Decoding the Trends: A Comparative Analysis of Software Energy Efficiency Discussions

OVIDIU BURAC, University of Twente, The Netherlands

Energy Consumption has become a critical software design consideration given the importance of the mobile device and data center platforms. However, research into the needs, problems, solutions, and interest software engineers have regarding energy-efficiency is still limited and the rapid changes of the industry are not yet adequately reflected in current research on the topic. The goal of this paper is to provide an analysis on the topic of energy-related concerns of software developers, to reflect upon how the relevancy of this topic has evolved through time and to contrast the most up to date findings with the results and expectations of relevant literature, using data extracted from Q&A platform Stack Overflow by searching for specific energy-consumption-related terms. Using numeric features associated with post popularity, the interest in the topic of energy-efficiency was found to be smaller than average for Stack Overflow. Utilizing tag frequency, the correlation between the software energy efficiency topic and mobile, desktop and embedded platforms was analyzed. The findings were contrasted with results of relevant literature. We hope that the techniques used for analysis, the comparison between found results and relevant literature alongside the proposed hypothesis for these differences can aid future researchers in investigating these platforms and topics in even further depth.

Additional Key Words and Phrases: Q&A, Stack Overflow, Energy Consumption, Software

1 INTRODUCTION

In the past two decades software systems have undergone radical transformations. The ubiquity of smartphones[17], the proliferation of data centers[12], and the increased computational needs of both users and developers considering the popularization of large language models[21, 33], have further solidified the role of energy efficiency as a software design consideration. Not only has the adoption of mobile and data-center platforms escalated but the context of energy-efficient systems has also shifted given the introduction of new tools, libraries, languages, state of the art developments in the IT field[17] and the rapidly developing Machine Learning field[26], especially in the context of battery-powered devices[5, 2, 16]. As such, we believe it imperative that these energy-related needs, problems and solutions of developers are thoroughly examined and understood.

This study aims to update and expand upon previous StackOverflow research[23, 3, 25] by analysing the popularity of the energy-efficiency topic, the features of relevant posts and the changes of these aspects through time. This will be accomplished by investigating the features of questions relevant to the topic and the interest in energy-related questions. For this goal we will use data from software development Q&A website Stack Overflow, a platform often used in software engineering studies as a representative of the larger software development community. The consistent popularity of Stack Overflow as a learning and feedback tool throughout the last decade makes it a prime candidate for empirical studies regarding the challenges encountered by software engineers and the developments of said challenges year by year.

Comprehensive research regarding the specific topic of software energy consumption through the lens of the software engineering community is readily available[20, 15, 23]. However, this literature is approximately a decade old. Significant developments in the IT domain have happened in that time frame, such as the massively growing popularity of mobile operating systems, the decreasing utilization of Java as a programming language, et cetera, as described by the findings of Moutidis and colleagues[17]. Thus, the resulting changes in the field of programming must be reflected in newer data and analysis. Considering that Stack Overflow answers rarely get updated in light of new research or developments, as concurred by Zhang et al.[36], the relevancy and accuracy of the answers can decrease over time in light of new developments. This suggests the importance of updated analysis and interpretations given the same data sets. The presence of these outdated answers on StackOverflow thus allows for thorough comparative analysis of ways in which solutions to similar energy-related problems have evolved since the launch of Stack Overflow in 2008 and the latest Stack Overflow data dump (April 2nd 2024 at the time of writing).

This paper aims to establish an overview of the current landscape of energy-related problems(I), the interest software developers have in the subject matter(II) and a thorough analysis of the ways in which said topic developed since the launch of Stack Overflow(III), updating and expanding upon existing literature.

1.1 Research Questions

In order to further refine the scope and goals of this study the following research questions were proposed:

- **Q1**: What features characterize energy-related problems?
- **Q2**: What platforms and conditions are typically associated with energy-related problems?
- **Q3**: How has interest in software energy efficiency changed from the launch of Stack Overflow in 2008 to 2024?

2 RELATED WORK

To gather appropriate literature we have used Google Scholar and ACM using terms such as "Stack Overflow", "software energy efficiency", "software energy consumption", etc. Additionally, Inciteful.xyz was used to find literature related to the group of papers gathered initially through keyword search and also literature pieces connecting any two papers found.

2.1 Software Energy Efficiency Studies

We have found that the topic of software energy efficiency has been investigated using numerous approaches. In some instances this was

TScIT 41, July 5, 2024, Enschede, The Netherlands

done using the experiences of practitioners, by leveraging surveys as performed by Pang et al.[20] or Manotas and colleagues[15], to gather insight into the knowledge and perspectives of practitioners regarding energy-efficient software engineering. These studies suggest that, while developers do consider the energy-efficiency of software systems as a valuable consideration, the limited research and literature regarding the subject often leads to misconceptions and misunderstandings of the topic and appropriate solutions.

We additionally investigated several papers studying various techniques aimed at improving software energy consumption. The main premise of the examined papers was modifying existing software and comparing the effectiveness of state of the art methods. Some of the analyzed methods include: refactoring, investigated by Şanlıalp et al.[27], virtualization, studied by Katal et al.[12], utilization of specific collections, researched by Oliviera and colleagues[19], software configurations, examined by Weber et al.[34] and thread management, as explored by Pinto et al.[24]. These studies provide valuable insight into some of the solutions valuable in developing energy-conscious software solutions and their effectiveness.

2.2 Automated Text Analysis Studies

Given the relevancy of text mining techniques when it comes to larger data-scale studies such as those performed on StackOverflow, literature analysing these techniques is very relevant to researchers. A number of research papers have studied the popularity of Latent Dirichlet Allocation (LDA) techniques, an automated generative topic modelling approach, in order to discern themes in extensive text corpora[1, 28]. For instance, a study conducted by Silva and colleagues[28] has examined the ways in which LDA techniques are used, the topics most commonly modelled, and how they are documented across software engineering literature, concurring the popularity of this technique in published papers of journals and conferences.

2.3 StackOverflow Studies

The relevancy of Q&A forums as both a learning and troubleshooting tool has been thoroughly analyzed by various papers. The analysis of Kabir et al.[11] compared the effectiveness of Stack Overflow as a feedback and troubleshooting tool to current alternatives such as ChatGPT and deemed it significantly more useful in aiding software engineers due to correctness. Additionally, the findings of Dondio and colleagues[6] concur the relevancy of Stack Overflow as learning tool when compared to traditional learning methods.

Different studies focused on extracting the topics and interests of developers by using StackOverflow, given the quantity of literature suggesting it's relevance for developers. For instance, Barua et al.[3] focused on utilizing LDA topic modelling on post bodies in order to extract, categorize and analyze different topics of StackOverflow, finding distinct topics such as mobile development, web development, etc. The analysis of various popular concerns of battery-powered platforms, such as "app distribution", "connectivity" and "tools", was investigated Rosen and colleagues[25]. A different study by Pinto et al.[23], focused on investigating the specific topic of software energy efficiency within mobile, web and desktop development by utilizing various proposed metrics to gauge and analyze the popularity and interest in posts associated with the topic, finding posts associated with software energy efficiency significantly more popular than other StackOverflow posts. A separate study performed by Moutidis et al.[17] focused on investigating the popularity over time for different tag categories, such as programming languages, web frameworks, operating systems, etc. and the popularity of the tags within each category to offer a comprehensive review of the ways in which tag popularity on StackOverflow changed over time. All of these studies found mobile developers to be a significant, distinct community of StackOverflow that changes and evolves over time. The study of Pinto and colleagues[23] delving specifically into the energy-efficiency-related concerns and challenges of (but not limited to) mobile developers and offering significant insight into the topic of software energy efficiency.

3 METHODOLOGY

The posts of Q&A platform Stack Overflow will be used as data for the empirical study. At the time of writing, the most recent Stack Overflow data dump was published April 02, 2024[30], containing 24,101,803 questions and 35,603,624 answers for a total of 59,705,427 posts. While the data of 2024 does not span over the whole year, it was included in the analysis since the popularity and interest metrics proposed are normalized by average yearly metrics for StackOverflow. Each post on the website contains a title, a text body describing the question in further detail and at least 1 tag - a keyword used to categorize the question. The types of questions we are interested in examining, also considered true positives, are questions discussing improvement or concerns related to power, energy or battery efficiency of software applications. In this instance, examples of false positives could refer to questions discussing applications processing energy consumption data and necessitating help in implementing or debugging additional features. Another false positive example could be questions regarding application troubleshooting mentioning power saving features.

First, we investigated the number of questions associated with the tags "battery", "energy", "power-management" and "*-efficiency" tags, finding them to be 873, 236, 476 and 1142, respectively, where "*" is used as a wildcard character, replacing other possible words. However, for each tag, not all questions are necessarily related to software energy efficiency[29, 9, 31, 8]. Additionally, out of the 2673 questions associated with at least one of the tags, there are only 54 posts containing more than one of the investigated tags. Furthermore, selecting a smaller subset of questions by looking for additional energy-efficiency related terms, such as "battery", "energy" or "power", within the bodies of questions associated with these tags severely limits the number of available questions to 412, 188, 41 and 15, respectively. Even by combining tags and body terms not all resulting questions are related to software energy efficiency[4, 14, 10]. These results suggests the lack of a unified energy-efficiency-related tag, tag combination or tag and body term combination that could aid in selecting relevant data while eliminating false positives.

As such, in order to prevent the loss of potentially valuable questions due to tag limitations, we decided to query the body of the posts for search terms related to software energy consumption using a relational database. For data selection, we used a three-phase

approach based on selecting potentially relevant search terms, selection refinement, and data exclusion based on results confirming post irrelevancy.

We tried to extract as many relevant questions as possible using the following query terms: *sav* battery*, *improv* battery*, *conserv* battery*, *optimiz* battery*, *improve energy*, *energy consum* , *energy efficien*, *energy sav*, *save energy*, *improve power*, *power consum*, *power efficien*, *power sa*, *save power*, where there the character "*" is used as a wildcard such that the posts found contain with at least one of the enumerated terms contained anywhere within the body. The terms used in the query were partially inspired by a similar study performed by Pinto et al.[23]. Additional terms were added based on the author's ideas of terms potentially related to software energy efficiency. Some proposed terms, such as *thermal* and *temperature*, did not have any relevant results based on a randomized 100 posts selection so they were not introduced into the selection query. This step yielded 4811 questions and 5973 answers. Utilising a randomized sample of 100 questions, the relevancy of posts was found to be 37%.

The refining stage was comprised of altering the body of the each post to remove code snippets denoted by "<code>...</code>" in order to prevent the selection of results containing search terms exclusively contained within code snippets and not the text body itself. This was followed by querying all conjugations of the verbs "save/improve/conserve/consume/optimize" and the nouns "battery/energy/power", resulting in 2392 questions and 2972 answers. The 5420 excluded results were found to contain 2% relevant results based on a randomized selection of 100 posts.

For the final phase we analyzed term relevancy using randomized samples of 100 posts all containing a particular term. If the results were found to contain less than 2% relevant posts, the term was added to an exclusion query. We thus reached the exclusion query containing the terms: *plot*, *plotting*, *plotted*, *charge*, *charging*, *charged*, *battery mode*, *energy saver*, *battery saver*, *power saver*, *save mode*, *saver mode*, *power mode*, *saving mode*, *model*, *dataframe*, *data frame*, *dataset*, *data set*, *heat dissipation*. Using this exclusion query, we have gathered a total of 1768 questions and 2257 answers with a 66% relevancy based on a randomized 100 post sample. No further term exclusions from this group were performed because of the increasing relevancy of the remaining terms, such as * data* (19% relevant), *energy consumption* (13% relevant), *row* (17% relevant), etc. This result will be used as the primary investigation group, referred to as **Base group**.

An additional step, used in collecting a further refined data set, was gathered in an attempt to automate false positive removal. In order to categorize question as "Relevant" or "Irrelevant" we employed a two-phase approach, including manual categorization of a sample and automatic labelling using OpenAI's ChatGPT LLM. In this process, a total of 291 questions were labelled manually. First, a copy of all questions was given to ChatGPT alongside the structure of columns of the data file. It was consequently used to label each row using it's own interpretation of relevancy or irrelevancy to the topic of software energy efficiency. Then, the manually labelled data set was given for comparison and feature extraction. Using the labelled data set for self-supervised learning, we calculated the labelling accuracy using randomized samples of 100 questions. Thus, we found an achieved accuracy of 81% for relevant labelling (81% of questions labelled "relevant" were found to be relevant) and 68% for irrelevant labelling, with an overall accuracy of 74.5% resulting in 926 questions (with 1142 answers) tagged "Relevant" and 842 questions tagged "Irrelevant" (with 1115 answers). The "Relevant" questions and their answers were selected as the **LLM group**. Out of the 842 "irrelevant" questions, we calculated 32% to be relevant to the topic of software energy efficiency. As such, due to the quantity of relevant questions being mislabelled as irrelevant, estimating a loss of 270 relevant questions (15% of the Base group), it was decided that this group would not be used as the primary investigation group. However, the LLM group was subjected to empirical analysis for comparison with the Base group, with the associated comparisons and additional challenges being illustrated in the **Discussion** section.

Once the Base group data was collected we use a suite of empirical tests including analysis of normalized post count, success and popularity metrics, tag popularity, etc. to investigate the features of energy-efficiency related questions and answers.

For the **Tags** analysis of the Base group, additional tags exclusions were performed by utilizing a list of stopwords, defined as commonly used words that are deemed insignificant in the context of this study. The stopwords were mostly comprised of stopwords from the following popular Python natural-language processing libraries: spaCy, NLTK, Gensim. We additionally added several stopwords specifically related to Q&A platforms ("question" and "problem"), code-snippets and several terms of our query. The reason for this is to exclude terms associated with code snippets or words specific to our selection query and thus facilitate investigation of the platform-related and condition-related tags associated with the posts. The full list of stopwords is available as part of the data in Appendix A.1. Several interesting stopwords that were removed from tag analysis are:

Software code-specific stopwords: service, set, user, void, int, public, private, run, running, work, save, mode, intent, class, return, file, function, case, true, string, false, start, read, values, long, add, working, change, method, event, import, usage, state, question, problem, app, application, device, devices, code

Query-related stopwords: battery, power, time, consumption, data, energy

The next step was to use the data mining toolbox Orange3 to perform LDA topic modelling on the Base group. Unfortunately the maximal topic coherence achieved was 47% so these results were not used in the analysis but will be documented in the **Discussions** sections.

4 EMPIRICAL ANALYSIS

We first assessed several quantitative metrics related to our Base group of posts, focusing on changes in the number of posts per year to gauge the interest towards energy consumption over time. As depicted in Figure 1, the distribution of questions and answers throughout the observed period is illustrated. Each data point showcases the number of questions or answers for the respective year.

The Base group data suggests that the number of posts related to the topic of energy efficiency increased steadily starting with the

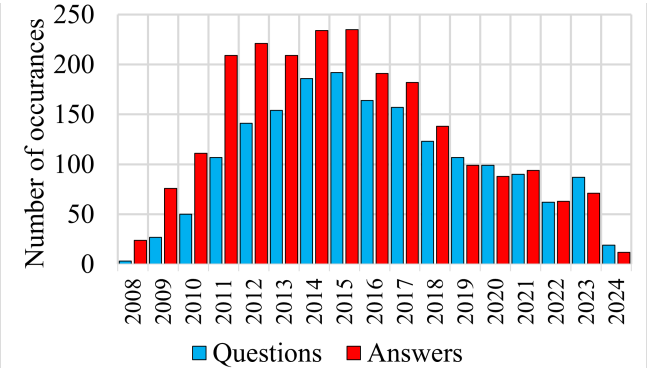

Fig. 1. Questions and answers, from the Base group, per year

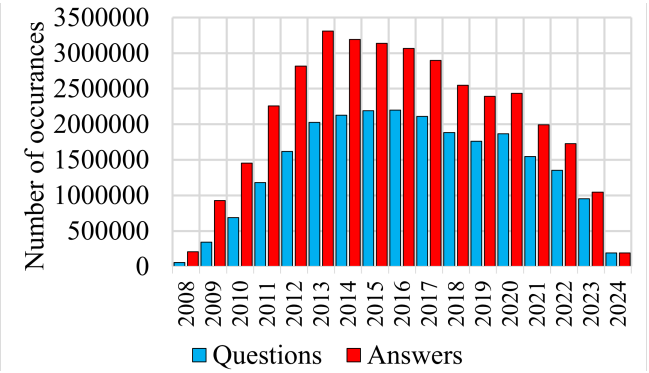

Fig. 2. Questions and answers, from StackOverflow, per year

launch of StackOverflow in 2008, with a 96% increase in the number of new posts from 2010 to 2011 and a peak in 2015 corresponding to a 35% increase when compared to 2011. The number of Base group questions and answers continuously declines post 2015 coming to a minimum of 29% in 2022 compared to 2015, followed by a 26% increase from 2022 to 2023.

Interestingly, the number of new posts on StackOverflow also changed through the years, increasing steadily from 2008 to 2013, with a maximum of 5,336,590 new posts in 2013, and gradually declining to a minimum of 37% in 2023 compared to 2013. As such, a better representative measure of interest in the topic would be comparing the number posts of the Base group divided by the number of questions and answers of StackOverflow for that year.

4.1 How has interest in software energy efficiency changed since the launch of Stack Overflow?

This result can be observed in Figure 3 and suggests a much more consistent interest in the topic between 2008 and 2015 with a maximum in 2011 and a gradual decline to a minimum of 45% in 2022 when compared to the number of posts in 2011. Curiously, there is an increase of 93% in 2023 compared to 2022, likely due to the 26% increase in the number of Base group posts in the same period as the 35% decrease in the number of new StackOverflow posts.

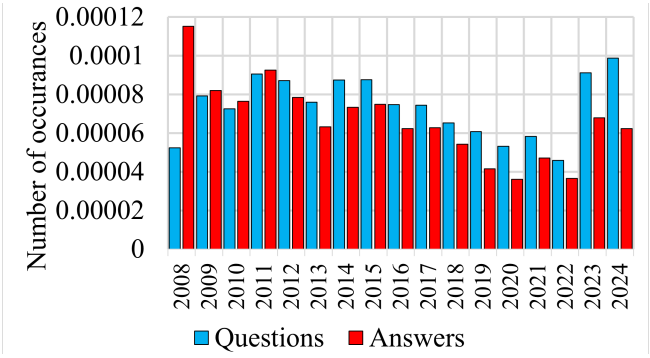

Fig. 3. Normalized count of questions and answers, from the Base group, per year

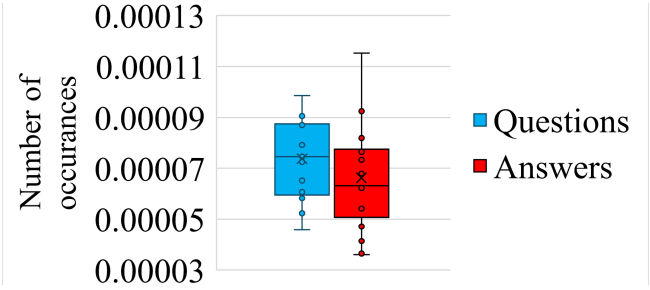

Fig. 4. Box Plot of normalized count of questions and answers, from the Base group

Using the box plot in Figure 4 we can observe the normalized counts of questions and answers of the Base group. The results of 2023, like all others, are still within the 1.5*IQR of Q1 and Q3, suggesting no outliers. Still, it is worth noting that the normalized questions counts of 2008, 2021 and 2022 fall under the Q1 value of 5.89E-05 while the 2011 and 2023 normalized question counts fall past the Q3 value of 8.73E-05. On the other hand, when it comes to the normalized answer count, 2019, 2020, 2021 and 2022 had values under the limit of Q1 (4.89E-05) while 2008, 2009, 2011 and 2012 had values higher than that of Q3 (7.79E-05).

While the normalized counts of questions and answers suggest no outliers, we still believe it worthwhile to further investigate the distribution of answers per question of the Base group. As such, the table of distinct answer counts of the Base group and the number of questions containing said answer count is given in Table 1. Additionally, the table contains the average question and answer popularity for all questions and answers associated with the respective answer count.

Observing the counts of answers, it is worth noting that most questions have less than 2 answers, with a mean of 1.28, a first quartile value of 1 and a 3rd quartile value of 2. Thus, according to the 1.5*IQR rule, all questions with an answer count above 3 are considered outliers. There is a total of 72 (4.1%) outlier questions, these questions encompassing 429 answers (19%) in total. These outlier questions have an average popularity score of 12.69 and an average answer popularity score of 1.83, which is 217% higher than

Table 1. Answer counts, number of posts associated with given answer count and respective question and answer popularity, from Base group

Answer count	**Nr. of posts**	$\mathbb{P}$	$\mathbb{PA}$
17	1	6.92	0.42
15	1	15.93	4.86
14	1	33.75	3.65
13	1	10.49	1.17
10	2	8.01	0.24
9	2	17.19	1.85
8	4	7.51	1.22
7	4	11.44	1.67
6	9	11.96	2.25
5	14	7.26	1.06
4	37	9.08	1.74
3	105	4.78	1.34
2	306	4.11	1.39
1	901	2.70	1.67
0	380	2.25	0

Table 2. Success rate of questions on StackOverflow

Source	Successful	Ordinary	Unsuccessful
Base Group	39.82%	38.69%	21.49%
StackOverflow	50.87%	34.99%	14.15%

that of average StackOverflow questions and 8.54% lower than the average StackOverflow answers. Out of those, there are unique posts with 17, 15, 14 and 13 answers respectively, these will be further investigated in the **Discussions** section.

4.2 What features characterize energy-related problems?

To further examine the relationship between questions and answers, we can utilize the AcceptedAnswerId and AnswerCount features of posts to compile a "success" metric. Users of StackOverflow can choose a single answer that best answers their question as the accepted answer. Using the presence or absence of an accepted answer alongside the number of answers to a question, we can categorize questions as "successful" if they have an accepted answer, as "ordinary" if they have answers but no accepted answer and as "unsuccessful" if they have no answers, similarly to the success metric defined by Pinto et al.[23]. The success metrics for both the Base group and StackOverflow overall can be observed in Table 2.

The results show that most questions from both groups have answers, with the 79% of the Base group, and 86% of the StackOverflow group, having an answer. Surprisingly, while the Base group questions are slightly more likely to have a not-accepted answer, they also have a significantly smaller chance of having a successful answer than the StackOverflow group. It is also worth noting that the Base group also has a higher likelihood of a question having no answers compared to the the average question on StackOverflow. Using the odds ratio, considering only the Successful and Unsuccessful categories, we find that questions of StackOverflow have 1.94 greater odds of having successful answers compared to the Base group, with a 95% confidence interval of 1.71 to 2.20 and a p-value of < 0.0001, suggesting a statistically significant result.

Table 3. Popularity of questions in Base group and associated variable values

	$\mathbb{S}$	$\mathbb{A}$	$\mathbb{C}$	$\mathbb{V}$	$\mathbb{P}$
Base Group	0.93	0.82	0.94	0.58	3.27
Median	1	1	1	0	-
Std. Deviation	2.71	0.70	1.40	1.26	-

Table 4. Popularity of answers in Base group and associated variable values

	$\mathbb{S}$	$\mathbb{C}$	$\mathbb{PA}$
Base Group	0.56	0.99	1.55
Median	1	0	-
Std. Deviation	1.23	1.88	-

An additional metric that could be used to analyze StackOverflow questions is popularity. A worthwhile example of such a metric for questions was defined previously by Pinto et al.[23]. To define the popularity, we can leverage the Score, AnswerCount, CommentCount, FavoriteCount and ViewCount features of each post, defining the Popularity metric as follows:

$$\mathbb{P} = \mathbb{S} + \mathbb{A} + \mathbb{C} + \mathbb{V}$$

We define $\mathbb{S}$ as the Score of a question normalized by the average Score of questions on StackOverflow, which is currently 2.3. The other values for answer, comment and view count respectively, are similarly normalized using the StackOverflow average for each corresponding variable. We normalize the values using the StackOverflow average for each feature in order to be able to compare the 2 groups and in order to avoid large absolute values. FavoriteCount is also a usable metric that could be used in the gauging popularity but since the favorite score for the StackOverflow group is 0.000081 with a standard deviation of 0.1, we decided to exclude it. The results of this evaluation can be observed in Table 3.

Using the same normalization approach as outlined for the Base group, each metric of the StackOverflow group is normalized to be 1, resulting in a popularity $\mathbb{P}$ score of 4. Using these results we can observe that the questions of the Base group are 18% less popular than the average question of StackOverflow, with the average score, answer count, comment count and view count being lower by 7%, 18%, 6% and 42%, respectively.

Additionally, we can investigate the popularity of Base group answers utilizing the same normalization method. In this case we calculate the popularity $\mathbb{PA}$ using the score and comment count variables, since the other post variables are exclusive to questions. As such, the formula used is:

$$\mathbb{PA} = \mathbb{S} + \mathbb{C}$$

In this case, since we are only using the normalized values for score and count, the StackOverflow answer popularity $\mathbb{PA}$ is 2. based on the answer popularity metrics of Table 4 we can observe that the mean score of answers is 44% lower than that of the average

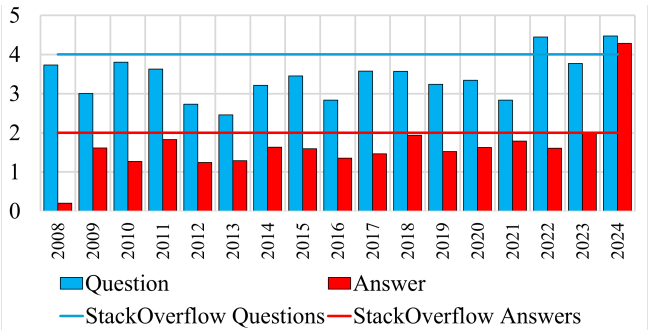

Fig. 5. Popularity of questions and answers in Base group, per year

Fig. 6. WordCloud of Question bodies, from Base group

StackOverflow answer score and the mean comment count is 1% lower than that of the average StackOverflow comment count. This results in a popularity score that is 23% lower than that of the average StackOverflow answer.

Additionally, we computed the Popularity metrics $\mathbb{P}$ and $\mathbb{PA}$ for each separate year using the same normalization approach as described previously. This result can be observed in Figure 5. The normalized question and answer popularity scores of 4 and 2, respectively, were added for a better visual interpretation. The results show that both the questions and the answers of the Base group are, on average, less popular than StackOverflow posts, with 2022 and 2024 being the only year when the questions were 11% and 12% more popular than the average StackOverflow question, otherwise being, on average, 18% less popular.

4.3 What platforms and conditions are typically associated with energy-related problems?

While the post bodies themselves offer significantly more detail regarding the context and problems of developers, and the post metrics such as score, comment, view and answer counts offer worthwhile insights into the popularity of these posts, the tags of posts are also worth investigating. Utilizing the data mining toolbox Orange3 we can compute the word cloud of the Base group question bodies and compare them against the word cloud generated using only tags, as can be seen in Figure 6 and Figure 7.

Observing the word cloud of the question bodies, it can be seen that it does contain terms relevant to platforms, programming languages, etc. but they are mixed with equally valuable keywords

Fig. 7. WordCloud of Question tags, from Base group

Table 5. Platform-related tags, ordered by number of posts

Platform	Tag Count	Post Count
android	823	602
ios	159	152
linux	91	71
windows	78	64
mobile	34	31
cloud	34	34

Table 6. Programming language-related tags, ordered by number of posts

Programing Language	Tag Count	Post Count
java	104	103
python	93	81
sql	73	52
javascript	43	43
objective-c	40	40
swift	36	35
opengl	24	22

related to the problems, solutions and fine-grained context of the questions, resulting in a mix where the repetitions of words within the body offer more insight into the struggles of developers but less relevant data regarding the platforms and conditions of those problems. On the other hand, the restrictive tag limitations of StackOverflow questions (atleast 1 and at most 5 tags per question) make the Tags variable an interesting examination candidate when it comes to question context. While tags were not particularly useful in gathering the Base group results since there are no unified tags relating to software energy consumption, they are very indicative of platform, programming languages, or device context, as can be seen using Figure 7. As such, we compiled a tabular overview of the most top 50 most popular tags, classifying them as platform-related, programming language-related or device-related, as appropriate. The results of this classification can be seen in Tables 5, 6 and 7.

These results show that, out of the 1768 questions with a total of 6382 tags, 12.9% of all tags were android-related, corresponding to 602 questions (34%). Additionally, 954 (54%) questions of the Base group contained at least one of the selected platform tags. Furthermore, 376 (21%) questions contained at least one selected programming language tag and 161 (9%) questions contained one

Table 7. Device-related tags, ordered by number of posts

Device	Tag Count	Post Count
iphone	56	55
arduino	44	42
embedded	41	39
raspberry	25	25

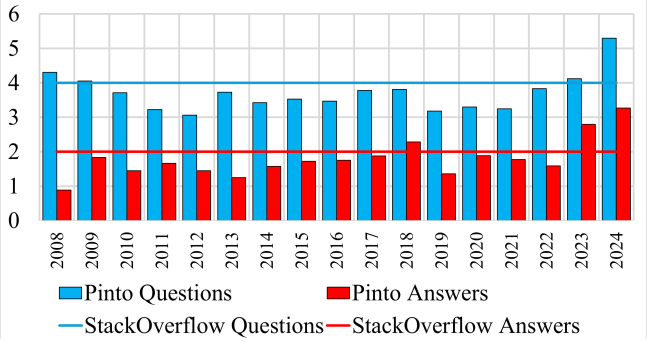

Fig. 8. Popularity of questions and answers of Pinto group, per year

or more selected device-related tags. Querying the tag field of the Base group questions, we find there is a total of 1220 questions (69%) utilizing at least one of the evaluated tags, these questions being associated with 1565 answers (69%).

5 DISCUSSION

The empirical analysis of the Base group of StackOverflow data suggests that the Base group questions are less popular, on average, than other StackOverflow questions by 18%. Additionally, the Base group questions are 22% less likely to be successful and 52% more likely to be unsuccessful when compared to the StackOverflow group. These results are surprising, especially considering the results of similar StackOverflow energy efficiency mining studies such as those of Pinto and colleagues[23], generalized StackOverflow studies such as those of Moutidis et al.[17] or the 2023 StackOverflow survey[30]. A worthwhile note is that we have additionally modelled the question and answer popularity results for the Query used in the "Mining Questions about Software Energy Consumption" paper of Pinto et al.[23] before the manual false-positive extraction phase. The results can be observed in Figure 8.

The results of this figure are particularly interesting when compared to that of Figure 5, considering that it represents the data group obtained before the manual filtering of Pinto's question group. This shows that, before the manual false positive extraction phase, the average popularity score in the time period presented in the paper (2008-2014) was 3.68 for questions and 1.69 for answers which is significantly lower than the scores of 10.45 and 1.82 documented after the false positive extraction phase. The similarity between these results can reasonably suggest the importance of manual validation of the posts relevancy to the topic of software energy efficiency before the analysis of the selected group, when it comes to calculating and comparing the popularity metric.

Table 8. Popularity of outlier questions and respective answers

Answer Count	$\mathbb{S}$	$\mathbb{A}$	$\mathbb{C}$	$\mathbb{V}$	$\mathbb{P}$	$\mathbb{PA}$
17	1	3.11	2.74	0.08	6.92	0.42
15	6.28	4.39	1.09	4.17	15.93	4.86
14	21.76	6.41	2.99	2.60	33.75	3.65
13	3.21	3.80	3.27	0.21	10.49	1.17

Additionally, the tag analysis suggests that context information relevant to the questions such as platform, programming language or device type can reasonably be inferred from the tags of StackOverflow posts. This does not however mean that the tags suggest certain topics. For example, in the case of platforms such as Android, questions related to software energy consumption[22] and questions not necessarily connected to development of energy-efficient software (such as disabling power saving features as a troubleshooting method)[13] co-exist under the same tag. Still, it is worth noting the strong correlation between software energy efficiency questions and mobile and embedded platforms such as Android, iOS, Java, iPhone, Arduino, etc.

5.1 Investigating outliers

In terms of questions considered outliers based on the number of answers they've received, we noted 4 unique questions with more than 10 answers each, possessing 17, 15, 14 and 13 answers, respectively[7, 18, 35, 32]. The question popularity, popularity-related variables and average answer popularity is thus given in Table 8.

based on the results we can observe that these questions are significantly more popular than both the average Base group question and the average Stack Overflow question. Compared to StackOverflow, the questions are 319% more popular on average, with their answers being 26% more popular on average. Delving into the questions themselves, the topics discussed are high-level design considerations for maximising power efficiency of daemon-like software, advantages and disadvantages of binary and ternary computing, specific and unique practices for optimizing power-efficiency of developed applications and power-consumption-related advantages and disadvantages of PHP vs Java. An interesting common feature shared by these questions is that, while the average StackOverflow question is most likely to ask for aid in changing or debugging a specific code snippet(78%), these posts offer more abstract questions aimed at expanding the understanding of developers regarding specific topics or practices. Interestingly, the more abstract questions on ternary computing and power-efficient practices are the only ones with accepted answers and an average answer popularity above that of StackOverflow, while the questions discussing the energy-consumption aspects of daemons and programming languages have comparatively low answer popularity. These findings seem to support the investigations of Manotas et al.[15] and Pinto et al.[23] regarding the high interest and limited knowledge of software practitioners about software energy efficiency.

We have additionally investigated the questions and answers of 2008 and 2024, being the years with the lowest and highest answer popularity, respectively. The question and answer popularity distributions for these years was modelled in Table 9. Additionally, it is

Table 9. Distribution of popularity of questions and answers for 2008 and 2024, from Base group

Year	$\mathbb{P} \geqslant 4$	$4 > \mathbb{P} \geqslant 0$	$\mathbb{PA} > 2$	$2 \geqslant \mathbb{PA} > 1$	$1 > \mathbb{PA} \geqslant 0$
2008	1	2	0	2	22
Year	$\mathbb{P} \geqslant 4$	$4 > \mathbb{P} \geqslant 0$	$\mathbb{PA} > 4$	$4 \geqslant \mathbb{PA} > 2$	$2 > \mathbb{PA} \geqslant 0$
2024	4	14	2	1	9

worth noting that for 2008, we only have 3 questions, having popularity scores of 6.92, 2.39 and 1.89. These questions have a total of 24 answers with the top 3 most popular answers, having $\mathbb{PA}$ scores of 1.35, 1.31 and 0.61, being considered outliers. Considering the data, the low average $\mathbb{PA}$ score for answers of this year is not surprising given the the large number of questions and the low $\mathbb{PA}$ scores of the outlier answers, which are still significantly more popular than other answers of the same questions. For 2024 we observed 19 questions with the outliers being the top 4 largest popularity scores of 33.87, 16.22, 10.61 and 9.43 alongside the lowest popularity score of -6.93. In this case there are only 12 answers, amongst which 1 outlier with a $\mathbb{PA}$ score of 40.91. Given the extreme value of this outlier and the comparatively small number of answers, the average $\mathbb{PA}$ score of this year being 114% larger than the respective StackOverflow score is not surprising.

5.2 LLM Group Analysis

We observed that the LLM group popularity was 3.23, a 1.22% decrease over the Base group. Additionally, for this group 37.69% of questions were found to be successful, 38.94% ordinary and 23.32% unsuccessful, a 5%, 1% and 9% difference, respectively. To verify the popularity difference between the Base and LLM groups we performed a two-tailed Mann–Whitney U test, resulting in a p-value of 0.8735, a z-test score of 0.1592 and an small effect size of 0.0031, suggesting no significant difference between the popularity of the Base and LLM groups.

5.3 Limitations

The most significant limitation of this research was the overwhelming amount of data considering the limited scope and time for this research, which prevented the possibility manual removal of false-positives or a comprehensive thematic analysis. The significant amount of time required to refine the post extraction, analyzing topic modelling performance and conducting the empirical analysis further removed the possibility of performing a manual analysis of the Base group.

An attempt at automated exclusion of false-positives from the Base group using OpenAI's ChatGPT LLM was made, but there were significant limitations to this approach. Firstly, it required a rather significant 291 question manually-labeled dataset, used for self-supervised learning, to improve the accuracy from 62% to 74.5%. Considering the classifying performance of less complex and more power efficient models such as linear regression (59%) or k-means clustering (62%) for the same task, the LLM performance was significantly better. However, the lack of consistency and high power consumption per training run when it comes to using this specific model for classifying makes this approach much less power and energy-efficient overall, especially when considering the quantity of required repetitions to get these results (in our case being 8 instances of separate, repeated training). No repeated instance of the classification problem yielded the same result, often resulting in a significantly lower accuracy of 50-60%.

Another unfortunate result is the poor topic coherence of the Base group under LDA topic modelling, between 35% and 47%. While the total of 4025 posts of the Base group pose a significant challenge when it comes to manual analysis, the data size is still considered too limited, considering the model. Additionally, the presence of false positives and the large variance in body, tags and title lengths significantly limits the performance of this approach.

5.4 Future Work

While we were able to get a group of questions which were 66% relevant to the desired energy efficiency topic, manually eliminating all irrelevant results from that data was impossible considering the time restrictions and the fact that this research was conducted by a single person. Still, the observations described in the **Discussions** section suggest that the remaining false-positives significantly affect the popularity calculations and likely the success calculations. As such, given the significant amount of data currently available, extending and replicating the experiments and analysis of the study by examining a subset of StackOverflow questions verified to contain only posts directly relevant to the topic of software energy efficiency could prove to be very valuable to both developers and researchers.

6 CONCLUSION

The goal of this paper was to evaluate the topic of software energy consumption on StackOverflow. Tag analysis was used to gather valuable insight into the most popular tags associated with software energy consumption by extracting the context of platforms, programming languages and devices associated with said posts. The most relevant context was found to be mobile platforms, followed by desktop and embedded platforms. Given our data selection methods, we have found these questions to generally be less popular and less successful compared to other StackOverflow questions. The striking difference between the provided results and relevant literature serve to reinforce the importance of manual data validation and outline the limitations of automated data filtering and categorization. While the results did not match the expected outcomes, we are confident in the methods and analysis conducted, and believe additional manual data validation to be necessary for better future evaluations.

ACKNOWLEDGMENTS

I would like to offer my gratitude to Fernando Castor for their supervision, feedback and suggestions regarding this research.

A APPENDIX

A.1 Link to data zip:

https://drive.google.com/drive/folders/1wtBpbBdYP8TYWEPHgR00IKcU2xOOZw3i?usp=sharing

A.2 AI Usage

In preparation of this work the author used OpenAI's ChatGPT model to conduct automated classification of the Base group dataset, with the resulting dataset being investigated as the LLM group. After using this tool/service, the author reviewed and edited the content as needed and takes full responsibility for the content of the work.

REFERENCES

[1] Sven Amann, Stefanie Beyer, Katja Kevic, and Harald Gall. 2015. Software Mining Studies: Goals, Approaches, Artifacts, and Replicability. en. In *Software Engineering: International Summer Schools, LASER 2013-2014, Elba, Italy, Revised Tutorial Lectures*. Bertrand Meyer and Martin Nordio, (Eds.) Springer International Publishing, Cham, 121–158. ISBN: 978-3-319-28406-4. DOI: 10.1007/978-3-319-28406-4_5.

[2] Inc. Apple. [n. d.] Introducing Apple Intelligence for iPhone, iPad, and Mac. en-US. (). Retrieved June 26, 2024 from https://www.apple.com/newsroom/2024/06/introducing-apple-intelligence-for-iphone-ipad-and-mac/.

[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. en. *Empirical Software Engineering*, 19, 3, (June 2014), 619–654. DOI: 10.1007/s10664-012-9231-y.

[4] Charles. 2023. Battery Storage modelling for a data centre. Forum post. (July 2023). Retrieved June 28, 2024 from https://stackoverflow.com/q/76717041/24823946.

[5] Yunbin Deng. 2019. Deep learning on mobile devices: a review. In *Mobile Multimedia/Image Processing, Security, and Applications 2019*. Vol. 10993. SPIE, (May 2019), 52–66. DOI: 10.1117/12.2518469.

[6] Pierpaolo Dondio and Suha Shaheen. 2020. Is StackOverflow an Effective Complement to Gaining Practical Knowledge Compared to Traditional Computer Science Learning? In *Proceedings of the 11th International Conference on Education Technology and Computers* (ICETC '19). Association for Computing Machinery, New York, NY, USA, (Jan. 2020), 132–138. ISBN: 978-1-4503-7254-1. DOI: 10.1145/3369255.3369258.

[7] goldenmean. 2010. Power Efficient Software Coding. Forum post. (Aug. 2010). Retrieved June 29, 2024 from https://stackoverflow.com/q/61882/24823946.

[8] Anisul Islam. 2017. How to Convert 6s Power Consumption Time Series Data To 1 Hour Data? Forum post. (Dec. 2017). Retrieved June 27, 2024 from https://stackoverflow.com/q/48040529/24823946.

[9] Julian. 2019. Notification channel unrecoverably broken (only some devices). Forum post. (Jan. 2019). Retrieved June 27, 2024 from https://stackoverflow.com/q/54167900/24823946.

[10] jxgn. 2012. Battery indicator for android. Forum post. (July 2012). Retrieved June 28, 2024 from https://stackoverflow.com/q/9579329/24823946.

[11] Samia Kabir, David N. Udo-Imeh, Bonan Kou, and Tianyi Zhang. 2024. Is Stack Overflow Obsolete? An Empirical Study of the Characteristics of ChatGPT Answers to Stack Overflow Questions. arXiv:2308.02312 [cs]. (Feb. 2024). DOI: 10.48550/arXiv.2308.02312.

[12] Avita Katal, Susheela Dahiya, and Tanupriya Choudhury. 2023. Energy efficiency in cloud computing data centers: a survey on software technologies. en. *Cluster Computing*, 26, 3, (June 2023), 1845–1875. DOI: 10.1007/s10586-022-03713-0.

[13] Ramanpreet Singh Khokhar. 2019. How to make foreground Services work in MIUI? Forum post. (Jan. 2019). Retrieved June 23, 2024 from https://stackoverflow.com/q/54009355/24823946.

[14] lucasdo. 2015. How Android API get battery percentage? Forum post. (Dec. 2015). Retrieved June 28, 2024 from https://stackoverflow.com/q/34253262/24823946.

[15] Irene Manotas, Christian Bird, Rui Zhang, David Shepherd, Ciera Jaspan, Caitlin Sadowski, Lori Pollock, and James Clause. 2016. An empirical study of practitioners' perspectives on green software engineering. In *Proceedings of the 38th International Conference on Software Engineering* (ICSE '16). Association for Computing Machinery, New York, NY, USA, (May 2016), 237–248. ISBN: 978-1-4503-3900-1. DOI: 10.1145/2884781.2884810.

[16] Yusuf Mehdi. 2024. Introducing Copilot+ PCs. en-US. (May 2024). Retrieved June 26, 2024 from https://blogs.microsoft.com/blog/2024/05/20/introducing-copilot-pcs/.

[17] Iraklis Moutidis and Hywel T. P. Williams. 2021. Community evolution on Stack Overflow. en. *PLOS ONE*, 16, 6, (June 2021), e0253010. Publisher: Public Library of Science. DOI: 10.1371/journal.pone.0253010.

[18] ojblass. 2020. Why binary and not ternary computing? Forum post. (Dec. 2020). Retrieved June 29, 2024 from https://stackoverflow.com/q/764439/24823946.

[19] Wellington Oliveira, Renato Oliveira, Fernando Castor, Gustavo Pinto, and João Paulo Fernandes. 2021. Improving energy-efficiency by recommending Java collections. en. *Empirical Software Engineering*, 26, 3, (Apr. 2021), 55. DOI: 10.1007/s10664-021-09950-y.

[20] Candy Pang, Abram Hindle, Bram Adams, and Ahmed E. Hassan. 2016. What Do Programmers Know about Software Energy Consumption? *IEEE Software*, 33, 3, (May 2016), 83–89. Conference Name: IEEE Software. DOI: 10.1109/MS.2015.83.

[21] David Patterson, Joseph Gonzalez, Quoc Le, Chen Liang, Lluis-Miquel Munguia, Daniel Rothchild, David So, Maud Texier, and Jeff Dean. 2021. Carbon Emissions and Large Neural Network Training. arXiv:2104.10350 [cs]. (Apr. 2021). DOI: 10.48550/arXiv.2104.10350.

[22] Philiz. 2018. Energy consumption on Android Studio Profiler. Forum post. (Oct. 2018). Retrieved June 23, 2024 from https://stackoverflow.com/q/52647045/24823946.

[23] Gustavo Pinto, Fernando Castor, and Yu David Liu. 2014. Mining questions about software energy consumption. In *Proceedings of the 11th Working Conference on Mining Software Repositories* (MSR 2014). Association for Computing Machinery, New York, NY, USA, (May 2014), 22–31. ISBN: 978-1-4503-2863-0. DOI: 10.1145/2597073.2597110.

[24] Gustavo Pinto, Fernando Castor, and Yu David Liu. 2014. Understanding energy behaviors of thread management constructs. In *Proceedings of the 2014 ACM International Conference on Object Oriented Programming Systems Languages & Applications* (OOPSLA '14). Association for Computing Machinery, New York, NY, USA, (Oct. 2014), 345–360. ISBN: 978-1-4503-2585-1. DOI: 10.1145/2660193.2660235.

[25] Christoffer Rosen and Emad Shihab. 2016. What are mobile developers asking about? A large scale study using stack overflow. en. *Empirical Software Engineering*, 21, 3, (June 2016), 1192–1223. DOI: 10.1007/s10664-015-9379-3.

[26] Konstantinos I. Roumeliotis and Nikolaos D. Tselikas. 2023. ChatGPT and Open-AI Models: A Preliminary Review. en. *Future Internet*, 15, 6, (June 2023), 192. Number: 6 Publisher: Multidisciplinary Digital Publishing Institute. DOI: 10.3390/fi15060192.

[27] İbrahim Şanlıalp, Muhammed Maruf Öztürk, and Tuncay Yiğit. 2022. Energy Efficiency Analysis of Code Refactoring Techniques for Green and Sustainable Software in Portable Devices. en. *Electronics*, 11, 3, (Jan. 2022), 442. Number: 3 Publisher: Multidisciplinary Digital Publishing Institute. DOI: 10.3390/electronics11030442.

[28] Camila Costa Silva, Matthias Galster, and Fabian Gilson. 2021. Topic modeling in software engineering research. en. *Empirical Software Engineering*, 26, 6, (Sept. 2021), 120. DOI: 10.1007/s10664-021-10026-0.

[29] slayerpjo. 2016. Getting around Meizu App Auto-Clean. Forum post. (Dec. 2016). Retrieved June 27, 2024 from https://stackoverflow.com/q/41112945/24823946.

[30] Inc. Stack Exchange. [n. d.] Stack Exchange Data Dump : Stack Exchange, Inc. : Free Download, Borrow, and Streaming. en. (). Retrieved May 1, 2024 from https://archive.org/details/stackexchange.

[31] Stella. 2013. How to export the Energy Usage result datas. Forum post. (Oct. 2013). Retrieved June 27, 2024 from https://stackoverflow.com/q/19557086/24823946.

[32] Thomaschaaf. 2013. PHP vs. Java are there energy consumption differences? Forum post. (Sept. 2013). Retrieved June 29, 2024 from https://stackoverflow.com/q/1318851/24823946.

[33] Neil C. Thompson, Kristjan Greenewald, Keeheon Lee, and Gabriel F. Manso. 2022. The Computational Limits of Deep Learning. arXiv:2007.05558 [cs, stat]. (July 2022). DOI: 10.48550/arXiv.2007.05558.

[34] Max Weber, Christian Kaltenecker, Florian Sattler, Sven Apel, and Norbert Siegmund. 2023. Twins or False Friends? A Study on Energy Consumption and Performance of Configurable Software. In *2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE)*. ISSN: 1558-1225. (May 2023), 2098–2110. DOI: 10.1109/ICSE48619.2023.00177.

[35] Wroclai. 2012. Optimising Android application before release. Forum post. (May 2012). Retrieved June 29, 2024 from https://stackoverflow.com/q/5626947/24823946.

[36] Haoxiang Zhang, Shaowei Wang, Tse-Hsun Chen, Ying Zou, and Ahmed E. Hassan. 2021. An Empirical Study of Obsolete Answers on Stack Overflow. *IEEE Transactions on Software Engineering*, 47, 4, (Apr. 2021), 850–862. Conference Name: IEEE Transactions on Software Engineering. DOI: 10.1109/TSE.2019.2906315.