*1.* *Discuss the validity of the hyper-parameters shown in Table 4 of paper.*

In order to find a suitable sampling rate for subsequent experiments. We randomly sampled 10% of the 62275 annotations as the training set, which is 6227 annotations. 10% was chosen because it is approximately equal to the average number of comments for 10 projects, it can be seen from Table 1 below. In addition, we also selected the remaining 1% as the test set, i.e., 622 annotations, to evaluate models trained with different sampling rates.

Table 1. Statistics of the comments in the 10 projects.

|  |  |  |  |
| --- | --- | --- | --- |
| Projects | Comment | #SATD | % of SATD |
| Ant | 4,098 | 131 | 0.60 |
| ArgoUML | 9,452 | 1,413 | 2.08 |
| Columba | 6,478 | 204 | 0.60 |
| EMF | 4,390 | 104 | 0.41 |
| Hibernate | 2,968 | 472 | 4.05 |
| JEdit | 10,322 | 256 | 1.50 |
| JFreeChart | 4,408 | 209 | 0.89 |
| JMeter | 8,057 | 374 | 1.86 |
| JRuby | 4,897 | 622 | 5.57 |
| SQuirrel | 7,215 | 286 | 1.04 |
| Avg. | 6,228 | 407 | 0.65 |
| Total | 62,275 | 4,071 | 6.54 |

When experimenting with one of these hyperparameters, we employ an approach called control variables, which is when one of the variables is evaluated while the others are held constant. Below are the experimental results for different hyperparameters.

For word embedding dimensions, we chose three appropriate dimension sizes: 100, 200, and 300. With each dimension size, we first use SeqGAN-based oversampling to bridge the extreme gap between SATD and non-SATD comment texts from nine projects to train a CNNGRU model. Then, we predict comments in the rest project with the pre-trained model, so as to calculate Precision, Recall and F-Measure. Table 1 shows the results. It can be seen that the larger the word embedding dimension, the better performance the model achieves.

Table 1. Average results of five technical debts at different dimensions.

|  |  |  |  |
| --- | --- | --- | --- |
| Dimensions | Precision | Recall | F-measure |
| 100 | 0.563 | 0.525 | 0.543 |
| 200 | 0.652 | 0.530 | 0.585 |
| 300 | 0.687 | 0.556 | 0.615 |

For the number of filters, we chose three numbers: 64, 128, 192. Table 2 shows the results. It can be seen that it can be seen that the more filters the model uses, the higher precision recall, and F-measure it can achieve. However, we found that increasing the number of filters beyond 128 resulted in only a slight performance gain. The more filters a model uses, the more parameters it has, which increases its complexity.

Table 2. Average results of five technical debts at different number of filters.

|  |  |  |  |
| --- | --- | --- | --- |
| Filters | Precision | Recall | F-measure |
| 64 | 0.702 | 0.533 | 0.606 |
| 128 | 0.695 | 0.567 | 0.625 |
| 192 | 0.709 | 0.563 | 0.628 |

For the window size of filters, we consider these five window sizes, because according to pattern-based methods, the lengths of SATD patterns observed by humans are found to mostly fall within the range of 1 to 5.

Table 3. Average results of five technical debts at different number of filters.

|  |  |  |  |
| --- | --- | --- | --- |
| Window size of filters | Precision | Recall | F-measure |
| (1,2,3) | 0.682 | 0.522 | 0.591 |
| (2,3,4) | 0.697 | 0.560 | 0.621 |
| (3,4,5) | 0.705 | 0.573 | 0.632 |
| (1,2,3,4) | 0.681 | 0.551 | 0.609 |
| (2,3,4,5) | 0.709 | 0.578 | 0.637 |
| (1,2,3,4,5) | 0.713 | 0.585 | 0.643 |

The length of the comment text in the data set is mostly between 0 and 128, as shown in Table 4. If the maximum length is too small, a certain number of comments will be truncated. Although a large maximum length value can be used, it will increase the calculation overhead. According to statistics, the comment length within 128 accounts for 99.88% of all samples. And those greater than 128 only accounted for 0.12%. That is, 0.12% sequences are cropped due to the threshold of 128 words. Also, by convention maxlen is usually chosen to be an exponential multiple of 2 [1]. Therefore, in this paper, we have a uniform length of 128. We set the parameter maxlen to 128 to indicate the length of the comment sentence, which can cut the sentence with more than 128 words in the text comment and keep it at 128. For sentences with less than 128 words, we add them to 128. Use zero padding at the beginning until the comment length reaches the maximum. And Sentence length padding is to maintain consistency by adding zero tensor (zero tensor is a placeholder that does not represent any information) to avoid the existence of empty elements. Adding a zero tensor at the beginning has the same effect as adding at the end. For consistency, this article chooses to padding at the beginning.

[1] Cunha W, Mangaravite V, Gomes C, et al. On the cost-effectiveness of neural and non-neural approaches and representations for text classification: A comprehensive comparative study[J]. Information Processing & Management, 2021, 58(3):102481.

[2] Li P, Li X, Pan H, et al. Text-based indoor place recognition with deep neural network[J]. Neurocomputing, 2019, 390.

Table 4. Sequence length statistics in 10 projects.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Projects | 16 | 32 | 64 | 128 | more than 128 |
| Ant | 3484 | 436 | 156 | 19 | 3 |
| ArgoUML | 7637 | 1248 | 436 | 111 | 20 |
| Columba | 5889 | 477 | 84 | 14 | 4 |
| EMF | 3562 | 567 | 216 | 40 | 5 |
| Hibernate | 2337 | 453 | 139 | 33 | 6 |
| JEdit | 9847 | 343 | 99 | 27 | 6 |
| JFreeChart | 4115 | 247 | 36 | 5 | 5 |
| JMeter | 7232 | 619 | 177 | 22 | 7 |
| JRuby | 4897 | 4242 | 485 | 134 | 30 |
| SQuirrel | 5930 | 864 | 324 | 82 | 15 |
| Total. | 54275 | 5739 | 1801 | 383 | 77 |
| % of different length | 87.14% | 9.21% | 2.89% | 0.61% | 0.12% |

For the dropout, it prevents overfitting and provides a way of approximately combining exponentially many different neural network architectures efficiently [1]. It shows that when the dropout rate is equal to 0.5, the effect is the best, the reason is that when the dropout rate is 0.5, the network structure randomly generated by dropout is the most.

[1] Srivastava N, Hinton G, Krizhevsky A, et al. Dropout: A Simple Way to Prevent Neural Networks from Overfitting[J]. Journal of Machine Learning Research, 2014, 15(1): 1929-1958.

*2. In section 3.2.2, explain the processed of padding.*

**Response:** We set the parameter *maxlen* to 128, which can cut out sentences with more than 128 words in the text comment to keep it at 128. For sentences with less than 128 words, use the *pad\_sequences* function to add it to 128. Use zero-padding at the beginning to the maximum comment length. Figures 3 and 4 show an example of padding to the maximum comment length.

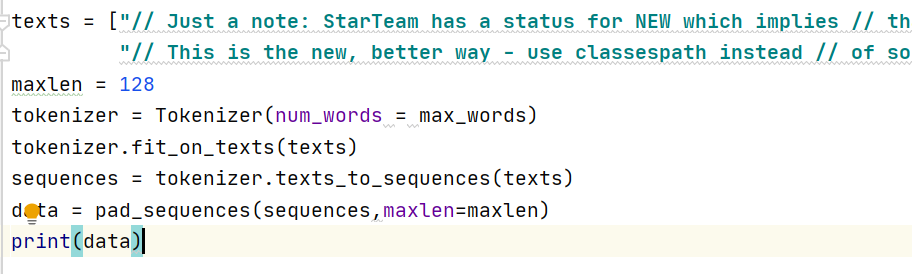


Figure 1. An example of padding to the maximum comment length.

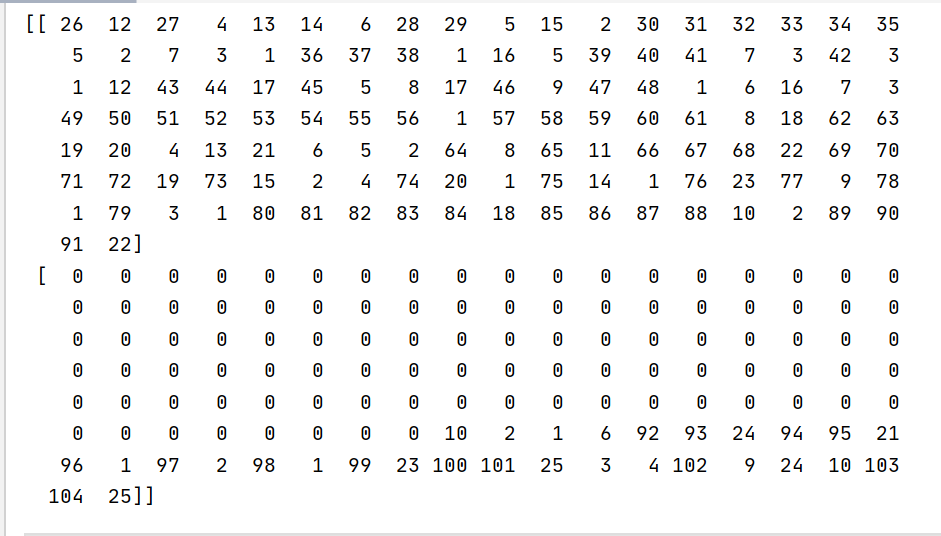


Figure 2. The result of padding to the maximum comment length.

*3.The descriptive statistics of the sequence length.*

**Response:** The distribution of all comment text length are shown in Figure 1. The length of the annotation text in the data set is mostly between 0 and 128, as shown in Table 5. If the maximum length is too small, a certain number of comments will be truncated. Although a large maximum length value can be used, it will increase the calculation overhead. According to statistics, the comment length within 128 accounts for 99.88% of all samples. And those greater than 128 only accounted for 0.12%.

图表, 直方图

描述已自动生成

Figure 1. Distribution of Comment length.

Table 5. Sequence length statistics in 10 projects.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Projects | 16 | 32 | 64 | 128 | more than 128 |
| Ant | 3484 | 436 | 156 | 19 | 3 |
| ArgoUML | 7637 | 1248 | 436 | 111 | 20 |
| Columba | 5889 | 477 | 84 | 14 | 4 |
| EMF | 3562 | 567 | 216 | 40 | 5 |
| Hibernate | 2337 | 453 | 139 | 33 | 6 |
| JEdit | 9847 | 343 | 99 | 27 | 6 |
| JFreeChart | 4115 | 247 | 36 | 5 | 5 |
| JMeter | 7232 | 619 | 177 | 22 | 7 |
| JRuby | 4897 | 4242 | 485 | 134 | 30 |
| SQuirrel | 5930 | 864 | 324 | 82 | 15 |
| Total. | 54275 | 5739 | 1801 | 383 | 77 |
| % of different length | 87.14% | 9.21% | 2.89% | 0.61% | 0.12% |