StackEmo: Towards Enhancing User Experience by Augmenting Stack Overflow with Emojis

Akhila Sri Manasa Venigalla
Research in Intelligent Software & Human Analytics (RISHA) Lab
Department of Computer Science and Engineering
Indian Institute of Technology Tirupati
Tirupati, India
cs19d504@iittp.ac.in

Sridhar Chimalakonda
Research in Intelligent Software & Human Analytics (RISHA) Lab
Department of Computer Science and Engineering
Indian Institute of Technology Tirupati
Tirupati, India
ch@iittp.ac.in

ABSTRACT

Many novice programmers visit Stack Overflow for purposes that include posing questions and finding answers for issues they come across in the process of programming. Many questions have more than one correct answer on Stack Overflow, which are accompanied by number of comments from the users. Comments help developers in identifying the answer that better fits their purpose. However, it is difficult to navigate through all the comments to select an answer. Adding relevant visual cues to comments could help developers in prioritizing the comments to be read. Comments logged generally include sentiments of users, which, when depicted visually, could motivate users in reading through the comments and also help them in prioritizing the comments. However, the sentiment of comments is not being explicitly depicted on the current Stack Overflow platform. While there exist many tools that augment or annotate Stack Overflow platform for developers, we are not aware of tools that annotate visual representations of sentiments to the posts. In this paper, we propose *StackEmo* as a Google Chrome plugin to augment comments on Stack Overflow with emojis, based on the sentiment of the comments posted. We evaluated *StackEmo* through an in-user likert scale based survey with 30 university students to understand user perception towards *StackEmo*. The results of the survey provided us insights on improving *StackEmo*, with 83% of the participants willing to recommend the plugin to their peers. The source code and tool are available for download on GitHub at: https://github.com/rishalab/StackEmo, and the demo can be found here on youtube: https://youtu.be/BCFlqvMhTMA.

CCS CONCEPTS

• **Software and its engineering → Software notations and tools**.

KEYWORDS

stack overflow, emotion analysis, latent dirichlet allocation, emojis

ESEC/FSE '21, August 23–28, 2021, Athens, Greece

ACM ISBN 978-1-4503-8562-6/21/08…$15.00
https://doi.org/10.1145/3468264.3473119

ACM Reference Format:
Akhila Sri Manasa Venigalla and Sridhar Chimalakonda. 2021. *StackEmo*: Towards Enhancing User Experience by Augmenting Stack Overflow with Emojis. In *Proceedings of the 29th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE '21), August 23–28, 2021, Athens, Greece*. ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3468264.3473119

1 INTRODUCTION

Question and Answer (Q&A) websites such as Stack Overflow support knowledge sharing by facilitating users to ask and answer questions in the field of programming [3]. Users can comment on the answers to give suggestions or reviews to improve the answer, which currently account to 78M[1] comments on the platform. The number of questions on Stack Overflow is increasing at a rapid rate, with about 3.16 questions being posted per minute[2]. About 2.7M[3] of the total questions on Stack Overflow are unanswered. Researchers have identified that social factors such as gender, age, status of users in the community, linguistics, sentiments of the posts, and so on influence the response time for a question [10, 12, 18]. Analysing sentiments of posts on Stack Overflow could provide insights on the emotions of users, help in assessing quality of questions [15] and in obtaining desirable answers to questions [6, 12].

Users of Stack Overflow with reputation 30 can post comments on answers that are not locked[4]. These comments specify changes and suggestions to improve answers posted. Most of the questions on Stack Overflow have more than one answer that is correct. Apart from the number of upvotes for answers, comments provide scenarios for which certain solution works or not, facilitating users to choose the best suited answer among the available ones. Displaying sentiments in these comments as emojis appended to the comments could motivate users in choosing the best suited answer based on the comments accompanying it. Several studies have been performed on analysing sentiments of software artifacts to understand developer perceptions, predict status of artifacts and so on [7–9, 19]. Guzman et al. have analysed commit comments of repositories on GitHub and provided their insights on positive and negative emotions in the commits with respect to factors such as programming languages used and number of collaborators to the repository [8].

[1] https://data.stackexchange.com/
[2] https://bit.ly/3eh2aGj
[3] https://stackoverflow.com/unanswered/tagged/?tab=noanswers
[4] https://meta.stackexchange.com/questions/19756/how-do-comments-work

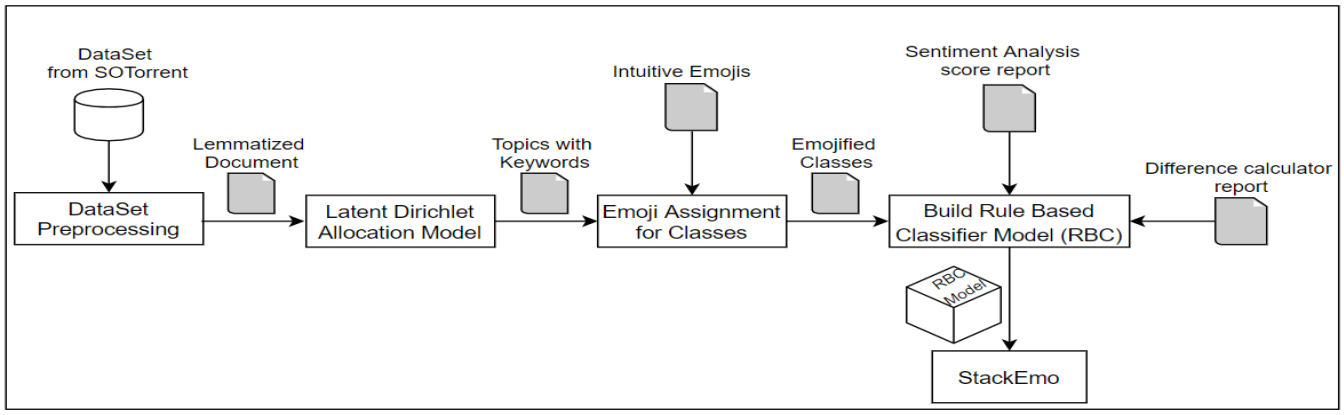

Figure 1: Design Methodology of *StackEmo*

LDA based Keywords	Sentiment Analyser Score	Class Name	Emojis
[understand, better, than]	[score > 0]	Useful	😎/😊
[understand, better, than]	[score < 0]	Not Useful	😒/🤦/😵
[right, works, correct]	[score - any] (generally >=0)	Like/Correct	👍
[wrong, not, work]	[score - any] (generally <=0)	Dislike/Incorrect	🤦/👎/👎
[question, how, what]	[score - any]	Asking Question	🤷/❓
[here, answer, this]	[score - any]	Answering Comment	💁
[nice, decent, thanks, perfect]	[score - any] (generally >=0)	Positive Feedback	🙏/😃/😇
[bad, help, doesn't]	[score - any] (generally <=0)	Negative Feedback	😕/😠/👿
[curious, anxious, how, why, not]	[score - any]	Curiosity	🤔/😱

Figure 2: Keywords and Emojis of Classes

It has also been emphasized that emotions of developers in agile teams should be considered while adopting agile principles to make changes to agile projects [9]. *MEME* has been developed as a method to extract emotions from commits, pull requests and issues of repositories on GitHub [22]. An integrated approach proposed in [4] that classifies Stack Overflow posts based on the emotions being expressed, into 6 emotion categories, reiterates the importance of understanding emotions in open source platforms such as Stack Overflow. Analysing emotions on Stack Overflow and depicting them in a visual format could further improve user experience.

As emphasized by Novielli et al. [13], the idea of analysing emotions of developer-communications, on platforms such as Stack Overflow and GitHub, is gaining importance. Emojis (😎, 🙏) are considered to enhance communication and largely contribute towards understanding sentiments being expressed [5]. Also, to the best of our knowledge, we are not aware of existing work that augments Stack Overflow with emojis based on emotions of the posts. Existing tools such as *StackDoc* [20], *ExampleCheck* [16] and *SOTagger* [21] proposed to augment Stack Overflow, are limited to providing further technical or contextual information based on the questions asked and solutions mentioned. We present *StackEmo*, as a Google Chrome plugin to augment Stack Overflow with emotions 😞🤦‍♀️😇 of the comments.

2 DESIGN METHODOLOGY OF *STACKEMO*

To design *StackEmo*, we considered the commonly used *Sentiment Analyser* library of Natural Language Toolkit (NLTK). This library provides a score for a given textual sentence and is commonly used by researchers, for sentiment analysis on twitter [11, 17].Higher score values contribute to more positive sentiments and lower score values contribute to more negative sentiments.

However, we have identified through manual inspection of comments, that scores depicting only positivity or negativity do not suffice to represent emotions. Though there exists approaches such as *HOMER*, *EmoTxt* and so on [4], they are limited to six emotion categories and one neutral category and are not specific to Stack Overflow platform. Hence, we have tried to classify comments into nine emotion based categories, based on manual inspection of 200 comments. The most prominent sentiments being conveyed by the comments were listed, which resulted in nine categories. The nine classes are labelled based on the prominent emotions observed, based on card sorting approach [14], which involved debates and discussions among the authors. Each author initially considered multiple words from LDA clustering of 200 comments, divided all the words into nine classes and labelled these classes, using open card sorting approach. Then the words in the nine classes, categorized by each of the author were compared, to further identify similarity and dissimilarity of class labels. Further discussions and debates among the authors helped in normalizing all the class labels.

We have extracted Stack Overflow comments from SOTorrent [2] dataset and passed the same to LDA model, with number of classes parameter equated to nine. The classes were named by analysing the keywords in each topic, and the keywords were accompanied by their synonyms. A difference calculator function has then been defined to calculate similarity of a given sentence with each of the defined classes, and consequently assign the emotion of most similar class to the sentence. *StackEmo* has been developed as Google Chrome plugin to Stack Overflow. We have followed a 5 step process, as shown in Figure 1 in developing *StackEmo*.

Step 1 - DataSet Extraction. We have downloaded the file - `Comments.xml.7z` from SOTorrent[2] to analyse comments on Stack Overflow. We randomly selected about 50K comments from the dataset for tool development.

Step 2 - DataSet Preprocessing. We have preprocessed the resulting dataset of 50K comment posts for stopword removal and lemmatized the document using *spaCy*.

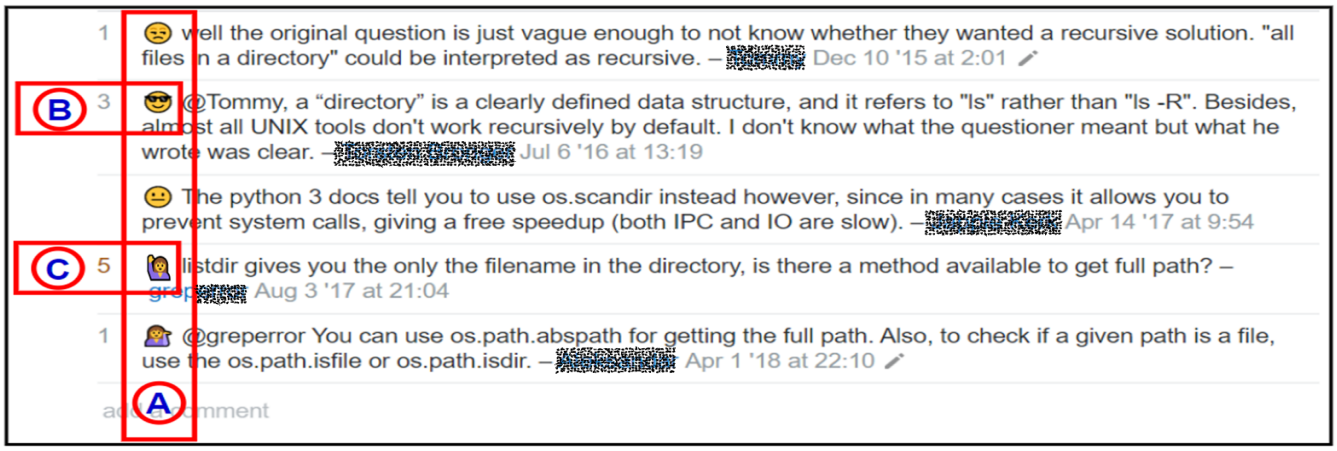

Figure 3: Snapshot of results of *StackEmo* Plugin. [A] indicates comments appended with emojis by *StackEmo*, [B] indicates a comment that is appended with *Useful* class emoji and [C] indicates a comment appended with *Asking Question* class emoji

Step 3 - Latent Dirichlet Allocation Model. We applied LDA model to classify the processed dataset into nine classes. The basis for our choice of using LDA model to categorize text is the existing work of Allamanis et al. [1] and Venigalla et al. [21] to contextually classify questions on Stack Overflow. We relied on manual inspection of 200 comments on Stack Overflow for deciding number of classes, and the categorization was based on card sorting approach.

Step 4 - Emoji Assignment for Classes. Based on the resulting output categories of LDA, we prepared keyword sets, which includes groups of keywords (bigrams and trigrams) for each class and labelled the classes with relevant emojis as shown in Figure 2, resulting in emojified classes.

Step 5 - Build Rule Based Classifier Model. We then defined a Rule Based Classifier (RBC) that takes into consideration results generated by difference calculator and NLTK sentiment predictor to classify the text passed based on predefined classification rules.

StackEmo extracts the comments of the post, selected by user, on Stack Overflow, which are processed to omit non-aplhanumeric characters and words in the comments are converted to their base forms. The sentiment of these words is analysed and the extent of positivity/negativity of the sentence is identified through sentiment score obtained as a result of this analysis. The processed text is analysed by difference calculator and the difference of the comment with each of the classes is calculated to identify the class having least difference. Sentiment score and class with least difference are analysed with respect to the rules defined, and the comment is classified into one of the nine classes, and the corresponding emoji is appended to the comment, as mentioned in Figure 2.

3 USER SCENARIO

Consider *Veda* to be a programmer who visits Stack Overflow with a query aimed to *list all files of a directory using python*. She is then directed to the corresponding question on Stack Overflow . She goes through the first answer presented and wishes to learn more about *slicing* and also the authenticity of the presented answer. She scrolls down to view the comments and finds it time consuming to read through all the comments present. She then adds *StackEmo* as an extension to Google Chrome and reloads the question page. She is then displayed with comments that are appended with emojis, as shown in Figure 3[A]. *Veda* then reads through the comments that are intended to enhance the answer, based on the emojis that belong to classes *Useful* and *Like*, that imply positive suggestions, specifically referring to a solution that *works* and that the solution *helps in improving the user understanding*, as shown in Figure 3[B]. She conveniently ignores comments augmented with emojis assigned for class - *Asking Question* (Figure 3[C]), as it is perceived to deal with further questions.

4 EVALUATION

We followed similar approach of in-user survey used by Zhang et al. in [23] to evaluate *StackEmo* with 30 university students in the age group 18 to 23 years. 28 of them are undergraduate computer science students and the remaining two participants are post graduate students of computer science, who were in their third semester (age group of 21-23 years). 25 of the 28 participants were in their fifth semester (age group of 19-20 years), 2 participants in their seventh semester (age group of 20-21 years) and one participant was in her third semester (age group of 18-19 years) of under graduation.

All the volunteers were requested to install *StackEmo* and analyse the emoji-augmented comments. A clear documentation that specifies location which *StackEmo* could be downloaded, steps to be followed to install it as chrome plugin and steps to evaluate the plugin, is sent to volunteers. An in-lab survey has been conducted for 25 undergraduate participants and 2 post graduate participants for a duration of 15 minutes. An email was communicated to rest of the 3 participants. They were instructed to follow the steps mentioned in the shared document to assess *StackEmo*. As a part of the survey, all participants were instructed to freely browse comments on Stack Overflow and review the appended emojis. Then, participants were requested to navigate to user survey link provided and

answer the questions using 5-point likert scale. The questionnaire used for survey and results of the survey are presented here[5].

5 RESULTS

To evaluate the survey responses, we have given weighted values for each point of the likert scale, such that *Strongly Agree* corresponds to value five, while *Strongly Disagree* corresponds to value one. The average score for each question is then considered as the evaluation metric of *StackEmo* based on user responses. The results of the survey indicate that 85% of the participants liked the interface of *StackEmo*. 76.8% of participants have agreed that *StackEmo* has satisfactorily rendered emotions to posts, with appropriate emojis. However, three of the participants reported instances of emojis not being appropriate for the comment. The survey also revealed that *StackEmo* has helped volunteers to identify useful comments and motivated them to view comments and get better insights. Participants have mentioned their suggestions to enhance *StackEmo* by reconfiguring the emojis. 25 of 30 participants have agreed to recommend *StackEmo* to their peers. Two of the participants also pointed out that *StackEmo* fails to render emojis when clicked on *"show more comments"*.

6 THREATS TO VALIDITY

StackEmo has been developed as a first step towards augmenting Stack Overflow posts with emotions being conveyed by the posts. *StackEmo* classifies the comments into one of the nine proposed classes and augments the comments on Stack Overflow with predetermined emojis. The class labels and choice of number of classes is based on the results of manual inspection and discussions among the authors and two other developers, as a part of card sorting approach. However, this number of classes and the class labels of existing nine classes could differ based on the developers' perception of data.

The keywords specified for each class are based on manual analysis of top 20 comments for each class, generated by LDA model. The rank of these comments depends on probability values generated by the model for comments belonging to each class. Classification based on the number of comments inspected, might result in inaccuracies, considering the huge amount of data available. To validate the correctness of classification, we have manually analysed emotions of 30 random comments on Stack Overflow. We observed that about 8 comments were tagged with different emojis than expected.

The correctness of sentiment scores obtained is dependant on efficiency of Sentiment Analyser provided by NLTK. Researchers have also mentioned the inefficiency of *nltk sentiment analyser* in the area of software engineering. There is thus, a need to adopt better analysers to analyse sentiment of posts present on Stack Overflow and other coding platforms. Also, the keyword synonyms are generated by synonym generator of *wordnet* library provided by NLTK. As a result, few synonyms might not have been considered, if not reported by the generator, and few might be out of context in view of the domain being considered. The choice of emojis is also based on general knowledge and acceptance of the purpose of emojis, which we plan to revise in the next version. Moreover, specific comments on Stack Overflow could express more than one emotion. However, only the emotion more prominently expressed, is considered for classification of the comment and emoji of the corresponding class is appended to the comment.

StackEmo was evaluated based on the user responses to the questionnaire in the survey, and thus is highly dependent on user perception. A concrete survey, using experimental and control groups, that considers monitoring of user actions with and without *StackEmo* could provide better insights about the plugin. Also, a study that includes both quantitative and qualitative analysis could be adopted with two participant groups, who use and do not use *StackEmo* respectively. This could help in understanding the user perception and usefulness of *StackEmo*.

7 CONCLUSION AND FUTURE WORK

Considering the increasing importance of sentiment analysis in various domains of software engineering and the effort being involved by programmers in learning through comments on Stack Overflow, we proposed an approach to augment Stack Overflow comments with emojis corresponding to their emotions. In this paper, we presented *StackEmo* that augments comments on Stack Overflow by appending them with emojis based on the sentiment of comments. The comments presented on Stack Overflow, for an answer are extracted and processed, when a question is being viewed on Stack Overflow . The processed comments are then analysed for the emotion being conveyed and classified into one of the predefined nine classes. An emoji is then appended to the comment based on predefined rules and displayed on Stack Overflow . *StackEmo* thus aims to provide insights about the comments, and could help users in reading through comments that only express a specific emotion, or in avoiding comments that express specific emotions. Presence of emojis on the platform might motivate the users to learn better by reviewing more comments and also might motivate practitioners to address more number of comments. However, the plugin has to be further evaluated to be able to comment on motivation factors of users. We have evaluated *StackEmo* with 30 university students, using a 5 point likert scale based questionnaire, through a in-user survey. 25 of the participants reported positive experience with the plugin and agreed to recommend *StackEmo* to their peers.

As a part of future work, we plan to improve the sentiment analysis by considering analysers specific to software engineering domain. We also plan to conduct a large scale user study to arrive at a consensus on the number of classes and features of these classes. We plan to investigate more concrete approaches to classify the posts based on emotions being conveyed. As suggested by the participants of user survey, we also plan to extend *StackEmo* to augment question and answer posts with emojis based on the sentiment of the posts.

ACKNOWLEDGMENTS

We thank all the volunteers for their time and honest feedback. We also thank our undergraduate student, Dheeraj Vagavolu for helping us in the development of the tool.

REFERENCES

[1] Miltiadis Allamanis and Charles Sutton. 2013. Why, when, and what: analyzing stack overflow questions by topic, type, and code. In *2013 10th Working Conference*

[5] https://docs.google.com/spreadsheets/d/1Svo_gha139IxZE_VXr2MsOeTqNlzQVcGDWyiC87wZOI/edit?usp=sharing

on Mining Software Repositories (MSR). IEEE, 53–56.

[2] Sebastian Baltes, Lorik Dumani, Christoph Treude, and Stephan Diehl. 2018. Sotorrent: Reconstructing and analyzing the evolution of stack overflow posts. In *Proceedings of the 15th international conference on mining software repositories*. 319–330.

[3] Anton Barua, Stephen W Thomas, and Ahmed E Hassan. 2014. What are developers talking about? an analysis of topics and trends in stack overflow. *Empirical Software Engineering* 19, 3 (2014), 619–654.

[4] Luis Adrián Cabrera-Diego, Nik Bessis, and Ioannis Korkontzelos. 2020. Classifying emotions in Stack Overflow and JIRA using a multi-label approach. *Knowledge-Based Systems* 195 (2020), 105633.

[5] Zhenpeng Chen, Yanbin Cao, Xuan Lu, Qiaozhu Mei, and Xuanzhe Liu. 2019. SEntiMoji: an emoji-powered learning approach for sentiment analysis in software engineering. In *Proceedings of the 2019 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. 841–852.

[6] Zhenpeng Chen, Yanbin Cao, Huihan Yao, Xuan Lu, Xin Peng, Hong Mei, and Xuanzhe Liu. 2021. Emoji-powered sentiment and emotion detection from software developers' communication data. *ACM Transactions on Software Engineering and Methodology (TOSEM)* 30, 2 (2021), 1–48.

[7] Daniela Girardi, Nicole Novielli, Davide Fucci, and Filippo Lanubile. 2020. Recognizing developers' emotions while programming. In *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering*. 666–677.

[8] Emitza Guzman, David Azócar, and Yang Li. 2014. Sentiment analysis of commit comments in GitHub: an empirical study. In *Proceedings of the 11th Working Conference on Mining Software Repositories*. ACM, 352–355.

[9] Kashumi Madampe, Rashina Hoda, and Paramvir Singh. 2020. Towards Understanding Emotional Response to Requirements Changes in Agile Teams. In *New Ideas and Emerging Results track of the 42nd IEEE/ACM International Conference on Software Engineering, ICSE2020*.

[10] Anna May, Johannes Wachs, and Anikó Hannák. 2019. Gender differences in participation and reward on Stack Overflow. *Empirical Software Engineering* (2019), 1–23.

[11] Saif M Mohammad, Xiaodan Zhu, Svetlana Kiritchenko, and Joel Martin. 2015. Sentiment, emotion, purpose, and style in electoral tweets. *Information Processing & Management* 51, 4 (2015), 480–499.

[12] Nicole Novielli, Fabio Calefato, and Filippo Lanubile. 2014. Towards discovering the role of emotions in stack overflow. In *Proceedings of the 6th international workshop on social software engineering*. ACM, 33–36.

[13] Nicole Novielli, Fabio Calefato, and Filippo Lanubile. 2018. A gold standard for emotion annotation in stack overflow. In *2018 IEEE/ACM 15th International Conference on Mining Software Repositories (MSR)*. IEEE, 14–17.

[14] Celeste Lyn Paul. 2008. A modified delphi approach to a new card sorting methodology. *Journal of Usability studies* 4, 1 (2008), 7–30.

[15] Amit Kumar Mondal Mohammad Masudur Rahman and Chanchal K Roy. 2016. Embedded emotion-based classification of stack overflow questions towards the question quality prediction. In *Proc. 28th Int. Conf. Softw. Eng. Knowl. Eng.* 521–526.

[16] Anastasia Reinhardt, Tianyi Zhang, Mihir Mathur, and Miryung Kim. 2018. Augmenting stack overflow with API usage patterns mined from GitHub. In *Proceedings of the 2018 26th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. ACM, 880–883.

[17] Aliaksei Severyn and Alessandro Moschitti. 2015. Twitter sentiment analysis with deep convolutional neural networks. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 959–962.

[18] Karina Kohl Silveira, Soraia Musse, Isabel H Manssour, Renata Vieira, and Rafael Prikladnicki. 2019. Confidence in programming skills: gender insights from StackOverflow developers survey. In *Proceedings of the 41st International Conference on Software Engineering: Companion Proceedings*. IEEE Press, 234–235.

[19] Akhila Sri Manasa Venigalla and Sridhar Chimalakonda. 2021. Understanding Emotions of Developer Community Towards Software Documentation. In *2021 IEEE/ACM 43rd International Conference on Software Engineering: Software Engineering in Society (ICSE-SEIS)*. IEEE, 87–91.

[20] Akhila Sri Manasa Venigalla, Chaitanya S Lakkundi, Vartika Agrahari, and Sridhar Chimalakonda. 2019. StackDoc-A Stack Overflow Plug-in for Novice Programmers that Integrates Q&A with API Examples. In *2019 IEEE 19th International Conference on Advanced Learning Technologies (ICALT)*, Vol. 2161. IEEE, 247–251.

[21] Akhila Sri Manasa Venigalla, Chaitanya S Lakkundi, and Sridhar Chimalakonda. 2019. SOTagger-Towards Classifying Stack Overflow Posts through Contextual Tagging (S). *31st SEKE. KSI Research Inc. and Knowledge Systems Institute Graduate School* (2019), 493–639.

[22] Karl Werder and Sjaak Brinkkemper. 2018. MEME-Toward a Method for EMotions Extraction from GitHub. In *2018 IEEE/ACM 3rd International Workshop on Emotion Awareness in Software Engineering (SEmotion)*. IEEE, 20–24.

[23] Tianyi Zhang, Di Yang, Crista Lopes, and Miryung Kirnt. 2019. Analyzing and supporting adaptation of online code examples. In *Proceedings of the 41st International Conference on Software Engineering*. IEEE Press, 316–327.