Several studies have explored sentiment analysis within the context of software engineering (SE), particularly focusing on open-source platforms like GitHub. However, it has been observed that off-the-shelf sentiment analysis tools often perform poorly on SE-specific texts. This is due to the unique language and context used in software development discussions.

* **Guzman et al. (2014)** conducted an early study using SentiStrength to analyze commit comments in GitHub projects and found correlations between sentiment and project variables like team distribution and the day of the week the comment was posted. They found that negative sentiment was more prevalent in Java projects and on Mondays. This study is foundational in applying sentiment analysis to the SE domain, and highlights the need for contextual analysis.
* **Islam and Zibran (2018)** developed SentiStrength-SE to address the shortcomings of general sentiment analysis tools when applied to SE texts. They identified issues such as misclassification of domain-specific words and the presence of code snippets, which are unique to software engineering contexts. This tool, which improves upon the original SentiStrength by incorporating a domain-specific dictionary, contextual information, and preprocessing steps, was shown to outperform SentiStrength.
* **Lin et al. (2018)** investigated the effectiveness of various sentiment analysis tools on software-related datasets, including Stack Overflow discussions, app reviews, and JIRA issue comments. The study revealed that none of the state-of-the-art tools could reliably assess sentiment in Stack Overflow discussions, with all tools struggling on positive and negative sentences, achieving recall and precision lower than 40% on negative sentences. This highlights the challenges of adapting general sentiment analysis tools for specific SE contexts. They also made their datasets and tools publicly available.
* **Ahmed et al. (2017)** created SentiCR, a sentiment analysis tool specifically designed for code review comments, motivated by the poor performance of existing sentiment analysis tools on SE datasets. SentiCR, trained on manually labeled code review comments, achieved higher mean accuracy, precision, and recall in identifying negative review comments using the Gradient Boosting Tree algorithm. This study demonstrates the importance of domain-specific training when creating sentiment analysis tools. They made their tool and dataset publicly available.
* **Huq et al. (2019)** explored the relationship between developer sentiment and the introduction of bugs by analysing GitHub pull requests. Using SentiStrength-SE, they found that fix-inducing commits were associated with more positive comments and reviews but also more negative messages in preceding commits. The results suggest that negative sentiment might precede the introduction of bugs. This study demonstrates a novel application of sentiment analysis for predicting potential issues in code development.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Study | Method | Dataset | Performance | Key Features |
| Ahmed et al. (2017) - SentiCR | Gradient Boosting Tree | Code review comments (1600 samples) | Accuracy: 83% Precision: 0.87 Recall: 0.88  F1: 0.87 | * Domain-specific training * Specialized preprocessing * Available tool and dataset |
| Islam & Zibran (2018) - SentiStrength-SE | Rule-based + Lexicon | JIRA issues, Stack Overflow, code reviews (5000+ samples) | Accuracy: 77.3% Precision: 0.83 Recall: 0.81  F1: 0.82 | * SE-specific dictionary Context-aware analysis * Enhanced preprocessing |
| Lin et al. (2018) | Tool comparison study | Stack Overflow, App reviews, JIRA (6000+ samples) | General tools: Precision & Recall < 40% on negative sentences | * Comparative analysis * Public benchmark dataset |

Table 1. Comparison of State-of-the-art Approaches

Our proposal builds upon these works by leveraging modern transformer architectures (BERT) while incorporating domain-specific training approaches. This combination aims to achieve high performance in analysing PR review sentiments.