Comparison with TSE 2017:

In Section 4.2 Baseline Methods, we add another baseline as follows:

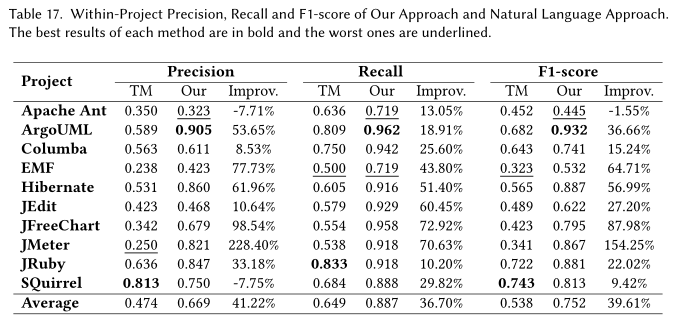
*“Natural Language Processing Based SATD Classification. Maldonado et al. [11] put forward an approach to automatically identify the two most common types of SATD comments (i.e., design debt and requirement debt), which built a maximum entropy classifier based on natural language processing (NLP). Although they used the same dataset and same experiment setting with ours, we cannot directly compare their experiment results with ours, since they separately reported the precision, recall and F1-score of each type of SATD comment. Besides, our dataset also contains other types of SATD comment. In order to make comparison, Huang et al. [10] also follow their work to do basic preprocessing (i.e., stemming and removing punctuation characters) and build the maximum entropy classifier to predict whether a comment contains SATD or not. Note that, our dataset contain more comments from Apache ant and EMF projects and to make our paper self-contained, we follow the approach of language processing based approach to make a series of comparison experiments in RQ5, RQ6.”*

In RQ5, we add some results of within-project precision, recall and F1-score compared our approach with the approach proposed by Maldonado et al. [11], which is as follows:

*“ Table 15 shows the precision, recall and F1-score of our approach and NLP method in the 10 within-project classification experiments. Generally speaking, it is obvious that our approach achieves great improvements in precision, recall as well as F1-score. On average, our approach achieves the 41.22%, 36.70% and 39.61% improvement in precision, recall and F1-score than the NLP method. The Wilcoxon signed-ranked tests show that the improvements in precision, recall and F1-score are all statistically significant at the p-value < 0.05. The Cliff’s delta of F1-score is 0.66 (larger than 0.474), which indicates that our approach significantly improves F1-score over the text mining approach by a substantial margin.*

......

*When compared with NLP approach, for most of the 10 projects, the F1-scores of our approach are higher than that of NLP approach and especially, for JMeter, our approach gets an improvement of 154.25%. While, for Apache Ant, our approach is -1.55% lower than that of NLP approach.*”



In RQ6, we add some results of cross-project precision, recall and F1-score compared our approach with the approach proposed by Maldonado et al. [11], which is as follows:

“*Then, we compared the results of our approach and natural language processing approach (see Table 18). It is easy to find that our approach also performs much better than natural language processing approach. After computation, the improvement in F1-score is statistically significant by Wilcoxon signed-rank test at p-value < 0.05 (which is 0.03). Additionally, the Cliff’s delta (δ) of F1-score is 0.52 ( in the large level), which means that our approach improves F1-score over the text mining approach by a large margin.*

*...*

*Comparing the average F1-score of the ten 9 → 1 experiments with that of the 10 within-project experiments, the text-mining method’s average F1-score improves 10.6% (0.696 versus 0.629), NLP approach’s average F1-score improves 15.99% (0.624 versus 0.538), while our approach’s average F1-score improves 1.86% (0.766 versus 0.752)*”