Joy Bushnell
Haifa Aldossari
Samiha Sharmin Shimmi

I. Introduction

In this project we implemented the paper *One-Day Flies on StackOverflow Why the vast majority of StackOverflow users only posts once*. The paper presents five research questions to study the possible reasons why StackOverflow users post only once. The following five research questions are from the paper:

1. Are one-day flies more likely to create questions eventually marked as duplicates?
2. Are one-day flies more likely to post questions with uncommon tags?
3. Are one-day flies more likely to have their posts removed (either by themselves or by a moderator)?
4. Are one-day flies more likely to attract less views?
5. Are one-day flies more likely to receive no answer to their questions?

We added one more research question with two parts: Do one-day flies remain one-day flies over time? Also we look at if the user posts again what is the average number of days before they post again.

In the following sections we discuss the methodology for each question, the results, the conclusion and the contributions of each member.

II. Methodology

In this section we discuss the methodology of each research question. We start with the queries we used to create our onedayflies table and notonedayflies table. After these queries we go into more detail into the methods used for each research question.

one-day flies query to create one-day flies table
//this query gives the owner_user_id of the one-day flies

```
Create table onedayflies2 as select owner_user_id,count(id) from posts where post_type_id=1 and owner_user_id in (select id from users where creation_date<='2012-01-31' and reputation between 0 and 1)group by id having count(id)=1 ;
```

//this query gives the posts table of one-day flies.

```
create table onedayfliespost as select * from posts where post_type_id=1 and owner_user_id in (select owner_user_id from onedayflies2);
```

This query gives the posts table of not one-day flies.
create table notonedayflies as Select * from posts where post_type_id=1 and owner_user_id not in (select owner_user_id from onedayflies2);

RQ 1 Are one-day flies more likely to create questions eventually marked as duplicates?

New users sometimes ask duplicate questions which are flagged by the community as duplicate. This can discourage them to post more questions. In this question, we tried to figure out if there is a relationship between one-day flies and duplicate questions.

The table used to get duplicate questions is postlinks. But this table was not available in PostgreSQL dump. So, for this research question, we wrote several SQL queries in Stack Exchange to get the data. We worked on the current one-day flies and generated the answers for this research question. If a question is marked as duplicate, the linktypeid field in postlinks table is 3. We wrote our queries based on this field.

The following queries were used:

The following query gets the questions which are duplicate by one-day flies on current dataset

```
select count(*) from postlinks
where postid in
(
select id from posts where owneruserid in (select owneruserid from(
select ownerUserId,count(id)c from posts
where posttypeid=1 and owneruserid in
(select id from users where creationdate<='2017-10-23'
and reputation between 0 and 1)group by owneruserid having count(id)=1
)a)
) and linktypeid=3
```

The following query gets the questions which are not duplicate by one-day flies on current dataset

```
select count(*) from postlinks
where postid in
(
select id from posts where owneruserid in (select owneruserid from(
select ownerUserId,count(id)c from posts
where posttypeid=1 and owneruserid in
(select id from users where creationdate<='2017-10-23'
```

```
and reputation between 0 and 1)group by owneruserid having count(id)=1
)a)
) and linktypeid<>3
```

The following query gets duplicates questions by the users who are not one-day flies on current dataset

```
select count(*) from postlinks
where postid in
(
select id from posts where owneruserid not in (select owneruserid from(
select ownerUserId,count(id)c from posts
where posttypeid=1 and owneruserid in
(select id from users where creationdate<='2017-10-23'
and reputation between 0 and 1)group by owneruserid having count(id)=1
)a)
) and linktypeid=3
```

The following query gets the questions which are not duplicates by the users who are not one-day flies in the current dataset

```
select count(*) from postlinks
where postid in
(
select id from posts where owneruserid not in (select owneruserid from(
select ownerUserId,count(id)c from posts
where posttypeid=1 and owneruserid in
(select id from users where creationdate<='2017-10-23'
and reputation between 0 and 1)group by owneruserid having count(id)=1
)a)
) and linktypeid<>3
```

RQ 2 Are one-day flies more likely to post questions with uncommon tags?

For this research question the authors expected that many one-day flies use uncommon tags which could cause the one-day flies to have less views and consequently less answers. Hence they would not continue to post.

For research question 2 we started by exporting the data from the cloud PostgreSQL database using the SQL queries below and then we used python to extract the tags and count the frequency of tags of both one-day flies and other users. After counting the top tags, we limited

the tags written to the csv file to the top 50. After the top 50 tags are selected we graphed the top 10 using Libreoffice. See tagfreq.py. Below are the SQL statements used in this research question.

```
\o one-dayTags.csv
select tags from one-day fliespost where post_type_id=1;

\o otherTags.csv
Select tags from notone-day flies;
```

RQ 3 Are one-day flies more likely to have their posts removed?

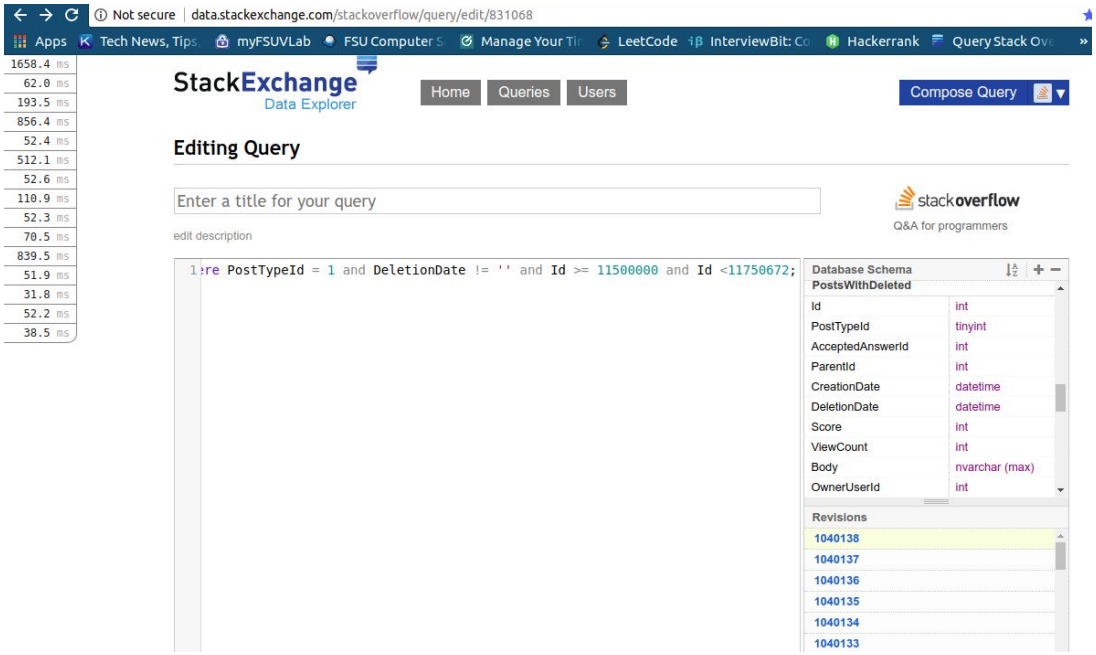
For research question 3, we looked into and compared the number of deleted posts to one-day flies and other posts. The theory provided by the authors is that one-day flies were more likely to have their posts deleted either because they created a duplicate post or for another reason.

On stackexchange.com we found a table called PostsWithDeleted that contains deleted posts. We retrieved all deleted posts up to 11 million ids which will contain all the posts from one-day flies and not one-day flies using stackexchange.com. Then the files that were downloaded are combined into one csv file and the one-day flies' ids and other ids are matched to the ids found in the combined csv and counted. Below we show the queries used and the results of the total from deletedPosts.py.

```
select count(closed_date) from one-day fliespost where closed_date IS NOT NULL;
select count(closed_date) from notone-day flies where closed_date IS NOT NULL;
```

Stack Exchange Query

```
Select id, DeletionDate from postswithdeleted where PostTypeId = 1 and DeletionDate != '' and Id >= 1000000 and Id <2000000;
```



RQ 4 Are one-day flies more likely to attract less views?

The new users do not have a reputation. It might happen that, the other users do not view the questions of users who do not have a good reputation. This leads to less views and thus discourages the users who post their first question. We tried to see if they are discouraged because of low view count.

We have the set of one-day flies and the users who are not one-day flies. We wrote SQL queries in the database to generate the results of this question. We discarded the highest viewed and least viewed questions (5%,10%, and 20%) from the dataset. Then we compared the results.

The following query gets view count for all one-day flies
select avg(view_count) from onedayfliespost where post_type_id=1;

The following query gets minimum and maximum view count for all one-day flies
select min(view_count),max(view_count) from onedayfliespost where post_type_id=1;
//min 2 max 35103

The following query gets view count for all one-day flies after discarding 5%

select avg(view_count) from onedayfliespost where post_type_id=1 and view_count>2+.05*2 and view_count<35103-.05*35103;

The following query gets view count for all one-day flies after discarding 10%

```
select avg(view_count) from onedayfliespost where post_type_id=1 and  view_count>2+.1*2 and view_count<35103-.1*35103;
```

The following query gets view count for all one-day flies after discarding 20%

```
select avg(view_count) from onedayfliespost where post_type_id=1 and view_count>2+.2*2 and view_count<35103-.2*35103;
```

The following query gets view count for the users who are not one-day flies

```
select avg(view_count) from posts where owner_user_id not in (
select id from users where id  in (
select owner_user_id from onedayfliespost)) and post_type_id=1  ;
```

The following query gets minimum and maximum view count for the users who are not one-day flies

```
select min(view_count),max(view_count) from posts where owner_user_id not in (
select id from users where id  in (
select owner_user_id from onedayfliespost)) and post_type_id=1  ;
//min 1, max 1051784
```

The following query gets view count for the users who are not one-day flies after discarding 5%

```
select avg(view_count) from posts where owner_user_id not in (select id from users where id  in (select owner_user_id from onedayflies2)) and post_type_id=1  and view_count>1+.05*1 and view_count<1051784-.05*1051784;
```

The following query gets view count for the users who are not one-day flies after discarding 10%

```
select avg(view_count) from posts where owner_user_id not in (select id from users where id  in (select owner_user_id from onedayflies2)) and post_type_id=1  and view_count>1+.1*1 and view_count<1051784-.1*1051784;
```

The following query gets view count for the users who are not one-day flies after discarding 20%

```
select avg(view_count) from posts where owner_user_id not in (select id from users where id  in (select owner_user_id from onedayflies2)) and post_type_id=1  and view_count>1+.2*1 and view_count<1051784-.2*1051784;
```

RQ 5 Are one-day flies more likely to receive no answers to their questions ?

To answer this question, we compared the the set of questions done by one-day flies with the ones done by the other users. We used R to answer this question. First we got the answer count for both one-day flies and other users and then we counted how many null values since this indicates that there are no answers for the questions. Below is the query used to retrieve the answer counts for one-day flies.

```
\o answercount.csv
select answer_count from onedayfliespost where post_type_id=1;
```

RQ 6 Do one-day flies remain one-day flies over time?

To answer this question, we take the current users who have posted questions after the maximum date of our data set and see if they are in our one-day flies data set.

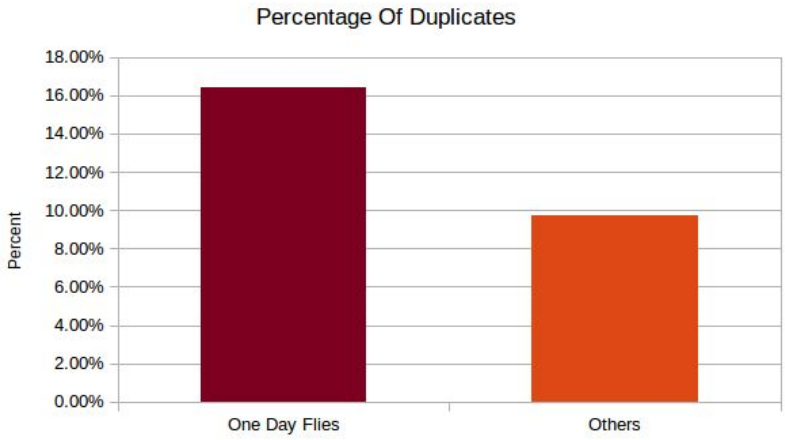
We take the one-day flies from our data set and take the current users who have posted questions after the max date from our data set and match them. This data has been retrieved from stack exchange. (http://data.stackexchange.com/stackoverflow/query/edit/831068) We can compare both sets and determine if users remain one-day flies over time and find the average number of days for a user to post again based on this data. We used R to answer this question. Below are the query used on Stack Exchange.

Stack Exchange Query

```
select distinct OwnerUserID, CreationDate from Posts where PostTypeId=1 and CreationDate > '2012-07-31';
```

III. Results

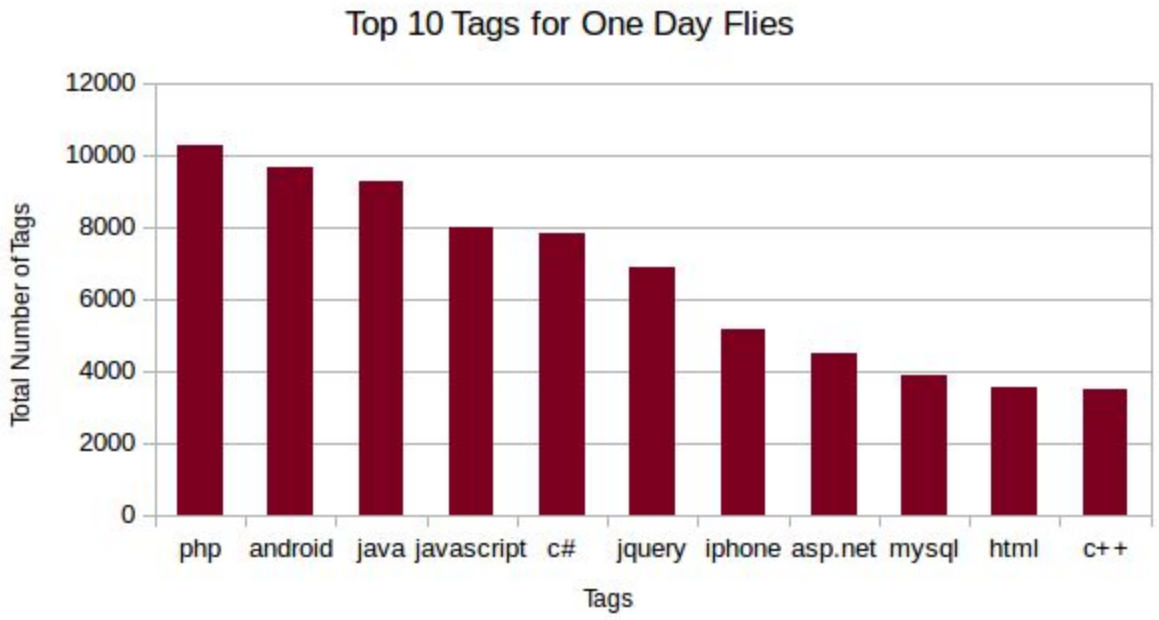
RQ 1 Results

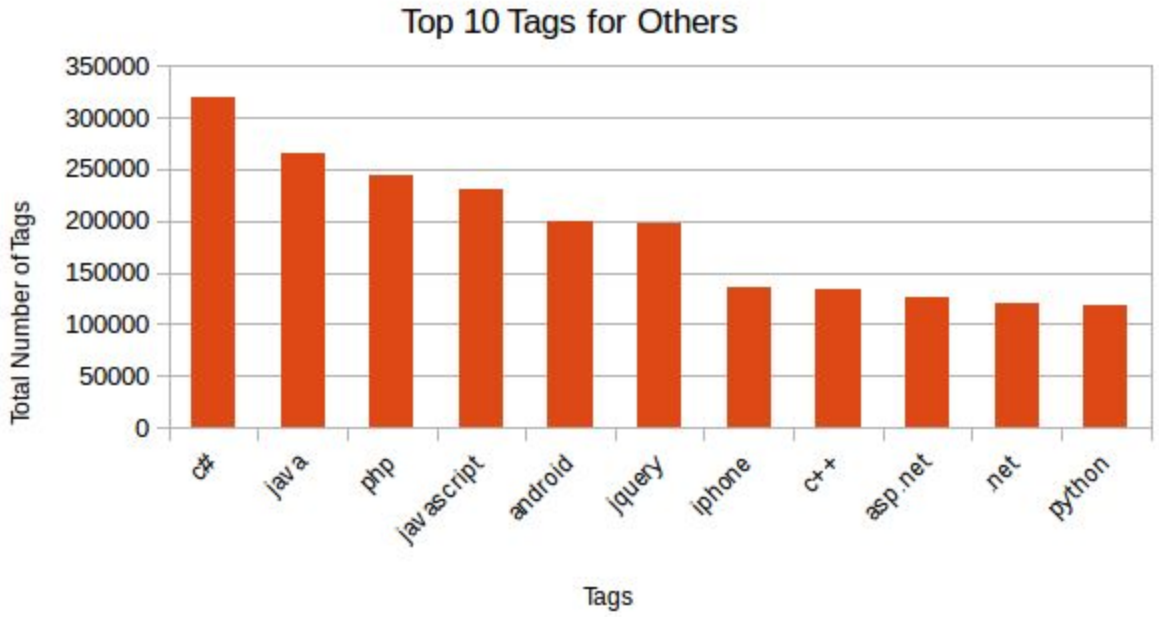


In our finding we found around 16% and 10% duplicates respectively for one-day flies and not one-day flies. In the original paper the authors mentioned that their results were surprising. They found 2.2% and 2.9% of questions were duplicates by one-day flies and not one-day flies respectively. We used the live dataset and took current one-day flies. We think because of this reason, the result is not close for this research question.

RQ2 Results

In the following two graphs we compare the results of the top 10 tags of one-day flies to all other users. For one-day flies the top 10 results are fairly similar to all other users. We see that the top tag is ‘php’ for one-day flies, this is the third most frequent in all other users posts. Likewise the top tag for all other users is ‘c#’ which is the fifth most frequent tag for one-day flies. This clearly debunks the theory that one-day flies use uncommon tags. Below the graphs are the results from the top 50 tags for both one-day flies and other users placed in a chart.





one-day flies Tags	Total Number of Tags		Other Tags	Total Number of Tags
php	10257		c#	319854

android	9672		java	264373
java	9235		php	243494
javascript	7984		javascript	230951
c#	7829		android	200209
jquery	6871		jquery	197902
iphone	5175		iphone	135863
asp.net	4502		c++	133623
mySQL	3894		asp.net	124940
html	3565		.net	119929
c++	3467		python	118112
python	2938		mySQL	104165
objective-c	2771		html	103187
facebook	2675		objective-c	91380
.net	2507		SQL	85290
css	2391		css	80795
SQL	2380		ruby-on-rails	79992
ios	2116		ios	78067
c	1902		c	62754
xml	1762		ruby	49500
ajax	1749		wpf	47212
vb.net	1747		SQL-server	46318
ruby-on-rails	1715		xml	43175
xcode	1427		ajax	40240

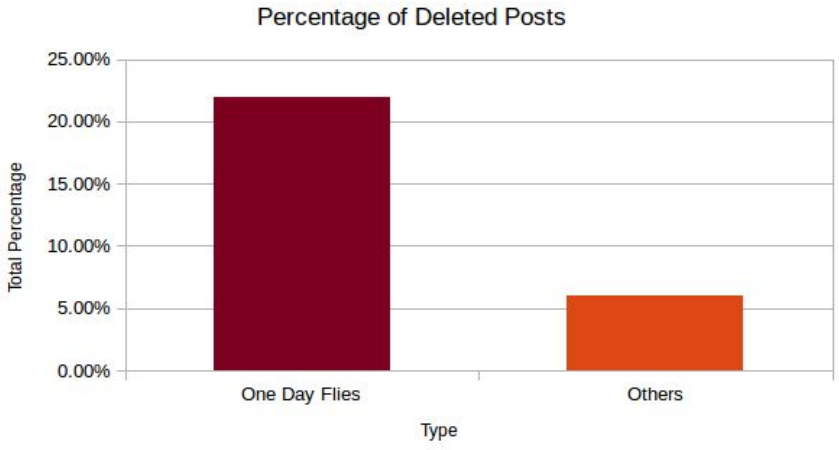
flash	1401		regex	38873
database	1344		asp.net-mvc	38114
wordpress	1325		database	36883
wpf	1293		django	35016
windows	1263		windows	33933
homework	1230		xcode	33783
SQL-server	1172		linux	33423
actionscript-3	1102		arrays	31366
flex	1089		vb.net	30617
eclipse	1061		ruby-on-rails-3	29334
image	1049		eclipse	28350
linux	1036		facebook	27542
ruby	1004		json	26745
arrays	966		winforms	26025
regex	918		multithreading	25675
forms	917		string	25269
web-services	893		asp.net-mvc-3	24703
json	879		visual-studio-2010	23279
apache	873		performance	22412
ipad	867		wcf	22334

excel	859		osx	21204
wcf	831		visual-studio	21109
.htaccess	786		linq	20933
api	774		silverlight	20791
django	771		image	20673
application	760		algorithm	20638

In the paper the authors found that the top 5 tags of both one-day flies and all others were permutations of each other. In our results we can also see that the results are identical and differ only in their order. From this we conclude as the authors did that one-day flies do not use uncommon tags.

RQ3 Results

For one-day flies we find that 28,580 posts out of 128,260 posts were deleted. The deleted posts comprise approximately 22% of all one-day flies posts. For other users in the set the total number of deleted posts was 67385 out of 1048574. This constitutes 6% of the all the other posts. The number of closed posts for one-day flies makes up 6% of all one-day flies posts and total number of closed posts for all other users makes up approximately 8% of all posts.



Our results showed that one-day flies had a deleted question ratio of 22% and all other users had 6% of questions are deleted. For the paper they found that the one-day flies were deleted at a rate of 15.4% and for high post count users 10.9% of questions were deleted. This discrepancy between our results and the paper's could be due to the use of live StackExchange

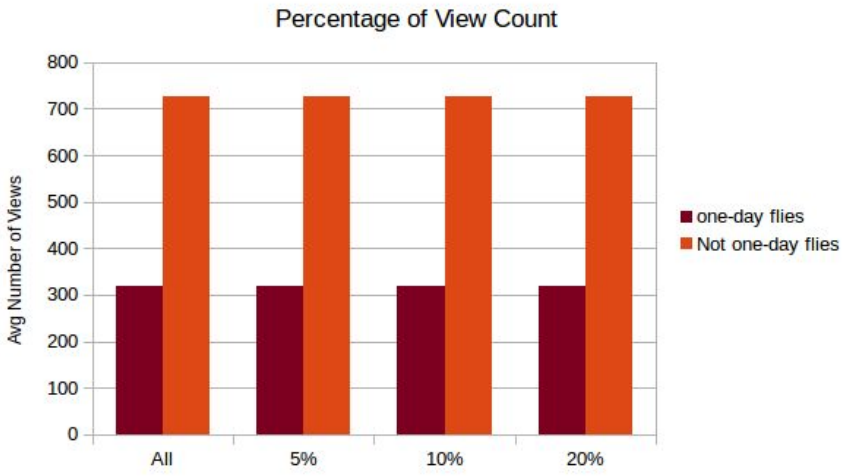
data and the fact that the paper used random sampling of 150,000 one-day flies and not one-day flies in this research question. They also used the DOM structure instead of the StackExchange table PostsWithDeleted.

RQ4 Results

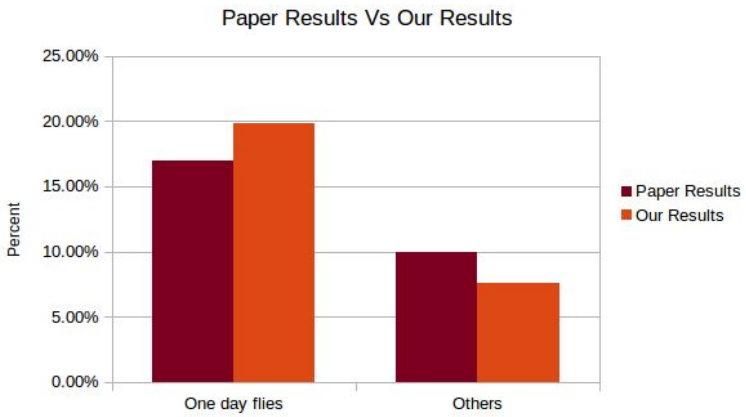
one-day flies
Average
317.63

Not one-day flies
Average
726.16

Average of both
521.89



In the original paper they got average view count 190 where we got 317. They got similar result after discarding extreme values (5%,10%,20%). We also got the same result after discarding the extremes. We used separate datasets and our average view count changed.



RQ5

The paper found that the number of unanswered questions for one-day flies was approximately 17% and the percentage of unanswered in the other group is approximately about 10%. In our results we found that approximately 19% of one-day flies' questions are unanswered and approximately 7% of all other users have no answers. The difference here is negligible between our results and the authors' results, hence we conclude that the difference between one-day flies and all other users is not large enough to account for why one-day flies only post once.

RQ6

The total number of one-day flies that are not one-day flies anymore is 23,557 out of the 128,260 users which is approximately 18%. This means that less than a quarter of the one-day flies do not remain one-day flies and 82% of the one-day flies did not post again. We also found that the average number of days before one-day flies post again was 1154 days. Hence the majority of one-day flies remain one-day flies.

IV. Conclusion

We tried to find the overall behavior of one-day flies from our six research questions. We found that the percent of duplicate questions is higher for one-day flies than for all other users, that one-day flies do not post questions with uncommon tags, and that the number of deleted questions is higher for one-day flies. We also found that the overall view count is lower for one-day flies than for other users. We see that the number of unanswered questions is higher for one-day flies than for all other users and from our final research question, we found that the majority of one-day flies remain one-day flies. We found that most of our results match the paper and the results that do not, are negligible differences. Hence the results show that there is no significant reason for one-day flies behavior.

V. Team member contributions

We spent over a week trying to implement the database (ORACLE, MySQL and PostgreSQL). Then we chose a PostgreSQL database on Google cloud and we struggled to access the instance from our local machines. Finally, we got access through Joy's computer via remote access and from Samiha's computer via Pgadmin. Samiha ran queries to get the one-day flies data into two tables, one for user ids and the other for one-day flies' posts. She also created a query for the not one-day flies table.

We broke up the research questions and each group member worked on two research questions. Samiha had research questions 1 and 4, Joy worked on research questions 2 and 3 and Haifa worked on questions 5 and 6. Joy used her computer to download data from the Google cloud to csv files for questions 2, 3, 5 and 6. She also worked on StackExchange to download data needed for questions 3 and 6. Joy wrote the README file, everyone contributed to the detailed report and to the presentation especially for their research questions.