Proceedings of the Eighteenth International AAAI Conference on Web and Social Media (ICWSM 2024)

Emoji Promotes Developer Participation and Issue Resolution on GitHub

Yuhang Zhou[1], Xuan Lu[2], Ge Gao[1], Qiaozhu Mei[3], Wei Ai[1]

[1] College of Information Studies, University of Maryland, College Park, USA
[2] School of Information, University of Arizona, Tucson, USA
[3] School of Information, University of Michigan, Ann Arbor, USA
tonyzhou@umd.edu, luxuan@arizona.edu, gegao@umd.edu, qmei@umich.edu, aiwei@umd.edu

Abstract

Although remote working is increasingly adopted during the pandemic, many are concerned by the low-efficiency of remote working. Missing in text-based communication are non-verbal cues such as facial expressions and body language, which hinders effective communication and negatively impacts work outcomes. Prevalent on social media platforms, emojis, as alternative non-verbal cues, are gaining popularity in the virtual workspaces well. In this paper, we study how emoji usage influences developer participation and issue resolution in virtual workspaces. To this end, we collect GitHub issues for a one-year period and apply causal inference techniques to measure the causal effect of emojis on the outcome of issues, controlling for confounders such as issue content, repository, and author information. We find that emojis can significantly reduce the resolution time of issues and attract more user participation. We also compare the heterogeneous effect on different types of issues. These findings deepen our understanding of the developer communities, and they provide design implications on how to facilitate interactions and broaden developer participation.

Introduction

Hybrid and remote working was considered a norm in the (post-)pandemic era. Yet, many people are concerned about the low efficiency in remote working, and some companies have launched the "return-to-office" policy to force employees to resume in-person working.[1] How to increase working efficiency in a virtual workspace has been a pressing problem. Some of the low efficiency can be attributed to the communication inconvenience in the virtual workspace. Despite the limited time on virtual meeting, text-based messaging is still a major format of communication in virtual workspace (Blanchard 2021). Missing in text-based communication are non-verbal cues such as facial expressions and body language, which convey subtle information about speakers' intent and sentiment, reduce misunderstanding, and are critical to successful communication. Emoji, as alternative non-verbal cues and prevalent in online communication, could be crucial to an effective discussion in the virtual workspace.

[1]https://www.bloomberg.com/news/articles/2022-11-10/musk-s-first-email-to-twitter-staff-ends-remote-work, retrieved May 2023.

Indeed, beyond social network platforms, such as Twitter and Instagram, emojis are also gaining popularity on professional online platforms, such as GitHub. As the largest host of source code in the world, GitHub offers distributed version control and source code management via the Git protocol. On GitHub, many distributed development activities are coordinated through *issues*,[2] which can be posted by any user to report software bugs, enhancement suggestions, or to solicit help. Conversations organized through issues and their comments can directly influence the quality of projects (Kavaler et al. 2017). Adequate and timely response to an issue is critical for the solution of the problem and the improvement of work outcomes.

Emojis have been supported in GitHub issues as early as 2014 (see an example in Figure 1), and has become increasingly popular in recent years. In June 2017, 0.58% of the issues contain emojis (Lu et al. 2018), while the ratio is 1.71% in our collected issues in June 2021.

Despite the increasing popularity, little is known whether using emojis benefits the developer community. That is, does using emojis improve the "outcome" of an issue, and does using more emojis bring more work outcomes for developers? Research on emoji usage on social media platforms provides supportive evidence, citing emojis as ideal nonverbal cues to express sentiment, strengthen expression, adjust tone (Hu et al. 2017), and engage the audience (Cramer, de Juan, and Tetreault 2016) in online communication, where facial expressions or body gestures are not available. However, in the more technical and professional conversations on GitHub, semantics and distribution of emojis differ from that used by the general population. Table 1 shows the 20 most frequently used emojis in our collected GitHub issues (details in Section *GitHub Issues and Problem Formulation*) and those released by the Unicode organization,[3] which ranks emojis based on their median frequency of use across multiple sources, representing the emoji usage in a more general context. Only five emojis (🎉, 🙏, 😄, 👍, and 😅) overlap in the two lists. Several of the most popular emojis on GitHub have domain-specific meanings, such as 🍾

[2]https://help.github.com/en/articles/about-issues, retrieved May 2022.

[3]https://home.unicode.org/emoji/emoji-frequency/, retrieved in May 2022.

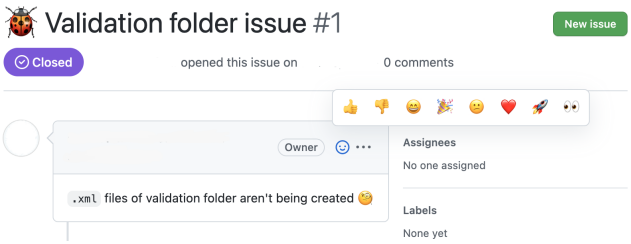

Figure 1: An example of emoji usage in GitHub Issues. The author puts 🐞 (lady beetle) in the issue title and 🧐 (face with monocle) in the issue body.

Platform	20 Most Frequently Used Emojis
GitHub Issues	👍🔴😄🚀👋🎉✅💸⚠️🤔 ✨🙂❤️😅🐞😊💰🙏📐😃
Unicode Consortium	😂❤️🤣👍😭🙏😘🥰😍😊 🎉😄💕🥺😅🔥☺️🤦❤️🐩

Table 1: 20 Most Frequently Used Emojis in GitHub Issues and reported by Unicode Consortium. We exclude emojis used as part of a template predefined by repository owner.

(bugs) and 🚀 (feature launching), while the popular emojis identified by the Unicode Consortium are mostly emotions and facial expressions. The gap in the emoji distribution is consistent with that reported in Lu et al. (2022).

In this work, we take the initiative to quantify the effect of emojis in promoting participation in the developer community. We ask whether emojis attract more user participation to issues and eventually help resolve them. Moreover, since issues may serve different functionalities on GitHub, we also explore the heterogeneous effects of emojis across different issue content. We summarize our hypotheses below:

H1 Using emojis in an issue increases the participation of GitHub users.

H2 Issues with emojis are more likely to be resolved and resolved in a shorter time period.

H3 There exist heterogeneous effects by issue content.

To verify our hypotheses, we use causal inference techniques to estimate the causal effect of emoji usage on issues. We first collect a dataset of GitHub issues. Next, we construct the confounders that may influence both the application of emojis and the resolution / participation of the issue. We then apply propensity score matching to quantify the causal effect of emojis. Besides estimating the average treatment effect (ATE), we also perform a fine-grained analysis to quantify the heterogeneous effect by issue types. Finally, we conduct a case study to observe how emojis lead to causal effect.

Related Work

Our work is mainly built on two streams of existing literature: the emoji functionalities and the GitHub issues.

Emoji Functionalities

The prevalence of emojis has led researchers to study their functionalities. Most previous research focuses on analyzing the emoji function from the semantic level to the audience level on social media (Hu et al. 2017; Ai et al. 2017; Cramer, de Juan, and Tetreault 2016; Zhou and Ai 2022; Zhou et al. 2024), and only a few studies are conducted on professional platforms. In general, emojis are used to decorate texts (Miller et al. 2017), adjust tones, provide additional emotional information, and engage the audience (Cramer, de Juan, and Tetreault 2016). These emoji-related research works mostly collect data from social media platforms, and researchers have identified the differences in emoji usage between communities and versions, such as apps, languages, cultures, genders, and platforms (Tauch and Kanjo 2016; Lu et al. 2016; Chen et al. 2018; Barbieri et al. 2016; Zhou, Lu, and Ai 2024).

Several research studies notice the uniqueness of emojis on professional platforms, such as GitHub, and explore the intention of emojis. Lu et al. (2018) first studied the emoji distribution on GitHub and used manual annotation to infer the intention of emoji usage. This study shows that the fraction of GitHub issues with emojis increases sharply after March 2016. Recently, researchers have also applied machine learning models to automatically label the intention of using emojis in GitHub posts to help project maintainers monitor the quality of communication (Rong et al. 2022). Moreover, Lu et al. (2023) also discusses the emoji effects on team resilience based on GitHub data. Researchers have also used emojis in GitHub issues for downstream tasks in software engineering, such as predicting developers' dropout (Lu et al. 2022), improving sentiment analysis (Chen et al. 2021). We extend this line of literature by quantifying the *causal effect* of using emojis on the outcome of GitHub issues, and specifically the resolution and user participation of GitHub issues. For other causal inference work on the NLP application, none of them utilize this technique on GitHub platform and emojis (Feder et al. 2021; Eckles and Bakshy 2017).

User Engagement in GitHub Issues

Many previous studies in the software engineering area focus on the sentiment analysis in social coding platforms. By exploring the relationship between sentiments on the social coding platform and developer productivity, researchers believe that understanding the relationship can help improve developer communication and production (Novielli and Serebrenik 2019; Sanei, Cheng, and Adams 2021). For issue discussions, it is widely recognized that the sentiments in issues can affect the productivity of developers (Mäntylä et al. 2016). Researchers analyze sentiments in issue discussions and find that the presence of emotions, especially positive emotions, is correlated with a shorter issue life cycle (Murgia et al. 2014; Ortu et al. 2015), such as its resolution time. Researchers have also verified that sentiments in issues are correlated with their social engagement, such as follow-up discussion and response (Sanei, Cheng, and Adams 2021). Taking into account emojis' functionality of

	Collected Issues	Issues w/ Emojis
# Issues	203,098	14,686
# Unique Users	99,062	9,398
# Unique Repos	90,115	9,185

Table 2: Statistics of the datasets used for causal inference.

expressing sentiments and adjusting tones (Hu et al. 2017), we hypothesize that using emojis in issues can affect both the follow-up discussion and resolution of the issue.

A common way to attract user participation in software development is by adding social signals, such as the Stack Overflow reputation score, GitHub badges, and GitHub followers (Trockman et al. 2018; Tsay, Dabbish, and Herbsleb 2014; Merchant et al. 2019). These signals are used as indications of developers' expertise and commitment, which reduces users' assessment cost and promotes more user participation (Trockman et al. 2018). Literature has already verified emojis for their social signaling function. Roberston, Magdy, and Goldwater (2021) has shown that skin-toned emoji can be regarded as a signal to provide users' identities and readers can perceive it. We hypothesize that emojis in GitHub issues are also social signals that help the audience perceive hidden information in GitHub issues, increasing the transparency of content to promote participation.

GitHub Issues and Problem Formulation

Data Collection

In our experiment dataset, we collect all GitHub issues in public repositories between June 1, 2020 and June 19, 2021 via a third-party project, GHTorrent.[4] GHTorrent monitors GitHub public event timeline. When someone creates, closes, or comments on an issue, GHTorrent retrieves the event content and dependencies, such as the related user and repository profiles. Only a small portion (1.6%) of issues use emojis, but they represent a growing trend, so we are interested in estimating the effect of using emojis for these "emoji issues", also known as Average Treatment Effect on the Treated, or ATT. To make the dataset trackable, we sample issues with or without emojis using a ratio of 1:10 from the raw dataset. Such a ratio ensures that most treated issues find the untreated issues with similar propensity scores, while ensuring computational efficiency for further processing.

In this work, we focus on the emojis used intentionally by the issue authors. But in practice, an issue may contain emojis because the repository sets up templates that the issue authors must follow. The use of templates confounds the estimation of the causal effect of emojis on GitHub issues, but since the template format is determined by the repository owner, it is difficult to model "template" representations through the embedding method. Also, many issues are written by bots and do not reflect users' intention. So we remove all the issues with predefined templates or posted by bots. More details of the data preprocessing, detecting templated issues and bots are described in Appendix. After preprocessing and filtering, we construct a dataset of 203,098 issues where 14,686 issues contain one or more emojis. Table 2 reports more detailed statistics of the used issues.

[4]https://ghtorrent.org/

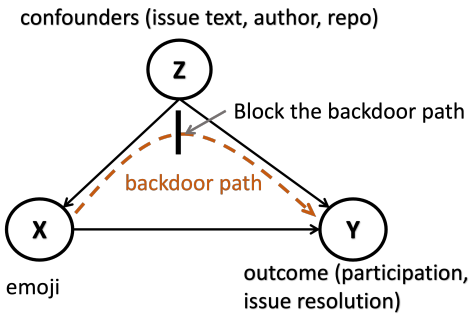

Figure 2: Causal graph for the backdoor adjustment

Outcome Variables

Given the available data from GHTorrent and consistent with previous studies on issue discussions (Sanei, Cheng, and Adams 2021), we use the comments below each issue, the closing status and time of an issue, to represent the participation and issue resolution, respectively. Specifically, we construct the following outcome variables:

Developer Participation (H1) We use whether getting comments, the number of comments, and the number of unique commenting users to measure the user participation level.

Issue Resolution (H2) We calculate whether an issue is closed in 30, 60, 90, and 180 days, and the time before closing (for issues closed in 180 days after being posted) to measure the likelihood and speed of issue resolution.

Ideally, the observed difference of the outcome variables between the treatment group (issues with emojis) and the control group (issues without emojis) is the effect of using emojis. However, both the emoji usage and the outcome might be influenced by the same confounders, such as the issue contents and the authors. Therefore, instead of reporting the observed average difference between two groups, we use a principled causal inference technique to estimate the treatment effect of using emojis.

Propensity Score Matching

Formally, our objective for this project is to estimate the causal effect of emoji usage (binary cause X) on the issue resolution or user participation (Y) with multiple confounders (Z), such as the length of the issue, the topic of the issue, and the popularity of the repository. The relationship of the cause, outcome and confounder relationship are visualized in Figure 2. We use do-calculus language (Pearl 1995) to quantify the causal impact of emojis. By mathematical formulation, our aim is to estimate $P(Y|do(X))$.

Since we cannot observe the counterfactual outcome of adding or deleting emojis for an issue, we can only estimate the causal effect by comparing observable issues with or

Category	Confounder Variables
Issue Text	created week, # of characters in the title, # of tokens in the title, # of characters in the body, # of tokens in the body, politeness score, topic distribution, issue type labels
Issue Author	author account age (days), # author followers, # author following, # public repos of the author
Repository	repo age (days), # repo stars, # repo forks, # open issues in the repo, # repo watch

Table 3: Confounders collected for the causal effect of emoji usage.

without emojis. To quantitatively measure the effect of emoji usage, since the confounders can influence the outcome, we apply the backdoor adjustment (Pearl 1995) to block every back-door path between X and Y. Controlled confounders Z should not be descendants of X. When Z meets the backdoor criterion,

$$P(Y|do(X=x)) = \sum_z P(Y|X=x, Z=z)P(Z=z)$$

The average treatment effect (ATE) of X (emojis) on Y (that is, the closing time or the comment number) is

$$\begin{aligned} ATE &= \mathbb{E}[Y|do(X=1)] - \mathbb{E}[Y|do(X=0)] \\ &= \mathbb{E}_Z[\mathbb{E}[Y|X=1,Z] - \mathbb{E}[Y|X=0,Z]] \end{aligned}$$

Although we can follow the equations to calculate the treatment effect, in our scenario, Z is a high-dimensional vector that represents all confounders. It is difficult to find an issue containing emojis and an issue not containing emojis, but with the same values of Z. (Rosenbaum and Rubin 1983) has proposed an alternative way to calculate causal effect by matching not on Z, but on the propensity score $R = P(X=1|Z)$, which is called propensity score matching (PSM). The equations of calculating ATE become

$$ATE = \mathbb{E}_R[\mathbb{E}[Y|X=1,R] - \mathbb{E}[Y|X=0,R]]$$

Therefore, we need to match issues containing emojis with issues having similar R but not containing emojis. To estimate the propensity score $R = P(X=1|Z)$ for each issue, we need to observe all possible confounders Z that can influence both the usage of emojis and the outcome of the issues.

Collection of Confounders

To control for the confounder data of issues, we construct the following covariates based on available data from three sources: the text of the issue, the author of the issue, and the repository profile.

Issue Text Confounders

For issue covariates, we collect its creation time, length of its title and body (measured as the number of characters and tokens). In addition, to control for the semantics of the issues,

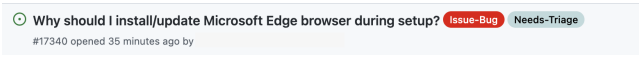

Figure 3: The example of an issue with self-annotated labels

we measure the issue type, issue topic, and issue politeness as proxies for the semantics of the issue.

Issue Types Authors post issues for various reasons, such as reporting bugs, seeking help, or requesting new features, and they sometimes label issue types explicitly with self-annotated tags. Figure 3 shows an example where the author labeled the issue with "Issue-Bug" and "Needs-Triage."

According to the data from GHTorrent, the 3 mostly frequently used and content-related labels in GitHub issues are *bug* (reporting bugs), *question* (asking for help), and *feature* (requesting new features). Using the above three self-annotated labels as ground-truth, we train three classifiers to automatically label whether the issue belongs to bug, feature, or question type. Details of the classifiers are given in Appendix B.

Topics in Issues Besides issue types, issues also contain multiple topics, such as game, programming language, or operating system. To capture the latent topics, we use the unsupervised topic modeling, the Latent Dirichlet Allocation (LDA) model, to generate the representation of the topic (Blei, Ng, and Jordan 2003). The reason we do not directly use contextualized document embeddings, such as BERT embeddings (Devlin et al. 2018), as the confounder is that there is no clear interpretation of each dimension of document embedding, and it is hard for us to check whether this confounder is balanced after matching. Therefore, we chose LDA topic embeddings to represent the confounder of issue content. After enumerating the topic number as 10, 20, 30, and 40 to train multiple LDA models, we observe that when setting the topic number as 30, the keywords (words with the highest weights) of each topic are the most interpretable, and the number of overlap keywords between topics is the smallest. Thus, for each issue, we obtain a 30-dimensional topic distribution vector, where each dimension is the distribution of the topic in the issue.

Politeness in Issues Since emojis have the functionality to signal politeness (Escouflaire 2021) and politeness of issues may encourage issue resolution and user participation, we measure the issue politeness score as a confounder. Danescu-Niculescu-Mizil et al. (2013) has proposed a politeness classifier with domain-independent lexical and syntactic features, which is trained on Wikipedia and Stack Exchange data and achieves 78.19% on their Stack Exchange dataset. For each issue, we apply the pretrained classifier to calculate the politeness score as the issue's politeness.

Issue Author and Repository Confounders

Just as we need to control for the demographics for human subject, we should also control the "demographics" of an issue. Therefore, we extract author and repository information of an issue, such as their popularity and primary programming language, which may confound the treatment (using emoji) and outcome. For authors, we collect the number of

followers and followings, the number of public repositories, and the account age. For repositories, we collect the number of stars, the number of forks, the number of open issues, the programming language used, and the repository age. Table 3 summarizes our collected covariates.

Causal Inference of Emojis on Issues

Propensity Score Estimation

In our scenario, the propensity score R is the probability of issues using emojis given confounders (Z). Following the econometrics literature, we choose a parametric model, namely logistic regression (LR), to estimate the propensity score (Angrist 2008, Chapter 3). Our LR will predict whether the issue contains emojis given confounder variables Z, and the probability of inference on each issue is the estimated propensity score. Since most of the issues in our dataset do not contain emojis, the model can easily achieve high accuracy by only majority class selection. To address the class imbalance, we first perform undersampling to ensure the same number of issues with and without emojis in the training dataset. After training, we apply the model on the original unbalanced dataset and find matched untreated issues for each treated issue.

Evaluating the propensity score estimation is tricky as we cannot use the out-of-sample accuracy. An issue is either treated or untreated in the dataset, and we do not know the ground truth of its probability of being treated. Instead, we rely on the balance check (detailed in Section *Nearest Neighbor Matching*) to examine whether the matched samples (of both treated and untreated) are similar or comparable other than their treatment status. Specifically, we want to check whether the covariate distribution is balanced between the matched samples.

Nearest Neighbor Matching

Since the assessment of the propensity score estimation is closely related to our chosen matching method, we first introduce the nearest neighbor matching. Here, we choose the most common implementation for the propensity score matching, pair matching (Austin et al. 2021) (one-to-one matching). We pair each treated issue with an untreated issue and then average the difference of the matched pairs as the average treatment effect on treated (ATE). Literature suggests that matching with replacement may lead to the condition that an untreated issue is used in multiple pairs, which may cause the variance estimation of the treatment effect to be more complex (Hill and Reiter 2006), we implement matching without replacement in this paper. Once an untreated issue is selected to match a treated issue, it will no longer be used to pair with other treated issues.

We implement pair matching using greedy nearest-neighbor matching (NNM) instead of optimal matching. Optimal matching finds pair matchings that minimize the sum of the score difference within each pair (Austin et al. 2021). Although it can find the global optimal matching, it is computationally infeasible for our scale of data, and previous study has shown that optimal matching does not behave much better in forming balanced pairs (Gu and Rosenbaum 1993). For greedy matching, we randomly select a treated issue and match it with an untreated one with the closest propensity score, and repeat the process for other treated issues.

We further ensure the quality of matching by applying a caliper restriction. That is, a matched pair is acceptable only if the within-pair difference of the propensity scores is less than the predefined caliper width. We discard treated issues that cannot be matched to untreated issues of similar propensity score. We set the caliper width to 0.2 of the standard deviation (SD) of the propensity score, which has been shown to work well in multiple settings (Austin 2011). We also observe that the proportion of emoji usage for each issue type is not similar, so to keep a balanced distribution of issue types, we enforce the exact matching on issue types. That is, issues will only be matched to other issues of the same type. We will discuss issue types in more details in Section *Heterogeneous Effects by Issue Types*.

To assess the matching of the propensity score, we expect the distribution of the confounders in the treatment and control groups to be balanced. Standardized mean difference (SMD) is a commonly used measurement to examine the balance of the covariate distribution between groups (Zhang et al. 2019). The guidelines indicate that 0.1 or 0.25 represent reasonable cutoffs and a higher value of SMD indicates that the covariate distribution in one group is too different from one another for reliable comparison (Stuart, Lee, and Leacy 2013). We visualize the SMD values of each covariate before and after matching in Figure 5 in Appendix. The nearest neighbor matching greatly reduces the distribution difference between the confounders of issues with emojis and without emojis. The SMD values of confounders are all below the 0.1 threshold, showing the similar covariate distribution between groups. More details of SMD check and additional refinement are discussed in Appendix.

Results and Discussion

Average Treatment Effect

By averaging the pairwise difference between treated and untreated issues, we estimate the unbiased treatment effect of emoji usage. We calculate the ATE of the outcome variables mentioned in Section *Outcome Variables* and show the results in Table 4. For the issue closing time variable, since GitHub repositories may automatically close some issues without any discussion for a long time and our dataset covers the issues over a one-year period, we cannot tell whether issues with long closing time are resolved. Therefore, we only consider the issue closing time less than 180 days, covering 98.6% of all closed issues. Besides the ATE, we also report the observed average difference between two groups before matching (Obs. Δ) as well as the average value in treatment groups (Avg.). Since we are measuring the average pairwise difference, we apply the paired z-test (Derrick et al. 2017) and the differences are all statistically significant at the 1% level.

Developer Participation We first examine whether emojis bring more discussion to an issue, which can be measured by the likelihood of getting comments or the number

Outcome Variable	ATE	Obs. Δ	Avg.
Developer Participation			
getting comments	0.041**	0.112	0.476
# of comments	0.118**	0.346	1.236
# of comment users	0.113**	0.257	0.811
Issue Resolution			
% closed in 180 days	0.033**	0.056	0.391
% closed in 90 days	0.033**	0.052	0.378
% closed in 60 days	0.035**	0.053	0.370
% closed in 30 days	0.038**	0.051	0.346
issue closing time (day)	-1.751**	-0.822	11.570

Table 4: The treatment effect of using emojis in GitHub issues. Obs. Δ: The observed average difference between the control group and treatment group without matching on propensity scores. Avg: the average value in the treatment group. Significance level: ** $p < 0.01$, * $p < 0.05$, paired z-test.

of comments. As shown in Table 4, issues with emojis are 4% more likely to receive comments. On average, issues with emojis get 0.118 more comments than those without emojis. We further check the number of users participating in the conversation and find that using emojis attracts 0.113 more users to participate in issue discussion. All results confirm our hypothesis H1 that using emojis in issues attracts more participation in issue discussions.

Issue Resolution People do not simply want their issues to be watched; they want their issues to be resolved – bugs fixed, features added, and questions answered. Beyond the increased participation, does emoji usage actually help to resolve the issues? By looking at the closing status of the issues at different time intervals, we find that issues with emojis are more likely to be closed in all time periods, and the effect is larger for the shorter time periods (ATE increases from 3.3% to 3.8% when interval decreases from 180 to 30 days). For issues that are closed in 180 days, we find that using emojis speeds up their resolution by an average of 1.751 days. Therefore, we may infer that the attention and participation that emojis attract are not from mere bystanders. Shorter closing time and larger proportion of closed issues with emoji use confirm our hypothesis H2.

Sensitivity Analysis

A major threat to validity for propensity score matching is that there might be unobserved confounders influencing the estimated causal effect of emojis. We perform a simultaneous sensitivity analysis to examine the extent to which our conclusion is robust to unobserved confounders (Gastwirth, Krieger, and Rosenbaum 1998). The intuition is that if an unobserved confounder invalidates our conclusion, it should be strongly correlated with the treatment and/or the outcome. We may use the observed confounders as reference points for the strength of such correlation, and ask the question: if an unobserved confounder has the same strength of correlation as that of the observed confounders, what would the significance of the treatment effect become, that is, how large would the p-value become? If the new p-value is still small, it means that the treatment effect is still significant even with the presence of such an unobserved confounder. Formally, we denote $\mathcal{T}$ as the upper bound of the odds ratio between the unobserved confounder and the treatment, and Δ as the upper bound of the odds ratio between the unobserved confounder and the outcome. The goal of the simultaneous sensitivity analysis is to see whether an unobserved confounder with a combination of $\mathcal{T}$ and Δ would make the estimated causal effect insignificant ($p \geq 0.05$) (Liu, Kuramoto, and Stuart 2013; Gastwirth, Krieger, and Rosenbaum 1998). Given the values of $\mathcal{T}$ and Δ, we can calculate the upper bound p-value as follows:

$$p(\theta) = \frac{\Delta}{1+\Delta} \quad p(\pi) = \frac{\mathcal{T}}{1+\mathcal{T}}$$
$$p^+ = p(\pi) \times p(\theta) + (1 - p(\pi)) \times (1 - p(\theta))$$
$$\textit{upper bound P-value} = \sum_{a}^{T} \binom{T}{a} (p^+)^a (1-p^+)^{T-a}$$

where T is the total number of discordant pairs in which the outcomes differ within the pair and a is the number of pairs in which the issue with emojis has an outcome and the issue without emojis does not (Liu, Kuramoto, and Stuart 2013). In our experiments, the values of T and a are 7,905 and 3,805 respectively.

We err on the conservative side in fixing the values of $\mathcal{T}$ and Δ as the largest odds ratio between observed confounders and the treatment variable and the outcome variable, respectively (Liu, Kuramoto, and Stuart 2013). For binary observed confounders, we use their odds ratios with the treatment and outcome variables. For continuous observed confounders, we fit logistic regression models between the confounders and treatment or outcome, and use the learned model parameters to estimate the odds ratio (Bland and Altman 2000). We report the values of $\mathcal{T}$ and Δ and the upper bound p-values for five binary outcome variables in Table 5. We observe that all the upper bound p-values are less than 0.05. The results can be interpreted as that the estimated treatment effect of emojis is still significant, even if there exists an unobserved confounder which has the same strength of association with the binary outcome and the treatment as the strongest ones of all observed confounders.

Besides the sensitivity analysis on weakening the unconfoundedness assumption, we also verify the robustness of our findings by repeating the analysis on alternative specifications (Athey and Imbens 2017), such as using dataset from a different time span, choosing a different propensity score estimator, or changing the number of topics in the LDA topic model. We repeat the analysis on the GitHub issues created between July 2018 and June 2019, which is before the pandemic shock, and find that the treatment effects of using emojis have the same direction and similar significance levels to our main result. More details are shown in Appendix. We also apply the Gradient Boosted Regression Tree method (rather than logistic regression) to re-estimate the propensity score, or change the topic number to 15 or 45 to retrain the LDA model and repeat the analysis. Both show similar

Binary Outcome Variable	Δ	$\mathcal{T}$	p-value
Developer Participation			
getting comments	4.290	1.163	0.015
Issue Resolution			
% closed in 180 days	4.280	1.163	0.015
% closed in 90 days	4.188	1.163	0.013
% closed in 60 days	4.542	1.163	0.020
% closed in 30 days	4.816	1.163	0.026

Table 5: Simultaneous sensitivity results for binary outcome variables. Δ denotes the largest odds ratio between the confounder and the outcome. $\mathcal{T}$ denotes the largest odds ratio between the confounder and the treatment. p-value is the calculated upper bound p-value.

patterns in the significance and directions of the estimated treatment effect (details in Appendix).

Heterogeneous Effects by Issue Types

The results in Section *Average Treatment Effect* show that using emoji can promote developer participation and speed up issue resolution. However, authors may post different types of issues, choose among thousands of emojis, and expect different kinds of responses. We wonder whether the emoji effect is consistent across issue types. In this section, we re-estimate the treatment effect of different issue types and explore the heterogeneous treatment effect (HTE) of emoji usage. We will follow the definition in Section *Issue Text Confounders*, and classify issues with three types: bug, question, and feature.

Since we only calculate the treatment effect of a subset of the treatment group, the definition of propensity score also changes from $p(\text{issue contains emojis}|\text{confounders})$ to $p(\text{issues contains emojis}|\text{confounders, issue type})$. We retrain the logistic regression model to estimate the new propensity score within each issue type and perform the nearest neighbor matching again. With the caliper restriction, we are able to ensure that the SMD value for each confounder variable is less than 0.25. We then recalculate the average pairwise difference as the HTE and present the results in Table 6 (the left half).

The HTE shows an interesting pattern that aligns with the types of issues. For bug and question issues, the issue authors hope to get them fixed or answered in a timely manner. Indeed, we observe significant treatment effects on issue resolution (the closing status and time). On the contrary, the effect on participation is not consistent with the ATE. Using emojis in question issues even reduces the number of comments and comment users. One possible reason is that bug and question issues with emojis are closed in a shorter time, participants may have less chance to comment, which is reflected as a negative treatment effect. On the other hand, using emojis in feature request issues does not have the significant effect on its closing status, but significantly increases the comments it receives. Such effects may also be desired, as the authors of the feature request may not expect to close issues as soon as possible, but do hope to attract more participation to move the ideas forward.

To summarize the HTE for different issue types, we find that using emojis benefits issue authors in achieving their *desired outcome*.

Emojis as Social Signals

Indeed, such heterogeneity in issue types sheds light on the possible explanations why emojis promote developer participation and issue resolution. As discussed in Section *Related Work*, social signals (Spence 2002) have been shown to be useful in attracting user participation. For example, Stack Overflow reputation scores, GitHub badges, and GitHub followers can signal a developer's expertise and commitment, which reduces users' assessment cost (Trockman et al. 2018; Tsay, Dabbish, and Herbsleb 2014; Merchant et al. 2019). Similarly emojis can also be regarded as observable signals to reduce the information asymmetry in online communication. We hypothesize that emojis on GitHub, as signals, can signal the overall topic and the author's attitude. By using emojis, the readers can better perceive the author's attitude and intent, which can reduce information asymmetry between authors and readers, and attract readers to participate in the issue resolution.

If such a hypothesis is correct, we should expect issue authors to purposefully select the emojis that signal their intent, and issues with the "right" emoji signals are more likely to have better outcome. Inspired by the observation of different emoji distribution between Unicode and GitHub in Table 1, we may expect different issue types are associated with different emojis. Using the emojis that are associated with specific types of issues serves as the social signals that promote the desired outcome.

To this end, we first calculate the association between the choice of emojis and the types of issues, using the pointwise mutual information (PMI) between emojis and issue types (Church and Hanks 1990). For each emoji e, the PMI value for the type of issue i is $\text{PMI}(e, i) = \log \frac{p(e,i)}{p(e)p(i)}$.

A positive PMI indicates the positive association between the emoji and the issue type. The top 10 most frequently used emojis with positive PMI for each issue type are shown in Table 7. We find that bug issues are positively associated with emojis with domain specific meanings such as 🦠, 🐞 and ✔️. Positive PMI emojis for question issues are mostly emotions and facial expressions. For feature issues, nearly half of the emojis are domain-specific (🚀, 💰 and 📐) and half express emotions.

Now that we have identified emojis associated with each issue type, we repeat the estimation of the HTE but restrict to issues with the type-associated emojis. We report the HTE of using type-associated emojis on the right side of Table 6. Compared to the HTE estimated on all issues, the HTE on type-associated emojis is in the same direction but at larger scales. For bug and question issues, type-associated emojis can further reduce the resolution time. For example, the increase in the proportion of issues closed in 180 days increases from 0.031 to 0.053 and from 0.024 to 0.036 for bug issues and question issues, respectively. Similarly, for feature issues, type-associated emojis promote more active

Outcome Variable	ATE	Heterogeneous Treatment Effect (HTE)			HTE on type-associated emojis		
		Bug	Question	Feature	Bug	Question	Feature
getting comments	0.041**	0.052**	-0.019	0.037**	0.060**	0.011	0.030**
# of comments	0.118**	-0.013	-0.128*	0.225**	-0.224**	-0.080	0.301**
# of comment users	0.113**	0.042	-0.059*	0.208**	-0.013	0.010	0.264**
% closed in 180 days	0.033**	0.031**	0.024*	0.007	0.053**	0.036**	0.005
% closed in 90 days	0.033**	0.033**	0.027*	0.007	0.053**	0.034**	0.008
% closed in 60 days	0.035**	0.036**	0.030*	0.009	0.054**	0.036**	0.014
% closed in 30 days	0.038**	0.044**	0.026**	0.009	0.060**	0.039**	0.008
issue closing time (day)	-1.751**	-2.673**	-0.378	-1.313	-2.688**	-0.470	-0.998
# Issues in Treatment Group	14,686	5,201	4,227	6,441	1,784	2,230	3,445

Table 6: The treatment effect of using all emojis and positive PMI emojis in GitHub issues per issue types. The ATE column is the same with Table 4 and is included here for reference. Significance level: ** $p < 0.01$, * $p < 0.05$, paired z-test.

Issue Type	10 Most Frequently Used Emojis
Bug	🔴🍾⚠️👹✔️❌🔲💬🙅🤖
Question	👍😄👋🤔🙂❤️😅😊🙏😃
Feature	👍😄🚀🎉✅✨❤️😊💰📐

Table 7: 10 Most frequently used emojis with positive PMI for each issue type.

participation, as the treatment effect on the number of comments increases from 0.225 to 0.301. These findings show that the treatment effect is larger when authors choose emojis that signals the type of the issues.

The analysis of HTE on type-associated emojis presents only preliminary findings about social signals as a possible mechanism that explains causal effect of using emojis. Future efforts are needed to fully examine emojis as social signals. Other possible explanations may also hold. For example, one known functionality of emojis is adjusting the tones and sentiments (Hu et al. 2017; Cramer, de Juan, and Tetreault 2016), and there also exists the correlation between sentiments and issue outcomes (Sanei, Cheng, and Adams 2021; Murgia et al. 2014; Ortu et al. 2015). It is possible that emojis first adjust the affective states (sentiments and tones) in issues, and the adjusted affective states bring more attention, which leads to more participation.

A Case Study

To better understand how emojis apply functions, we conduct a case study on 3 selected issues, where authors use emojis. Table 8 lists the titles of the issue, the bodies of the issue, and the attached comments. All the issues are closed and are automatically labeled as the question, feature and bug issue respectively.

In the first question issue, the authors use 😔 (pensive face) in the issue body. From the sentiment adjustment aspect, 😔 shows the author's disappointment and unhappiness emotion. If removing this emoji, the overall sentiment of the issues tends to be neutral. The emoji also indicates the author's frustration towards the inaccessibility to discord, and the hidden anxious attitude may nudge the repository owners to answer the question. From the comment below this issue, the developer notices the authors' attitude and offers a solution, so emoji promotes the issue resolution.

Issue Title	Hello
Issue Body	@[user] I currently can't access discord because I don't have a phone number to verify that I'm a real human. 😔
Comments	Use that free SMS service

(a) A question issue and the developers' comments. This issue is closed in 1 day and contains 1 comment below.

Issue Title	Add Portainer software docs
Issue Body	[pull request url] Need to add Portainer docs before 6.34 release 😉
Comments	Ok linking the page as a basic help looks good for me. :-) [url] well usually we don't add fill software docs. More basic thinks like how to access. But we could link this guide or the official docs

(b) A feature issue and the developers' comments. This issue is closed in 12 days and contains 3 comments below.

Issue Title	Docs readme typo
Issue Body	❌ 'index.js' ✔️ 'index.html'
Comments	Thanks for pointing that out, [at mention] .

(c) A bug issue and the developers' comments. This issue is closed in 0 days and contains 1 comment below.

Table 8: Issues examples with emojis. Comment number and closing days are listed below each sub-table.

For the second feature issue, the author proposes a new pull request to add a new feature and applies 😉 to the issue body. This emoji suggests that the author is satisfied with the new feature and the emoji makes the issue sentiments to be positive. Again, from the three comments below, the other developers receive the information in the signal and

appreciate the author's contribution.

For the third bug issue, the issue is about to fix a typo, and the author only uses two emojis to signal where to change. Although unlike two issues above, the semantics of this bug issue can also be expressed by words, the emojis make the issue more concise and formalized, signaling that the author would like to save the other developers' time, which shows consideration for readers' comfort. This issue is closed immediately, and the typo is also fixed.

Limitations and Implications

There are several limitations for our work. First, a major threat to validity for using propensity score is the unobserved confounders. Due to the limitations of the dataset, we cannot fully capture all covariates of an author or repository. However, with the results of Section *Sensitivity Analysis*, the simultaneous sensitivity analysis has demonstrated that our measured causal effect is robust to the strong unobserved confounder.

Although we have conducted qualitative studies and identified social signaling as a possible mechanism of the causal effect of emojis, it is not yet verified whether such signal can be widely accepted by other developers. Such questions would be best answered with a user study, such as a structured interview, a survey, or an randomized controlled experiment, which complements this study from the receiver side of the communication. We may expect the user study to verify whether readers can perceive emojis in issues as social signals, and be encouraged to participate or help resolving the issue.

Emojis are widely considered a ubiquitous language across languages. Our work only focuses on the GitHub English issues, and for the future work, we can expand our experiments to other languages or even other domain-specific communities to make our conclusion more generalizable.

Results from the causal inference provide new insights into the relationship between issue contents, emoji usage, and reader response on GitHub. We discuss the implications of our work below and hope that our work can motivate future research.

Our research reveals the causality between emoji usage and issue outcomes. We expect that authors can be motivated to use more emojis in issues, and increased efficiency of issue resolution can also attract more developers to the platform. In addition to putting more emojis in the issues, our results in Section *Heterogeneous Effects by Issue Types* also benefit the authors in deciding which emojis to add according to the issue types. Authors may choose emojis that signal the type and content of the issue. For example, if writing an issue about raising bugs, authors can put more emojis with domain-specific semantics, such as 🛑 (stop sign) or 🐾 (bug) to attract other developers' attention. For workplaces that would like to remain hybrid or remote, our work provides empirical evidence that adopting emojis could be a step to improve working efficiency.

Conclusion

In this work, we present the first empirical study of the causality between emoji usage and GitHub issue outcomes. To this end, we construct a dataset of more than 200K GitHub issues, as well as the data of potential confounders from the issue texts, the issue author, and repository information. We use propensity score matching to quantify causality and multivariate logistic regression to estimate the propensity score. We show significant effects of emoji usage on issues from two perspectives: issue participation and resolution. In the fine-grained analysis, we show the heterogeneous treatment effects by issue types, and to understand causality, we also explore HTE on type-associated emojis. More salient treatment effects caused by type-associated emojis preliminarily verify our hypothesis that emojis, as social signals, reduce information asymmetry to promote user participation and issue resolution. The promising results suggest that developers can add more emojis to help resolve their issues and attract more users to participate in discussions.

Broader Perspective and Ethical Considerations

The causal effects of emojis in increasing developer participation and resolving issues suggest that emojis can have a positive impact on improving work outcomes in online workspace, which may motivate researchers and practitioners to evaluate if the conclusion can be generalized to the real-life workspace to help establish more effective remote working. Although there is a dramatic increase in emoji usage on GitHub, the use of emojis is still much lower than on social media platforms, such as Twitter. GitHub website can encourage or guide users to use more emojis in issues. We have thoroughly discussed the positive and negative outcomes of our results in Section *Limitations and Implications*. In our work, the data in the GitHub issue dataset are all from a third-party public project, GHTorrent, and we do not collect or release other new datasets. During data preprocessing, we strictly follow ethical principles and do not attempt to infer the identities of the issue authors.

Acknowledgements

We would like to thank anonymous reviewers as well as Vanessa Frias-Martinez and Paiheng Xu for reviewing the paper and for providing helpful comments and suggestions.

References

Ai, W.; Lu, X.; Liu, X.; Wang, N.; Huang, G.; and Mei, Q. 2017. Untangling emoji popularity through semantic embeddings. In *ICWSM 2017*.

Angrist, J. D. 2008. *Mostly Harmless Econometrics: An Empiricistís Companion*.

Athey, S.; and Imbens, G. W. 2017. The state of applied econometrics: Causality and policy evaluation. *Journal of Economic perspectives*.

Austin, P. C. 2011. Optimal caliper widths for propensity-score matching when estimating differences in means and

differences in proportions in observational studies. *Pharmaceutical statistics*.

Austin, P. C.; Yu, A. Y. X.; Vyas, M. V.; and Kapral, M. K. 2021. Applying Propensity Score Methods in Clinical Research in Neurology. *Neurology*.

Barbieri, F.; Kruszewski, G.; Ronzano, F.; and Saggion, H. 2016. How Cosmopolitan Are Emojis? Exploring Emojis Usage and Meaning over Different Languages with Distributional Semantics. In *ACM-MM 2016*.

Blanchard, A. L. 2021. The effects of COVID-19 on virtual working within online groups. *Group Processes & Intergroup Relations*.

Bland, J. M.; and Altman, D. G. 2000. The odds ratio. *Bmj*.

Blei, D. M.; Ng, A. Y.; and Jordan, M. I. 2003. Latent dirichlet allocation. *Journal of machine Learning research*.

Chen, Z.; Cao, Y.; Yao, H.; Lu, X.; Peng, X.; Mei, H.; and Liu, X. 2021. Emoji-powered sentiment and emotion detection from software developers' communication data. *TOSEM*.

Chen, Z.; Lu, X.; Ai, W.; Li, H.; Mei, Q.; and Liu, X. 2018. Through a Gender Lens. *WWW 2018*.

Church, K.; and Hanks, P. 1990. Word association norms, mutual information, and lexicography. *CL 1990*.

Cramer, H.; de Juan, P.; and Tetreault, J. 2016. Sender-Intended Functions of Emojis in US Messaging. In *MobileHCI 2016*.

Danescu-Niculescu-Mizil, C.; Sudhof, M.; Jurafsky, D.; Leskovec, J.; and Potts, C. 2013. A computational approach to politeness with application to social factors. *arXiv preprint*.

Derrick, B.; Russ, B.; Toher, D.; and White, P. 2017. Test statistics for the comparison of means for two samples that include both paired and independent observations. *Journal of Modern Applied Statistical Methods*.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Eckles, D.; and Bakshy, E. 2017. Bias and high-dimensional adjustment in observational studies of peer effects. *arXiv preprint arXiv:1706.04692*.

Escouflaire, L. 2021. Signaling irony, displaying politeness, replacing words: The eight linguistic functions of emoji in computer-mediated discourse. *Lingvisticæ Investigationes*.

Feder, A.; Keith, K. A.; Manzoor, E.; Pryzant, R.; Sridhar, D.; Wood-Doughty, Z.; Eisenstein, J.; Grimmer, J.; Reichart, R.; Roberts, M. E.; et al. 2021. Causal Inference in Natural Language Processing: Estimation, Prediction, Interpretation and Beyond. *arXiv preprint arXiv:2109.00725*.

Gastwirth, J. L.; Krieger, A. M.; and Rosenbaum, P. R. 1998. Dual and simultaneous sensitivity analysis for matched pairs. *Biometrika*.

Gu, X. S.; and Rosenbaum, P. R. 1993. Comparison of multivariate matching methods: Structures, distances, and algorithms. *Journal of Computational and Graphical Statistics*.

Hill, J.; and Reiter, J. P. 2006. Interval estimation for treatment effects using propensity score matching. *Statistics in medicine*.

Hu, T.; Guo, H.; Sun, H.; Nguyen, T.-v.; and Luo, J. 2017. Spice up your chat: the intentions and sentiment effects of using emojis. In *ICWSM 2017*.

Kavaler, D.; Sirovica, S.; Hellendoorn, V.; Aranovich, R.; and Filkov, V. 2017. Perceived language complexity in GitHub issue discussions and their effect on issue resolution. In *ASE 2017*.

Liu, W.; Kuramoto, S. J.; and Stuart, E. A. 2013. An introduction to sensitivity analysis for unobserved confounding in nonexperimental prevention research. *Prevention science*.

Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint*.

Lu, X.; Ai, W.; Chen, Z.; Cao, Y.; and Mei, Q. 2022. Emojis predict dropouts of remote workers: An empirical study of emoji usage on GitHub. *PloS one*.

Lu, X.; Ai, W.; Liu, X.; Li, Q.; Wang, N.; Huang, G.; and Mei, Q. 2016. Learning from the Ubiquitous Language: An Empirical Analysis of Emoji Usage of Smartphone Users. In *UbiComp 2016*.

Lu, X.; Ai, W.; Wang, Y.; and Mei, Q. 2023. Team Resilience under Shock: An Empirical Analysis of GitHub Repositories during Early COVID-19 Pandemic. In *ICWSM 2023*.

Lu, X.; Cao, Y.; Chen, Z.; and Liu, X. 2018. A first look at emoji usage on github: An empirical study. *arXiv preprint*.

Manku, G. S.; Jain, A.; and Das Sarma, A. 2007. Detecting near-duplicates for web crawling. In *WWW 2007*.

Mäntylä, M.; Adams, B.; Destefanis, G.; Graziotin, D.; and Ortu, M. 2016. Mining valence, arousal, and dominance: possibilities for detecting burnout and productivity? In *MSR 2016*.

Merchant, A.; Shah, D.; Bhatia, G. S.; Ghosh, A.; and Kumaraguru, P. 2019. Signals matter: understanding popularity and impact of users on stack overflow. In *WWW 2019*.

Miller, H.; Kluver, D.; Thebault-Spieker, J.; Terveen, L.; and Hecht, B. 2017. Understanding emoji ambiguity in context: The role of text in emoji-related miscommunication. In *ICWSM 2017*.

Murgia, A.; Tourani, P.; Adams, B.; and Ortu, M. 2014. Do developers feel emotions? an exploratory analysis of emotions in software artifacts. In *MSR 2014*.

Novielli, N.; and Serebrenik, A. 2019. Sentiment and emotion in software engineering. *IEEE Software*.

Ortu, M.; Adams, B.; Destefanis, G.; Tourani, P.; Marchesi, M.; and Tonelli, R. 2015. Are bullies more productive? Empirical study of affectiveness vs. issue fixing time. In *MSR 2015*.

Pearl, J. 1995. Causal diagrams for empirical research. *Biometrika*.

Roberston, A.; Magdy, W.; and Goldwater, S. 2021. Black or White but never neutral: How readers perceive identity from yellow or skin-toned emoji. *CSCW 2021*.

Rong, S.; Wang, W.; Mannan, U. A.; de Almeida, E. S.; Zhou, S.; and Ahmed, I. 2022. An empirical study of emoji use in software development communication. *Information and Software Technology*.

Rosenbaum, P. R.; and Rubin, D. B. 1983. The central role of the propensity score in observational studies for causal effects. *Biometrika*.

Sanei, A.; Cheng, J.; and Adams, B. 2021. The Impacts of Sentiments and Tones in Community-Generated Issue Discussions. In *CHASE 2021*.

Spence, M. 2002. Signaling in retrospect and the informational structure of markets. *American Economic Review*.

Stuart, E. A.; Lee, B. K.; and Leacy, F. P. 2013. Prognostic score–based balance measures can be a useful diagnostic for propensity score methods in comparative effectiveness research. *Journal of clinical epidemiology*.

Tauch, C.; and Kanjo, E. 2016. The Roles of Emojis in Mobile Phone Notifications. In *UbiComp 2016*.

Trockman, A.; Zhou, S.; Kästner, C.; and Vasilescu, B. 2018. Adding sparkle to social coding: an empirical study of repository badges in the npm ecosystem. In *ICSE 2018*.

Tsay, J.; Dabbish, L.; and Herbsleb, J. 2014. Influence of social and technical factors for evaluating contribution in GitHub. In *ICSE 2014*.

Zhang, Z.; Kim, H. J.; Lonjon, G.; Zhu, Y.; et al. 2019. Balance diagnostics after propensity score matching. *Annals of translational medicine*.

Zhou, Y.; and Ai, W. 2022. #Emoji: A Study on the Association between Emojis and Hashtags on Twitter. In *ICWSM 2022*.

Zhou, Y.; Lu, X.; and Ai, W. 2024. From Adoption to Adaption: Tracing the Diffusion of New Emojis on Twitter. *arXiv preprint*.

Zhou, Y.; Xu, P.; Wang, X.; Lu, X.; Gao, G.; and Ai, W. 2024. Emojis decoded: Leveraging chatgpt for enhanced understanding in social media communications. *arXiv preprint*.

Paper Checklist

1. For most authors...
   (a) Would answering this research question advance science without violating social contracts, such as violating privacy norms, perpetuating unfair profiling, exacerbating the socio-economic divide, or implying disrespect to societies or cultures? Yes
   (b) Do your main claims in the abstract and introduction accurately reflect the paper's contributions and scope? Yes, see Section *Abstract* and *Introduction*
   (c) Do you clarify how the proposed methodological approach is appropriate for the claims made? Yes, see Section *GitHub Issues and Problem Formulation*, *Collection of Confounders* and *Causal Inference of Emojis on Issues*
   (d) Do you clarify what are possible artifacts in the data used, given population-specific distributions? No, the GitHub issues do not have demographic information
   (e) Did you describe the limitations of your work? Yes, see Section *Limitations and Implications*
   (f) Did you discuss any potential negative societal impacts of your work? Yes, see Section *Limitations and Implications*
   (g) Did you discuss any potential misuse of your work? No, we do not have a major concern about the potential misuse of our work
   (h) Did you describe steps taken to prevent or mitigate potential negative outcomes of the research, such as data and model documentation, data anonymization, responsible release, access control, and the reproducibility of findings? No, we do not have a major concern about the potential misuse
   (i) Have you read the ethics review guidelines and ensured that your paper conforms to them? Yes
2. Additionally, if your study involves hypotheses testing...
   (a) Did you clearly state the assumptions underlying all theoretical results? Yes, see Section *GitHub Issues and Problem Formulation*, *Collection of Confounders* and *Causal Inference of Emojis on Issues*
   (b) Have you provided justifications for all theoretical results? Yes, see Section *GitHub Issues and Problem Formulation*, *Collection of Confounders*, *Causal Inference of Emojis on Issues* and *Results and Discussion*
   (c) Did you discuss competing hypotheses or theories that might challenge or complement your theoretical results? Yes, see Section *GitHub Issues and Problem Formulation*, *Collection of Confounders*, *Causal Inference of Emojis on Issues* and *Results and Discussion*
   (d) Have you considered alternative mechanisms or explanations that might account for the same outcomes observed in your study? No
   (e) Did you address potential biases or limitations in your theoretical framework? Yes, see Section *Results and Discussion*
   (f) Have you related your theoretical results to the existing literature in social science? Yes, see Section *Related Work* and *Limitations and Implications*
   (g) Did you discuss the implications of your theoretical results for policy, practice, or further research in the social science domain? Yes, see Section *Limitations and Implications*
3. Additionally, if you are including theoretical proofs...
   (a) Did you state the full set of assumptions of all theoretical results? Yes, see Section *GitHub Issues and Problem Formulation*
   (b) Did you include complete proofs of all theoretical results? NA
4. Additionally, if you ran machine learning experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? Yes,

see Section *Collection of Confounders*, *Causal Inference of Emojis on Issues* and *Results and Discussion*

(b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? Yes, see Section *Collection of Confounders* and *Causal Inference of Emojis on Issues*

(c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? No, the variation is small

(d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? No

(e) Do you justify how the proposed evaluation is sufficient and appropriate to the claims made? Yes, see Section *Collection of Confounders* and *Causal Inference of Emojis on Issues*

(f) Do you discuss what is "the cost" of misclassification and fault (in)tolerance? Yes, see Section *Collection of Confounders*, *Causal Inference of Emojis on Issues* and *Results and Discussion*

5. Additionally, if you are using existing assets (e.g., code, data, models) or curating/releasing new assets, **without compromising anonymity**...

(a) If your work uses existing assets, did you cite the creators? Yes, see Section *Collection of Confounders* and *Causal Inference of Emojis on Issues*

(b) Did you mention the license of the assets? No

(c) Did you include any new assets in the supplemental material or as a URL? No

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? No

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? No

(f) If you are curating or releasing new datasets, did you discuss how you intend to make your datasets FAIR? NA

(g) If you are curating or releasing new datasets, did you create a Datasheet for the Dataset? NA

6. Additionally, if you used crowdsourcing or conducted research with human subjects, **without compromising anonymity**...

(a) Did you include the full text of instructions given to participants and screenshots? NA

(b) Did you describe any potential participant risks, with mentions of Institutional Review Board (IRB) approvals? NA

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? NA

(d) Did you discuss how data is stored, shared, and deidentified? NA

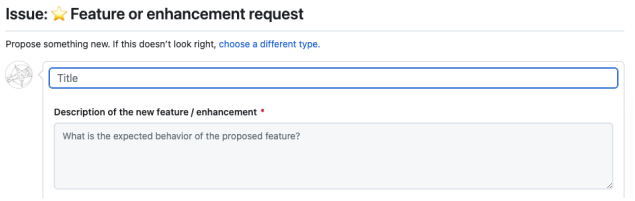

Figure 4: An issue template for feature request in the microsoft/PowerToys repository

Appendix

Identifying Issue Templates and Bots

To formalize the format of GitHub issues, a large number of repository owners set issue templates and force issue authors to use the predefined templates. For example, as shown in Figure 4, the repository, microsoft/Powertoys,[5] asks developers to use this template when composing issues related to the feature request. The emoji ⭐ (star) in this template are not added by the issue authors but by the repository owners. We remove all the issues in the template-applied repositories from our dataset by checking whether the repository contains "ISSUE_TEMPLATE" folder.

In practical issue usage, a small proportion of repository owners regard issue discussions as their development logs. Owners post their development progress as new issues and close this kind of issues in a short time. Owner-created issues to record progress are not the focus of our project, so we use two criteria to remove them. We first remove the repositories containing more than 10 issues and 90% issues written by the same author. The second is to remove the issues composed by the repository owners but closed within one day.

Since GitHub is a code-communication platform, many developers choose GitHub as the first community to test their newly-developed chat bots. The bot-created issues significantly bias the causal effect estimation of emojis. First, we use simhash for near-duplicate detection of issues (Manku, Jain, and Das Sarma 2007) with the output of an unsigned 64-bit integer, and if there are less than 5 difference digits, two issues are regarded as near-duplicate. We filter the authors with more than 100 issues, and more than 90% issues of the author are near-duplicate.

Issue Type Classifier

We extract 160,000 and 20,000 labeled issues as the training and test dataset respectively. For each issue type, we train a binary Roberta (Liu et al. 2019) classifier to identify whether the issue belongs to this type. When preparing the training positive instances, we first normalize the issue labels to increase the number of positive data. For example, we treat every label with the "Bug" substring (except "Not-Bug") as a bug label. The three trained classifiers can achieve 89.3%, 89.21% and 87.00% in the bug, feature, and question test sets, respectively. After automatically annotating with these three classifiers, we obtain three labels to indicate whether the issue is bug-related, question-related, or feature-related.

[5] https://github.com/microsoft/PowerToys/issues/new/choose

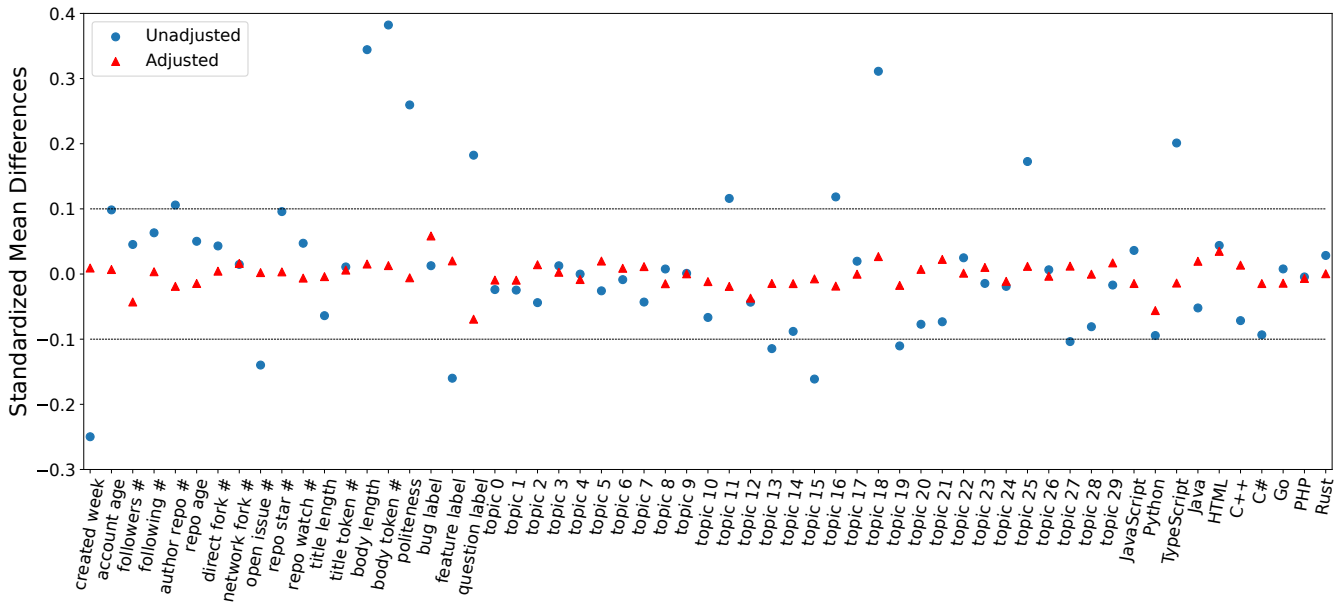

Figure 5: SMD value before matching and after matching for each covariate. After the matching process, the SMD value of all covariates are smaller than 0.1, indicating the similar distribution of all covariates between treatment and control group.

Outcome Variable	ATE	Obs. Δ	Avg.
Developer Participation			
getting comments	0.075**	0.219	0.687
# of comments	0.101**	0.827	2.124
# of comment users	0.162**	0.596	1.366
Issue Resolution			
% closed in 180 days	0.020**	0.062	0.500
% closed in 90 days	0.021**	0.055	0.475
% closed in 60 days	0.021**	0.054	0.458
% closed in 30 days	0.022**	0.054	0.426
issue closing time (day)	-1.621**	-0.079	15.441

Table 9: ATE of using emojis in GitHub issues from July 2018 to June 2019. The meaning of notation is the same as notations in Table 4. Significance level: ** $p < 0.01$, * $p < 0.05$, paired z-test.

SMD Balance Check and Refinement

SMD is defined as: $SMD = \frac{\overline{X_1} - \overline{X_2}}{\sqrt{(S_1^2 + S_2^2)/2}}$, where $\overline{X_1}$ and $\overline{X_2}$ are the sample mean for the treated and control groups, respectively. S_1^2 and S_1^2 are sample variances for two groups.

We calculate the SMD value and the result show that SMD value for confounder "# of characters in the issue body" (issue body length) is greater than 0.1. The SMD value of issue body length before matching is 0.344, which indicates that this confounder distribution has a large gap between treatment and control groups and only matching on the propensity score can not guarantee the balanced distribution.

To make the confounders more balanced, we make another refinement to the NNM method: the difference of issue body length in the matched pairs should not exceed 1 SD of the issue body length. For the matching procedure, we first filter the untreated issues that exceed the restriction of the propensity score and the body length of the issue and then find the issue with the closest propensity score. The SMD values of each confounder covariate before and after the matching are visualized in Figure 5. With the refinement, SMD values of all confounders are below 0.1 threshold.

Causal Inference Reproduction

Reproduction on a different year

We re-collect the GitHub issues in public repositories between July 1, 2018 and June 30, 2019 via GHTorrent and sample with a ratio of 1:20. We repeat the data preprocessing and obtain a new dataset with 361,257 issues where 17,630 issues contain one or more emojis. After propensity score estimation and conducting a NNM, we calculate the average treatment and report the results in Table 9. ATE values in Table 9 are still significant and comply with the findings in the causal effect for issues from 2020 to 2021.

Reproduction on various covariate specifications

To verify our results' robustness to misspecifications of the covariates, we repeat our analyses on three alternative specifications. In the first two alternatives, we set the numbers of topics as 45 and 15, re-train the LDA model, and repeat the analyses. In the third alternative specification, we remove the politeness score from the propensity score estimation and show all the values in Table 10. We observe that the estimated treatment effects under the three alternative specifications all share a similar size and significance level as our main result, which indicates that our findings are robust to misspecification in estimating the covariates.

Outcome Variable	ATE (45 topics)	ATE (15 topics)	ATE (no politeness)
Developer Participation			
getting comments	0.041**	0.046**	0.045**
# of comments	0.127**	0.098**	0.111**
# of comment users	0.117**	0.127**	0.122**
Issue Resolution			
% closed in 180 days	0.036**	0.033**	0.035**
% closed in 90 days	0.035**	0.031**	0.034**
% closed in 60 days	0.039**	0.035**	0.036**
% closed in 30 days	0.040**	0.034**	0.036**
issue closing time (day)	-1.961**	-1.765**	-1.730**

Table 10: Average treatment effect of using emojis in issues from June 2020 to June 2021 with 45-dimensional topic distribution, with 15-dimensional topic distribution, and without politeness score in propensity score estimation. The meaning of notation is the same as the notation in Table 4. Significance level: ** $p < 0.01$, * $p < 0.05$, paired z-test.

Outcome Variable	ATE
Developer Participation	
getting comments	0.033**
# of comments	0.094**
# of comment users	0.089**
Issue Resolution	
% closed in 180 days	0.022**
% closed in 90 days	0.023**
% closed in 60 days	0.025**
% closed in 30 days	0.025**
issue closing time (day)	-2.556**

Table 11: Average treatment effect of using emojis in GitHub issues from June 2020 to June 2021 with GBRT model as propensity score estimation method. Significance level: ** $p < 0.01$, * $p < 0.05$, paired z-test.

Reproduction on GBRT model

To test our findings' sensitivity to the model of propensity score estimation, we repeat the analysis by using the Gradient Boosted Regression Tree (GBRT) model for propensity score estimation. We report the estimates in Table 11. The treatment effects are similar to our main result in effect size and significance level, which suggests the robustness of our estimation to the model of propensity score estimation.