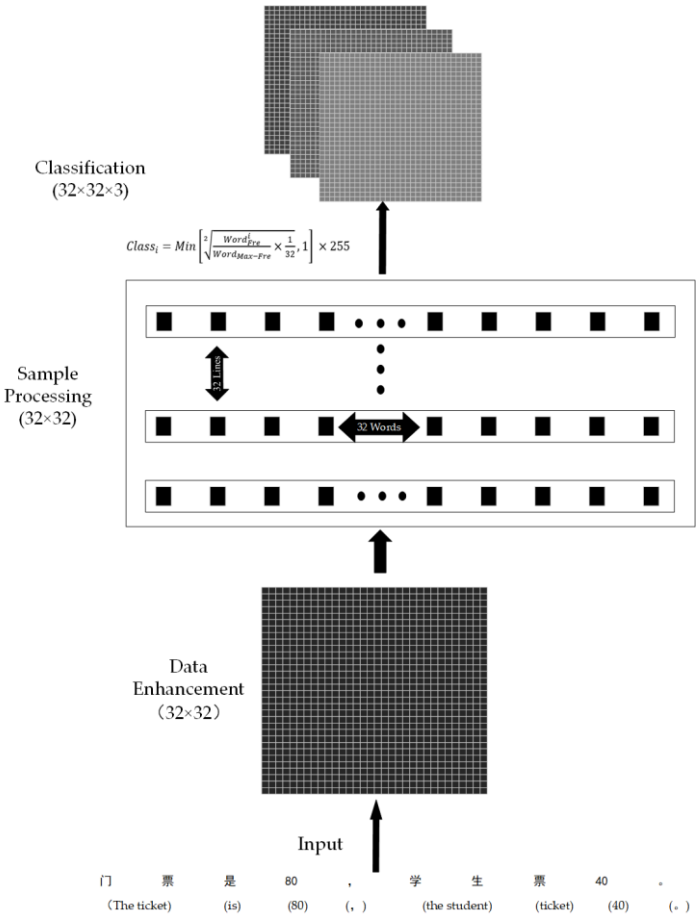

Figure 1. TTP flow diagram.

但 今日 2 位 贾跃亭 身边 人士 先后 向 新浪 科技 交叉 证实， 其实 老贾 还 在 加州， 因 手头 还有 一些 事务 没有 处理 完， 但是 距 回国 不会 太久。
但 今日 2 位 贾跃亭 身边 人士 先后 向 新浪 科技 交叉 证实， 其实 老贾 还 在 加州， 因 手头 还有 一些 事务 没有 处理 完， 但是 距 回国 不会 太久。
但 今日 2 位 贾跃亭 身边 人士 先后 向 中国通信服务 科技 交叉 证实， 其实 老贾 还 在 加州， 因 手头 还有 一些 事务 没有 处理 完， 但是 距 回国 不会 太久。
但 今日 2 位 贾跃亭 身边 人士 先后 向 瑞丰光电 科技 交叉 证实， 其实 老贾 还 在 加州， 因 手头 还有 一些 事务 没有 处理 完， 但是 距 回国 不会 太久。
但 今日 2 位 贾跃亭 身边 人士 先后 向 中国基建投资 科技 交叉 证实， 其实 老贾 还 在 加州， 因 手头 还有 一些 事务 没有 处理 完， 但是 距 回国 不会 太久。
但 今日 2 位 贾跃亭 身边 人士 先后 向 长兴国际 科技 交叉 证实， 其实 老贾 还 在 加州， 因 手头 还有 一些 事务 没有 处理 完， 但是 距 回国 不会 太久。
但 今日 2 位 贾跃亭 身边 人士 先后 向 新浪 科技 交叉 证实， 其实 老贾 还 在 加州， 因 手头 还有 一些 事务 没有 处理 完， 但是 距 回国 不会 太久。
但 今日 2 位 贾跃亭 身边 人士 先后 向 新浪 科技 交叉 证实， 其实 老贾 还 在 加州， 因 手头 还有 有 事务 从不 拍卖 完， 但是 距 回国 不会 太久。
但 今日 2 位 贾跃亭 身边 人士 先后 向 新浪 科技 交叉 求证， 事实上 老贾 还 在 加州， 因 手头 还有 一些 事务 远非 处理 完， 但是 距 回归 不会 太久。
但 今日 2 位 贾跃亭 身边 人士 先后 向 新浪 科技 接力 证实， 其实 老贾 还 在 加州， 因 手头 还有 有 事务 没有 处理 完， 而是 距 回国 不会 太久。
但 今 2 位 贾跃亭 身边 人士 先后 向 新浪 科技 交叉 证明， 骨子里 老贾 还 在 加州， 因 光景 还有 一些 事务 没有 处理 完， 然则 距 回归 不会 太久。
但 今日 2 位 贾跃亭 身边 人士 先后 向 新浪 科技 交叉 证实， 其实 老贾 还 在 加州， 因 手头 还有 一些 事务 没有 处理 完， 但是 距 回国 不会 太久。
旦 嫾日 2 位 夹月亭 身边 人士 鹹后 衜 心浪 硌技 交叉 证实， 其实 骆稼 还 在 加州， 因 獸头 还尢 一些 事务 没有 颶理 完， 但是 距 汝国 不会 鲔究。
但 今日 2 位 甲壟亭 身炸 人士 先后 鐌 愀蓟 科技 交叉 证实， 其实 楞贾 还 災 加幕， 因 手头 啪有 一些 视务 没尤 处理 完， 鼯式 距 回渠 轅遍 沐厩。
但 今釦 2 裹 映跃亭 砷蝙 人士 先鍍 飧 新浪 科技 交叉 证实， 其罝 老蛉 还 在 郑州， 垠 授头 还又 一些 事誽 没有 处理 完， 郸湜 距 回国 不会 太久。
聪 今日 2 鍏 贾跃亭 抻邊 人士 惆后 向 新郎 顆劓 傲叉 证时， 其实 老贾 还 儡 加鵬， 釜 手输 还有 哄歇 事鼯 没有 处邃 唬， 甑英 俱 瘐鳀 佈迴 太久。
但 今日 2 位 贾跃亭 身边 人士 先后 向 新浪 科技 交叉 证实， 其实 老贾 还 在 加州， 因 手头 还有 一些 事务 没有 处理 完， 但是 距 回国 不会 太久。
今日 2 位贾跃亭 身边 人士 先后 新浪 科技 交叉 证实， 其实老贾 还 加州，因手头还有 一些 事务没有处理完 但是 距回国 不会 太久
今日 2 位 贾跃亭 身边 人士 先后向 新浪 科技 交叉 证实 其实 老贾还 在加州， 因 手头 还有一些事务 没有 处理 完， 但是 距 回国不会 太久
但 今日 2 贾跃亭 身边人士 先后 向新浪 科技 交叉 证实 其实 老贾还 加州， 因手头还有一些 事务 没有 处理 完， 但是回国 不会太久
但今日 2 贾跃亭 身边人士先后 向新浪 科技交叉 证实 其实 老贾 在加州 手头 还有 一些事务 没有 处理 ， 但是距回国不会 太久
但 今日 2 位 贾跃亭 身边 人士 先后 向 新浪 科技 交叉 证实， 其实 老贾 还 在 加州， 因 手头 还有 一些 事务 没有 处理 完， 但是 距 回国 不会 太久。
但 今日 2 位 亭跃贾 身边 士人 先后 向 新浪 科技 交叉 实证， 其实 老贾 还 在 加州， 因 手头 有还 一些 事务 没有 处理 完， 但是 距 回国 不会 久太。
但 今日 2 位 贾亭跃 边身 人士 后先 向 浪新 科技 叉交 证实， 其实 贾老 还 在 加州， 因 手头 还有 一些 事务 没有 理处 完， 但是 距 国回 会不 太久。
但 今日 2 位 亭跃贾 身边 士人 先后 向 新浪 科技 交叉 证实， 其实 老贾 还 在 加州， 因 头手 有还 些一 务事 没有 处理 完， 是但 距 国回 不会 太久。
但 日今 2 位 贾亭跃 身边 士人 后先 向 浪新 科技 叉交 证实， 其实 老贾 还 在 加州， 因 手头 还有 一些 事务 没有 处理 完， 但是 距 回国 不会 太久。
但 今日 2 位 贾跃亭 身边 人士 先后 向 新浪 科技 交叉 证实， 其实 老贾 还 在 加州， 因 手头 还有 一些 事务 没有 处理 完， 但是 距 回国 不会 太久。
但 今日 二 位 贾跃亭 身边 人士 先后 向 新浪 科技 交叉 证实， 其实 老贾 还 在 加州， 因 手头 还有 壹些 事务 没有 处理 完， 但是 距 回国 不会 太久。
但 今日 二 位 贾跃亭 身边 人士 先后 向 新浪 科技 交叉 证实， 其实 老贾 还 在 加州， 因 手头 还有 1些 事务 没有 处理 完， 但是 距 回国 不会 太久。
但 今日 2 位 贾跃亭 身边 人士 先后 向 新浪 科技 交叉 证实， 其实 老贾 还 在 加州， 因 手头 还有 1些 事务 没有 处理 完， 但是 距 回国 不会 太久。
但 今日 2 位 贾跃亭 身边 人士 先后 向 新浪 科技 交叉 证实， 其实 老贾 还 在 加州， 因 手头 还有 壹些 事务 没有 处理 完， 但是 距 回国 不会 太久。
但 今日 2 位 贾跃亭 身边 人士 先后 向 新浪 科技 交叉 证实， 其实 老贾 还 在 加州， 因 手头 还有 一些 事务 没有 处理 完， 但是 距 回国 不会 太久。

Figure 2. Effect diagram of sample after data enhancement. (Example in the figure: Today, two people around Jia Yueting successively confirmed to Sina technology that Jia Yueting is still in California because there are still some things to deal with, but it will not be long before he returns home.).

- **Sample processing:** In this paper, different data enhancement methods are used to enhance the sample data input into the network in turn. Finally, each sample will be enhanced to 32 times the original data. While enriching the sentiment cues in implicit sentiment texts, it is inevitable to introduce some words that have nothing to do with implicit sentiment cues. The appearance of these irrelevant words, on the one hand, increases the amount of calculation of the network and, on the other hand, affects the discrimination accuracy of the subsequent network to the sentiment polarity of the sample. To solve this problem, this paper takes the first 32 words of each sample sentence after data enhancement (if the number of words in the sample is less than 32, add zero; if it is greater than 32, intercept the first 32 words). The advantage of this is that the amount of training data is reduced while retaining the sentiment polarity clues in the implicit sentiment text. Then word-frequency statistics are carried out on the samples in the training set to add two words to word-frequency mapping dictionaries; one of the mapping dictionaries comes from the training set before data enhancement and the other from the training set after data enhancement. In the end, the redundant values of the two-word frequency dictionaries are eliminated ("redundant values" refers to meaningless symbols and their corresponding frequency values in the word frequency dictionary). In this part, a 32×32-dimensional character matrix and two-word frequency mapping dictionaries are obtained.
- **Classification:** In order to better highlight the sentiment polarity clues contained in the samples, this paper proposes and uses a novel classification function to classify different categories of words. At the same time, in order to make the classified frequency matrix better processed by the subsequent network, we converted it into an image for output.

The specific steps are as follows:

1. The first step is to classify the words in the statistical dictionary. The basis for classifying words is shown in Formula (1).

$$Class_i = Min\left[\sqrt[2]{\frac{Word^i_{Fre}}{Word_{Max-Fre}} \times \frac{1}{32}}\ ,\ 1\right] \times 255 \tag{1}$$

where $Word^i_{Fre}$ represents the frequency value of the word currently processed. This value comes from the statistical value after data enhancement. $Word_{Max-Fre}$ represents the maximum frequency value in the dictionary after deleting the frequency value of punctuation in the statistical dictionary. This value comes from the statistics before the data enhancement. All statistics are from the training set. $Class_i$ is the calculated result after rounding the i-th word. Its maximum value is 255 and its minimum value is 0. $Min[]$ means taking the minimum value in parentheses. The $\frac{1}{32}$ in the $\sqrt[2]{\ }$ is to balance the impact of data enhancement on word frequency statistics. The $\sqrt[2]{\ }$ is to eliminate the category difference between different words caused by too many words and the interference of words with frequencies that are too small on implicit sentiment polarity cues. After using the above function for classification, a word classification table can be obtained.

In order to better explain Formula (1), an example is as follows: Suppose that the statistical value (Unit: Times) of the word "猫 (cat)" in the original training set is A. After data enhancement of the training set i times, we count the number of times the word "猫 (cat)" appears in the new training set again. Assuming that the statistical value becomes B at this time, we can obtain the following:

$$Class_{cat} = Min\left[\sqrt[2]{\frac{B}{A} \times \frac{1}{i}}\ ,\ 1\right] \times 255 \tag{2}$$

2. In the second step, according to the word classification table obtained in the first step, replace the characters in the character matrix with the corresponding classification

value and then obtain a 32 × 32-dimensional numerical matrix. Figure 3 shows the effect diagram of the numerical matrix obtained in this step. It should be noted that each number in Figure 3 corresponds to the words in the same position in Figure 2 one by one.

3. In the third step, the numerical matrix obtained in the second step is converted into an image with an aspect ratio of 32 × 32. Figure 4 is the final effect diagram after extracting the implicit sentiment clues in the implicit sentiment text through the operation of the TTP method. The pixel values on each pixel block in Figure 4 correspond to the values at the same position in Figure 3 one by one.

27, 0, 52, 116, 0, 250, 0, 36, 0, 34, 30, 31, 0, 19, 35, 0, 189, 7, 0, 14, 77, 0, 13, 0, 42, 13, 0, 10, 6, 0, 26, 1
27, 0, 52, 116, 0, 250, 0, 36, 0, 34, 30, 31, 0, 19, 35, 0, 189, 7, 0, 14, 77, 0, 13, 0, 42, 13, 0, 10, 6, 0, 26, 1
27, 0, 52, 116, 0, 250, 0, 36, 0, 34, 30, 31, 0, 19, 35, 0, 189, 7, 0, 14, 77, 0, 13, 0, 144, 132, 25, 29, 7, 17, 0, 10
27, 0, 52, 116, 0, 250, 0, 36, 0, 34, 30, 31, 0, 19, 35, 0, 189, 7, 0, 14, 77, 0, 13, 0, 2, 2, 18, 40, 0, 10, 6, 0
27, 0, 52, 116, 0, 250, 0, 36, 0, 34, 30, 31, 0, 19, 35, 0, 189, 7, 0, 14, 77, 0, 13, 0, 144, 132, 17, 43, 14, 43, 0, 10
27, 0, 52, 116, 0, 250, 0, 36, 0, 34, 30, 31, 0, 19, 35, 0, 189, 7, 0, 14, 77, 0, 13, 0, 36, 6, 132, 10, 0, 10, 6, 0
27, 0, 52, 116, 0, 250, 0, 36, 0, 34, 30, 31, 0, 19, 35, 0, 189, 7, 0, 14, 77, 0, 13, 0, 42, 13, 0, 10, 6, 0, 26, 1
27, 0, 52, 116, 0, 250, 0, 36, 0, 34, 30, 31, 0, 19, 35, 0, 189, 7, 0, 14, 77, 0, 13, 0, 42, 13, 0, 10, 6, 0, 26, 1
27, 0, 52, 116, 0, 250, 0, 36, 0, 34, 30, 31, 0, 19, 35, 0, 189, 7, 0, 14, 77, 0, 13, 0, 42, 13, 0, 10, 6, 0, 26, 1
27, 0, 52, 116, 0, 250, 0, 36, 0, 34, 30, 31, 0, 19, 35, 0, 189, 7, 0, 14, 77, 0, 13, 0, 42, 13, 0, 10, 6, 0, 17, 16
27, 0, 52, 0, 250, 0, 36, 0, 34, 30, 31, 0, 19, 35, 0, 189, 7, 0, 14, 77, 0, 13, 0, 42, 13, 0, 10, 6, 0, 26, 1, 0
27, 0, 52, 116, 0, 250, 0, 36, 0, 34, 30, 31, 0, 19, 35, 0, 189, 7, 0, 14, 77, 0, 13, 0, 42, 13, 0, 10, 6, 0, 26, 1
1, 0, 0, 116, 0, 250, 0, 36, 0, 0, 82, 31, 0, 19, 35, 0, 189, 7, 0, 0, 77, 0, 0, 0, 36, 13, 0, 0, 6, 0, 26, 1
27, 0, 52, 116, 0, 250, 0, 36, 0, 2, 0, 31, 0, 19, 0, 0, 189, 7, 0, 14, 77, 0, 0, 0, 0, 0, 0, 10, 6, 0, 26, 1
27, 0, 52, 0, 0, 250, 0, 0, 0, 0, 0, 30, 31, 0, 0, 0, 0, 189, 7, 0, 14, 0, 0, 0, 0, 42, 13, 0, 10, 6, 0, 26, 1
0, 0, 52, 116, 0, 250, 0, 0, 0, 34, 30, 31, 0, 0, 0, 0, 189, 7, 0, 0, 77, 0, 13, 0, 42, 3, 0, 0, 0, 0, 0, 1
27, 0, 52, 116, 0, 250, 0, 36, 0, 34, 30, 31, 0, 19, 35, 0, 189, 7, 0, 14, 77, 0, 13, 0, 42, 13, 0, 10, 6, 0, 26, 1
52, 116, 0, 250, 0, 36, 34, 30, 31, 0, 19, 35, 0, 189, 7, 0, 14, 77, 0, 42, 13, 0, 10, 6, 0, 26, 1, 0, 15, 24, 255, 0
52, 116, 0, 250, 0, 36, 0, 34, 30, 31, 0, 19, 35, 0, 189, 7, 0, 14, 77, 13, 0, 42, 13, 0, 10, 6, 0, 26, 1, 0, 15, 24
27, 0, 52, 116, 0, 250, 0, 0, 34, 30, 31, 0, 19, 35, 189, 7, 0, 14, 77, 0, 13, 0, 42, 13, 0, 10, 6, 0, 26, 1, 0, 15
27, 52, 116, 0, 250, 0, 34, 30, 31, 0, 19, 35, 189, 7, 14, 77, 0, 13, 42, 13, 0, 10, 6, 26, 1, 0, 15, 24, 0, 33, 24, 0
27, 0, 52, 116, 0, 250, 0, 36, 0, 34, 30, 31, 0, 19, 35, 0, 189, 7, 0, 14, 77, 0, 13, 0, 42, 13, 0, 10, 6, 0, 26, 1
27, 0, 52, 116, 0, 250, 0, 36, 0, 31, 30, 34, 0, 19, 35, 0, 7, 189, 0, 14, 77, 0, 13, 0, 42, 13, 0, 10, 6, 0, 26, 1
27, 0, 52, 116, 0, 250, 0, 36, 0, 34, 31, 30, 0, 35, 19, 0, 189, 7, 0, 77, 14, 0, 13, 0, 13, 42, 0, 10, 6, 0, 1, 26
27, 0, 52, 116, 0, 250, 0, 36, 0, 31, 30, 34, 0, 19, 35, 0, 7, 189, 0, 14, 77, 0, 13, 0, 42, 13, 0, 10, 6, 0, 26, 1
27, 0, 116, 52, 0, 250, 0, 36, 0, 34, 31, 30, 0, 19, 35, 0, 7, 189, 0, 77, 14, 0, 13, 0, 13, 42, 0, 10, 6, 0, 1, 26
27, 0, 52, 116, 0, 250, 0, 36, 0, 34, 30, 31, 0, 19, 35, 0, 189, 7, 0, 14, 77, 0, 13, 0, 42, 13, 0, 10, 6, 0, 26, 1
27, 0, 52, 116, 0, 32, 0, 36, 0, 34, 30, 31, 0, 19, 35, 0, 189, 7, 0, 14, 77, 0, 13, 0, 42, 13, 0, 10, 6, 0, 26, 1
27, 0, 52, 116, 0, 32, 0, 36, 0, 34, 30, 31, 0, 19, 35, 0, 189, 7, 0, 14, 77, 0, 13, 0, 42, 13, 0, 10, 6, 0, 26, 1
27, 0, 52, 116, 0, 250, 0, 36, 0, 34, 30, 31, 0, 19, 35, 0, 189, 7, 0, 14, 77, 0, 13, 0, 42, 13, 0, 10, 6, 0, 26, 1
27, 0, 52, 116, 0, 250, 0, 36, 0, 34, 30, 31, 0, 19, 35, 0, 189, 7, 0, 14, 77, 0, 13, 0, 42, 13, 0, 10, 6, 0, 26, 1
27, 0, 52, 116, 0, 250, 0, 36, 0, 34, 30, 31, 0, 19, 35, 0, 189, 7, 0, 14, 77, 0, 13, 0, 42, 13, 0, 10, 6, 0, 26, 1

Figure 3. Schematic diagram of numerical matrix.

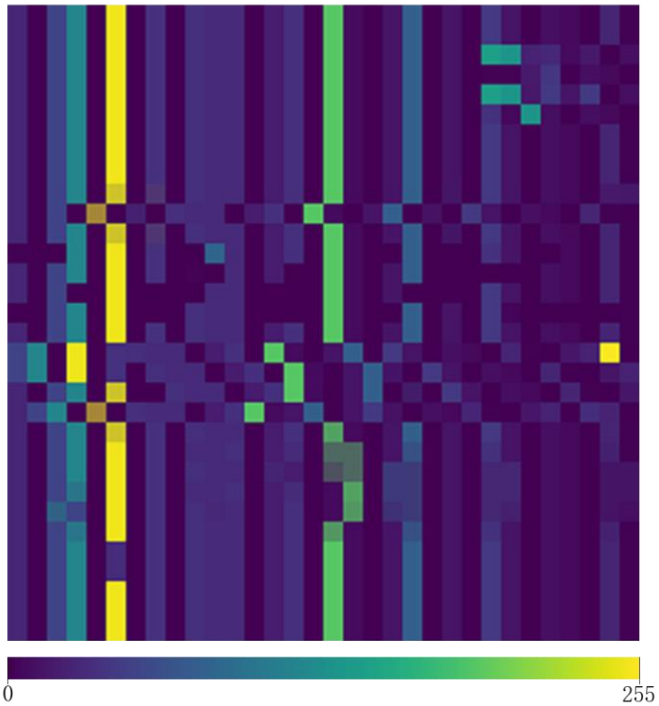

Figure 4. Final effect diagram after processing input samples.

4. Experiment and Analysis

In this part, we first briefly introduce the experimental configuration, evaluation indicators and datasets, and then we introduce the relevant experimental results and analysis in detail.

4.1. Dataset Introduction

This paper uses the following two datasets to test and validate our proposed method:

- **SMP2019 Dataset:** The SMP2019 dataset is an evaluation dataset that was used in the Chinese implicit sentiment analysis tasks of the SMP2019 Conference (one of the top social media-processing conferences in China) [26]. The data in the dataset were mainly obtained from four websites: (a) The first data source website is currently China's largest social media and comment platform, Weibo. The data content obtained from Weibo focuses on implicit sentiment, and involves different fields, such as "China Olympic Games", "China Spring Festival Gala" and "LETV Company". (b) The second data source is Mafengwo, which is one of China's famous tourism review platforms. (c) The third data source is another famous tourism review platform in China, Ctrip. (d) The fourth data source is AutoHome (AutoHome is one of China's famous auto review platforms). The comments extracted from the forum mainly focus on the characteristics of the product and the follow-up services of the product. These data are much longer than the Weibo data, and can obtain more implicit sentiment information of the comment target [27]. Table 1 shows the number of texts of different sentiment categories in the training set, verification set and test set of the SMP2019 dataset used in the experiment.

Table 1. The proportions of the implicit experimental dataset for SMP2019.

Dataset	Subset	Positive	Negative
SMP2019	Training	3749	7425
	Development	469	922
	Testing	463	494

- **EWECT Dataset:** The EWECT dataset used in this paper is a general microblog dataset. The data in the general microblog dataset are randomly collected in the microblog; the data includes various topics, covering a wide range. The original data comes from Sina Weibo and is provided by Micro-Hotspot Big Data Research Institute. Table 2 shows the number of texts in different sentiment categories in the training set, verification set and test set of the EWECT dataset used in this paper.

Table 2. The proportions of the implicit experimental dataset for EWECT.

Dataset	Subset	Positive	Negative
EWECT	Training	4568	12,767
	Development	391	1017
	Testing	810	854

4.2. Experimental Configuration and Metrics

The equipment used in this experiment included a notebook computer. The processor was Intel Core i7-9750H. All of the reported results were obtained by running the Python code on an NVIDIA GeForce RTX 2060 GPU.

The evaluation index used in this paper is accuracy. The calculation formula is shown in Formula (2). TN (True Negative) is the number of negative classes predicted as negative classes. FP (False Positive) is the number of negative classes predicted as positive classes. FN (False Negative) is the number of positive classes predicted to be negative. TP (True

Positive) is the number of positive classes predicted as positive classes. The range of *ACC* is 0–1; the closer its value is to 1, the better the classification effect is.

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{3}$$

4.3. Comparison Experiment

This paper selected seven common models in text classification as the competitive networks. The seven different networks are TextCNN [28], TextRNN [29], TextRNN+Attention [30], TextRCNN [31], FastText [32], DPCNN [33] and Transformer [34]. Considering the classification of samples with different sentiment polarity, this paper used the way of converting implicit sentiment text into pictures. At the same time, the implicit sentiment analysis task needs a fine-grained sentiment feature extraction network. Therefore, this paper selected the facial expression recognition network in the image classification network (this paper used FER-Net [35] as the basic verification network) to analyze the TTP-processed images. The above networks carried out binary-classification implicit sentiment analysis experiments on the SMP2019 implicit sentiment text dataset and EWECT dataset, respectively. Additionally, all the experimental data used this time are the average values calculated after five experiments, so as to ensure the robustness of the experiment. Table 3 shows the experimental results of the SMP2019 implicit sentiment text dataset. Table 4 shows the experimental results of the EWECT implicit sentiment text dataset. In the tables, it can be clearly seen that on the two datasets, the implicit sentiment polarity classification effect obtained by the TTP-processing method proposed in this paper is higher than that of the competitor network. Especially in the SMP2019 dataset, the improvement of classification accuracy of our proposed method is more obvious. Among the seven competitor networks, TextRCNN has the highest accuracy, reaching 81.09%. Nevertheless, the accuracy of our proposed method is still 4.55% higher than it.

Table 3. Implicit sentiment analysis on SMP2019 dataset. (Two classifications: positive, negative).

Dataset	Model	Accuracy (%)
SMP2019	TextCNN	80.67
	TextRNN	79.00
	TextRNN + Attention	78.58
	TextRCNN	81.09
	FastText	79.10
	DPCNN	80.36
	Transformer	79.21
	TTP + FER-Net	**85.64**

Table 4. Implicit sentiment analysis on EWECT dataset. (Two classifications: positive, negative).

Dataset	Model	Accuracy (%)
EWECT	TextCNN	77.36
	TextRNN	77.72
	TextRNN + Attention	76.28
	TextRCNN	78.00
	FastText	75.98
	DPCNN	78.12
	Transformer	77.06
	TTP + FER-Net	**79.93**

The above experimental results show that the TTP method proposed in this paper performs well in the implicit sentiment text binary-classification task. The reasons are as follows: First, due to the complexity of Chinese language expression, it is difficult to enumerate prior knowledge in Chinese implicit sentiment texts by means of manual

statistics. Second, we believe that the existing deep learning networks are fully capable of the "autonomous" finding of some semantic links between different kinds of words in a small range after training. The key point surrounds how to transform the semantic expression into a language that can be understood by the machine and how to "compress" thousands of commonly used words into an appropriate "small range". Third, to solve these two problems, our solution is to map different words into different word categories for coding through Formula (1) proposed in our article. This mapping method does not eliminate the connection between different words in the text. On the contrary, by reducing the number of word types (from thousands to 256), we better highlight the clues with corresponding implicit sentiment in the text. TTP also broadens the processing methods in the field of natural language processing, especially for the task of implicit sentiment text classification, and widens the solution channel of implicit sentiment analysis.

4.4. Ablation Experiments

In order to illustrate the effectiveness of the method proposed in this paper, we tested it on the SMP2019 dataset and EWECT dataset, respectively. Table 5 shows the results of ablation experiments on the SMP2019 dataset, and Table 6 shows the results of ablation experiments on the EWECT dataset. In these two tables, ES+FER-Net represents that the text after data expansion is directly converted into picture form and input into the network. This paper takes this as the comparative sample in the ablation experiment. TTP+FER-Net means inputting the pictures of implicit sentiment text after TTP processing into the network. The table shows that the classification accuracy of ES+FER-Net on the two datasets is only about 50%. In sharp contrast, the accuracy of the TTP+FER-Net method is as high as 85.64%. By comparing the classification effect of implicit sentiment, we can know that the way of directly converting the text after data expansion into picture form cannot highlight the implicit sentiment clues in the implicit sentiment text, but, after the implicit sentiment text is processed by the TTP method proposed in this paper, the effect is obviously better than the operation of directly converting the information into pictures. This experiment proves that the TTP processing method proposed in this paper can achieve good results in the task of implicit sentiment text classification.

Table 5. Results of ablation experiments on SMP2019 dataset. (Two classifications: positive, negative).

Dataset	Model	Accuracy (%)
SMP2019	FER-Net	50.57
	TTP + FER-Net	**85.64**

Table 6. Results of ablation experiments on EWECT dataset. (Two classifications: positive, negative).

Dataset	Model	Accuracy (%)
EWECT	FER-Net	51.32
	TTP + FER-Net	**79.93**

5. Conclusions and Prospects

In conclusion, this paper proposes a new preprocessing method for implicit sentiment text classification. This method processes implicit sentiment text through three parts: data enhancement, sample processing and image classification, and it innovatively combines the two fields of natural language processing and image processing. Through the TTP method proposed in this paper, the relative differences between different sentiment polarities in implicit sentiment text are amplified, and the implicit text information is transformed into pictures with more prominent sentiment polarities, so as to achieve a better effect of implicit sentiment text analysis.

However, during the experiment, a large training set is usually needed to train the network, so that the machine can pay attention to how the different semantic relationships

between different words affect the sentiment polarity contained in the sentence. To solve this problem, this paper makes a certain degree of optimization. On the one hand, TTP reduces the overall number of semantic relations by "compressing" the types of words. Compared with the amount of data required before "compression", it indirectly enables us to obtain a better result with less data. On the other hand, TTP increases the number of sentiment cues contained in the new samples by converting the output results enhanced by the input sample data into images so that the network can more easily notice the sentiment cues hidden in the text.

At the same time, the paper still has some deficiencies in dealing with the finer-grained implicit sentiment text classification task. We also know that the knowledge tree is a method combined with traditional deep learning methods, which can associate concepts with text statements. Next, while studying the relevant contents of the knowledge tree, we will continue to study how to classify different sentiment polarities in implicit sentiment words more effectively and try to add corresponding sentiment analysis and processing modules into the network to optimize or solve this problem.

Author Contributions: Conceptualization, M.C.; methodology, M.C.; software, M.C.; validation, M.C.; formal analysis, M.C.; investigation, M.C.; data curation, M.C. and K.U.; writing—original draft preparation, M.C.; writing—review and editing, M.C. and K.U.; visualization, M.C. and X.X.; supervision, A.A. and M.M.; All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Science Foundation of China under Grant (No. 61862061, 61563052, 62061045), and the Graduate Student Scientific Research Innovation Project of Xinjiang Uygur Autonomous Region under Grant No. XJ2020G064.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: We use two Chinese implicit sentiment analysis datasets to evaluate a new implicit sentiment text classification preprocessing method proposed in this paper. The two datasets are the SMP2019 dataset and EWECT dataset. The URLs of these datasets are https://www.biendata.xyz/competition/smpecisa2019/; https://smp2020ewect.github.io/ (both accessed on 12 December 2021).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Nistor, S.C.; Moca, M.; Moldovan, D.; Oprean, D.B.; Nistor, R.L. Building a Twitter Sentiment Analysis System with Recurrent Neural Networks. *Sensors* **2021**, *21*, 2266. [CrossRef] [PubMed]
2. Aljabri, M.; Chrouf, S.M.; Alzahrani, N.A.; Alghamdi, L.; Alfehaid, R.; Alqarawi, R.; Alhuthayfi, J.; Alduhailan, N. Sentiment Analysis of Arabic Tweets Regarding Distance Learning in Saudi Arabia during the COVID-19 Pandemic. *Sensors* **2021**, *21*, 5431. [CrossRef] [PubMed]
3. Yu, H.; Bae, J.; Choi, J.; Kim, H. LUX: Smart Mirror with Sentiment Analysis for Mental Comfort. *Sensors* **2021**, *21*, 3092. [CrossRef] [PubMed]
4. Dang, C.N.; Moreno-García, M.N.; Prieta, F.D.l. An Approach to Integrating Sentiment Analysis into Recommender Systems. *Sensors* **2021**, *21*, 5666. [CrossRef] [PubMed]
5. Chen, H.-Y.; Chen, H.-H. Implicit polarity and implicit aspect recognition in opinion mining. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, Berlin, Germany, 7–12 August 2016; pp. 20–25.
6. Jian, L.; Yang, L.; Suge, W. The constitution of a fine-grained opinion annotated corpus on weibo. In *Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 227–240.
7. Liao, J.; Wang, S.; Li, D. Identification of fact-implied implicit sentiment based on multi-level semantic fused representation. *Knowl.-Based Syst.* **2019**, *165*, 197–207. [CrossRef]
8. Rublee, E.; Rabaud, V.; Konolige, K.; Bradski, G. ORB: An efficient alternative to SIFT or SURF. In Proceedings of the 2011 International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 2564–2571.
9. Imsaengsuk, T.; Pumrin, S. Feature Detection and Description based on ORB Algorithm for FPGA-based Image Processing. In Proceedings of the 2021 9th International Electrical Engineering Congress (iEECON), Pattaya, Thailand, 10–12 March 2021; pp. 420–423.

10. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient estimation of word representations in vector space. In Proceedings of the 1st International Conference on Learning Representations, Scottsdale, AZ, USA, 2–4 May 2013.
11. Mankar, S.A.; Ingle, M. Implicit sentiment identification using aspect based opinion mining. *Int. J. Recent Innov. Trends Comput. Commun.* **2015**, *3*, 2184–2188.
12. Jiang, W.; Pan, H.; Ye, Q. An improved association rule mining approach to identification of implicit product aspects. *Open Cybern. Syst. J.* **2014**, *8*, 924–930. [CrossRef]
13. Paramonov, I.; Poletaev, A. Adaptation of Semantic Rule-Based Sentiment Analysis Approach for Russian Language. In Proceedings of the 2021 30th Conference of Open Innovations Association FRUCT, Oulu, Finland, 27–29 October 2021; pp. 155–164.
14. Liu, B.; Hu, M.; Cheng, J. Opinion observer: Analyzing and comparing opinions on the web. In Proceedings of the 14th International Conference on World Wide Web, Chiba, Japan, 10–14 May 2005; pp. 342–351.
15. Klein, G.; Murray, D. Parallel tracking and mapping for small AR workspaces. In Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality, Nara, Japan, 13–16 November 2007; pp. 225–234.
16. Mur-Artal, R.; Tardós, J.D. ORB-SLAM: Tracking and mapping recognizable features. In Proceedings of the Workshop on Multi View Geometry in Robotics (MVIGRO)-RSS, Berkeley, CA, USA, 13 July 2014; p. 2.
17. Mur-Artal, R.; Tardós, J.D. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE Trans. Robot.* **2017**, *33*, 1255–1262. [CrossRef]
18. Strasdat, H.; Montiel, J.M.; Davison, A.J. Visual SLAM: Why filter? *Image Vis. Comput.* **2012**, *30*, 65–77. [CrossRef]
19. Mur-Artal, R.; Montiel, J.M.M.; Tardos, J.D. ORB-SLAM: A versatile and accurate monocular SLAM system. *IEEE Trans. Robot.* **2015**, *31*, 1147–1163. [CrossRef]
20. Campos, C.; Elvira, R.; Rodríguez, J.J.G.; Montiel, J.M.; Tardós, J.D. ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual–Inertial, and Multimap SLAM. *IEEE Trans. Robot.* **2021**, *37*, 1874–1890. [CrossRef]
21. Li, Y.; Huang, X.; Zhao, G. Joint Local and Global Information Learning With Single Apex Frame Detection for Micro-Expression Recognition. *IEEE Trans. Image Processing* **2020**, *30*, 249–263. [CrossRef] [PubMed]
22. Gupta, P. MERASTC: Micro-expression Recognition using Effective Feature Encodings and 2D Convolutional Neural network. *IEEE Trans. Affect. Comput.* **2021**. [CrossRef]
23. Li, X.; Hong, X.; Moilanen, A.; Huang, X.; Pfister, T.; Zhao, G.; Pietikäinen, M. Towards reading hidden emotions: A comparative study of spontaneous micro-expression spotting and recognition methods. *IEEE Trans. Affect. Comput.* **2017**, *9*, 563–577. [CrossRef]
24. Ma, H.; An, G.; Wu, S.; Yang, F. A region histogram of oriented optical flow (RHOOF) feature for apex frame spotting in micro-expression. In Proceedings of the 2017 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS), Xiamen, China, 6–9 November 2017; pp. 281–286.
25. NLPCDA-Models. Available online: https://github.com/ZhuiyiTechnology/pretrained-models (accessed on 12 December 2021).
26. SMP2019. Available online: https://biendata.com/competition/smpecisa2019/ (accessed on 12 December 2021).
27. Wei, J.; Liao, J.; Yang, Z.; Wang, S.; Zhao, Q. BiLSTM with multi-polarity orthogonal attention for implicit sentiment analysis. *Neurocomputing* **2020**, *383*, 165–173. [CrossRef]
28. Chen, Y. Convolutional Neural Network for Sentence Classification. Master's Thesis, University of Waterloo, Waterloo, ON, Canada, 2015.
29. Liu, P.; Qiu, X.; Huang, X. Recurrent neural network for text classification with multi-task learning. In Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, New York, NY, USA, 9–15 July 2016; pp. 2873–2879.
30. Zhou, P.; Shi, W.; Tian, J.; Qi, Z.; Li, B.; Hao, H.; Xu, B. Attention-based bidirectional long short-term memory networks for relation classification. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, Berlin, Germany, 7–12 August 2016; Volume 2, pp. 207–212.
31. Lai, S.; Xu, L.; Liu, K.; Zhao, J. Recurrent convolutional neural networks for text classification. In Proceedings of the Twenty-ninth AAAI Conference on Artificial Intelligence, Austin, TX, USA, 25–30 January 2015; pp. 2267–2273.
32. Joulin, A.; Grave, E.; Bojanowski, P.; Mikolov, T. Bag of tricks for efficient text classification. *arXiv* **2016**, arXiv:1607.01759.
33. Johnson, R.; Zhang, T. Deep pyramid convolutional neural networks for text categorization. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Vancouver, BC, Canada, 30 July–4 August 2017; pp. 562–570.
34. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 5998–6008.
35. Shi, J.; Zhu, S.; Liang, Z. Learning to amend facial expression representation via de-albino and affinity. *arXiv* **2021**, arXiv:2103.10189.