Journal of Information Processing -Vol. 0 -1-5 (??? 1992)
[DOI: 10.2197/ipsjjip.0.1]

Technical Note

Magnet or Sticky? A Stack Overflow Tag-by-Tag Typology

KOTORI HIEDA[1a)] YU MINGZHE[1b)] YASUTAKA KAMEI[2c)]

Received: xx xx, xxxx, Accepted: xx xx, xxxx

Abstract: Stack Overflow (SO) is one of the most popular question and answer ~~websites among~~sites for software developers. SO stores posts assigned with tags that ~~indicate~~correspond to the ~~keyword that categories the~~keywords of each question. ~~For example, if~~If a developer asks ~~about~~a question related to Python and ~~puts "~~inputs "Python~~"~~" tag on the post, the developers ~~who are~~ interested in Python can ~~easily find and answer~~ participate in the post~~.~~ easily. Since 2008, SO has ~~started its service since 2008 and is still becoming popular. Therefore~~become one of the most trusted online communities. In the present study, we explore ~~how developers'~~developers' interest ~~shift~~ by analyzing how they use tags. We classify tags into four ~~types~~: (1) attractive, (2) stagnant, (3) fluctuating, and (4) terminal based on magnet values and sticky values. We analyze ~~the~~ data ~~from~~of table "~~"~~Posts~~" which includes about~~" of approximately 42 million posts ~~from Stack overflow~~in SO and table "~~"~~Users~~" where there are about~~" of approximately 9 million rows of user information. Results reveal that: (1) ~~There were some~~Some historical events in IT ~~such as the launch~~are retrieved, which include launching of new tools and ~~the termination~~terminating of services ~~when there were characteristics in~~with the transition of magnet value and sticky value. (2) The ~~types~~characteristics of ~~tags that are~~these classified ~~do not change much~~tags are the same.

Keywords: magnet, sticky, tag, user migration, OSS census

1. Introduction

The Pew Research Center (PRC) [1~~], an organization~~] is the U.S. fact finder that ~~studies~~provides information on social problems ~~in~~and demographic trends that shape the United States and the world~~, investigated society and population using US tax survey data. States that have a high percentage of people migrating from the outside~~. *Magnet states* are defined as ~~*magnet states* and~~ states where a high percentage of the ~~high~~ population migrated from the outside, whereas *sticky states* have a high proportion of ~~the population who continues to live~~those living in the same state since birth~~ are defined as *sticky states*. For example,~~ Nevada is *a magnet* state because 86% of ~~Nevada's~~the population migrated from other states~~ so Nevada state is quite *magnet*. Through such a survey, we can~~. It is possible to find the ~~tendency how~~movement of American citizens ~~move.~~

~~For~~ by studying this demographic trend. ~~For~~ software developers, ~~it is~~ understanding other developers' interests are important ~~to know the changes in~~ as the ~~interests~~popularity of ~~other~~ developers ~~because popularity among many developers should~~ have advantages. ~~Developers always want~~Many developers like to work with convenient and easy-to-use tools. To ~~make~~develop a project ~~better, excellent~~efficiently, developers ~~need to be interested in the project over the~~ must focus on long~~ ~~term.

~~Therefore, in~~ projects. In this study, we focus on new and existing ~~interests in the~~ topics of ~~StackOverflow.~~Stack Overflow (SO). Inspired by previous studies [2], we apply Magnet and Sticky metrics to the topics ~~that are~~ collected in ~~StackOverflow.~~SO. The ~~m~~Magnet metric ~~indicates~~is the number of new developers attracted to a topic and Sticky metric ~~indicates~~is the number of existing developers ~~that~~who stay with the topic. We examined ~~tags' magnets and sticky~~the values of tags "magnets" and tags "sticky" by classifying them ~~as~~to the tags *programming language, framework,* and *environment.* We also compared the news and history of software companies and web services. If changes in their characteristics are discovered, we examine factors responsible for the changes based on their magnet values and sticky values.

We address the following two research questions:

(RQ1) What are the values of magnet and sticky in Stack Overflow?
In many cases, the sticky value is higher than the magnet value. In addition, the magnet value rate decreases more than that of the sticky value. The magnet value and the sticky value can be easily used like the .NET Framework

(RQ2) How do magnet and sticky values change over time?
We can identify obsolete tags. When the tags move quadrant, we find that something happens.

2. –Definition of Magnet and Sticky

This section describes how we measure the appeal and adhesion of users on different topics. Following the Pew Research Center (PRC) definition, we use the Magnet and Sticky metrics to illustrate the migratory trends of the U.S. citizens. The PRC defines magnet states as states where a large proportion of adults are from other states. From the magnet metric, the proportion of adults residing in the magnet states were not born in the state. PRC defines the sticky state as the state where a large proportion of adults born there continue to live in the state. Thus, the sticky metric for the state is the proportion of adult residents born in the state. These definitions are good for a population study where a single adult can only occupy one state at a time. However, the definition is inapplicable to the topics discussed by the SO users as users can ask or answer questions on several topics at the same time. Therefore, we expand new definitions for SO topics (refer to Figure 1):

Graduate School of Information Science and Electrical Engineering, Kyushu University [2] –A«*?^XfA LSI W^"^" fe>#-
Faculty of Information Science and Electrical Engineering, Kyushu University
[a)] ~~hieda@f.ait.kyushu-u.ac.jp~~hieda@f.ait.kyushu-u.ac.jp
[b)] ~~yumingzhe@f.ait.kyushu-u.ac.jp~~
[c)] ~~kamei@ait.kyushu-u.ac.jp~~

Comment [A1]: Remark: Note that the corresponding text has been heavily edited. Please verify whether the edits convey the desired meaning.

Comment [A2]: Remark: Note that the corresponding text has been heavily edited. Please verify whether the edits convey the desired meaning.

companies and web services, if there are characteristic changes in magnet values and sticky values, we examined why it was like that. We addressed the following two research questions: **(RQ1) What are typical values of magnet and sticky in Stack Overflow?**
In many cases, the sticky value tended to be higher than the magnet value. In addition, the decrease rate was higher for the magnet value than for the sticky value. If it is easy to use and convenient like the .NET Framework, the magnet value and the sticky value are high.[b]

yumingzhe@f.ait.kyushu-u.ac.jp
[c] kamei@ait.kyushu-u.ac.jp

(RQ2) How do magnet and sticky values change over time?
We can identify which tags are obsolete. When the tags move quadrant, we find that something happens.

2. Definition of Magnet and Sticky

This section describes how we measure the appeal and adhesion of users on different topics on Stack Overflow in this study, we use the Magnet and Sticky metrics defined by the Pew Research Center for illustrating the migratory trends of citizens in the United States.
The Pew Research Center report defines magnet states as those states where a large proportion of adults who live there have moved from another state. Thus, the magnet metric for a state is the proportion of adult residents of a state who were not born in the state. Furthermore, the report also defines sticky states as those states where a large proportion of adults who were born there continue to live there. Thus, the sticky metric for a state is the proportion of adult residents who were born in the state
These definitions are sound for a study of populations, where a single adult can only occupy one state at a time. However, the definition cannot be applied directly to the topics discussed by the users of Stack Overflow where a user can ask or answer questions on several topics at the same time. Therefore, we expand

	User	2017	2018	Magnet 2017	Magnet 2018	Sticky 2018
TopiclTopic1	A B C D E G	● ● ● ●	● ● ● 	4/4	2/3	1/4
Topic2	B C E F G	● ● 	● ● ✲	2/4	3/3	1/2
Topic3	A B C E F	● ● 	● ● ✲	2/4	2/3	2/2

Fig. 1 –Example of Magnet and Sticky values definition
new definition to apply to topics in Stack Overflow as follows:

Magnet and Sticky in SOtack Overflow

Questions in Stack Overflow
SO contents are composed of the content of the question, questions and answers to the questions and comments [3], that are call] called Posts in the database of SOTorrent database. Each question has one or more tags that separate the question into different topics. Si-multaneously, posts in a Posts on a question have their own creator (for the question content, one is the questioner, and for the answer, one is the respondent) who is), a participant of the topics of, and the question. We also define the activity of asking or answering questions activity on some topic as a discussion of the topic. For example, a classical question in Stack overflowSO has three tags like java, apacheJava, Apache, and Linux which is asked by a user A and answered by a user B, and C, so A, and. Thus, C are participants of topic java, apacheJava, Apache, and Linux. topics.

Magnet. Magnet topics are those that attract a large proportion of new users. Thus; thus, we calculate the magnetism of a topic as the proportion of users who ask or answer questions during the time period under research to all new registered users who registered their account at thein a specific year.

Sticky. StickyIn sticky topics are those where, a large proportion of the users will keep participating in the discussion in the time period under research and the following.. Thus, we calculate the stickiness of a topic as the proportion of the users who discuss within the topic in the time period under research to who have also discussed in the following time period..

Example (Calculating magnet and sticky values). Let us explain how weTo calculate magnet and sticky values for someof topics that belong to a major category as an example. There are, we use a total of six6 questions (a, b, c, d, e, f) and seven7 users (A, B, C, D, E, F, G),); the Last Activity Date of question a, b, c is duringwas in 2017, and question d, e, f is duringin 2018. The registration date of user A, B, C, D is duringwas in 2017, and the registration date of user E, F, G is duringwas in 2018 [2].

To calculate the magnet metric, we observe that there are four new users who register his/her accountregistered their accounts in 2017 (A, B, C, and D), and all of them discuss in topic 1, whilewhereas two of them (B, C) participate in the discussion of topic 2 and 3. In this case, the Magnet value of topic 1 in 2017 is 4/4, topic 2 is 2/4, and topic 3 is 2/4.

Question	Last Activity Date	Tag1	Tag2	Topic1	Topic2
a	2017	1.0		1	
b	2017	1.1		1	
c	2017	1.0	3.1	2	3

Comment [Editor3]: Tip: Serial comma: In American English, a comma (called serial or Oxford comma) is inserted before "and" in a series of three or more items.

d	2018	2.1		2	
e	2018	3.0		3	
f	2018	1.2		1	

Fig. 2 ~~–~~Example of the merge of tags belonging to analogous subjects

To calculate the sticky metric~~,~~ on topic 1, ~~there are~~ three users ~~participate~~participated in the discussion in 2017 (A, B, and C~~) but only~~). Only one of them ~~also participate~~participated in the discussion in 2018 (A). Hence, the sticky value of project 1 is 1 /4. ~~on~~In topic 2, ~~there are 2~~two users ~~participate~~participated in the discussion in 2017 (B and C~~) but~~); however, only one of them ~~also participate~~participated in the discussion in 2018 (B). ~~Even though~~Though new users E, F, G ~~participate~~participated in the discussion in 2018, we still calculate the value of sticky as 1 /2. For the same reason, the sticky of topic 3 is 2/2 in 2018.

Example (Merging similar subjects into one topic). We merge subjects (i.e., tags) ~~belonging~~that belong to analogous subjects into one topic. For example, we consider that different version ~~number suffixes~~numbers (e.g., ~~"python~~the tag "Python-2.7~~"~~" and tag ~~"python~~"Python-3.6~~")~~") are one of the common examples of analogous tags.

We also ~~consider that we~~ need to merge ~~include~~ derivatives of the same technology on different platforms, merge derivatives of special tools in a certain tool family, or a combination of ~~a~~ technology with ~~its commonly~~a common used library, etc. For example, the tag ~~"~~"reactjs~~", "~~," "react-router~~", "~~," "reactjs-flux~~", "~~," "create-react-app~~""~~ should be merged into one topic ~~"~~"react~~"..~~." We can get this information from the ~~"~~"Related Tag~~""~~ column of the ~~"~~"Tag Info~~" on Stack Overflow."~~

~~" of SO."~~ Figure 2 shows that question a has tag 1.0, question b has tag 1.1, and question f has tag 1.2~~, which means according~~. According to our merge rule, they all belong to topic 1. ~~Simultaneously,~~The question c has tag 2.0 and tag 3.1, ~~which means~~showing that it belongs to topic 2 and 3~~ at the same time.~~. Therefore, ~~queston~~question d belongs to topic 2, and question e belongs to topic 3.

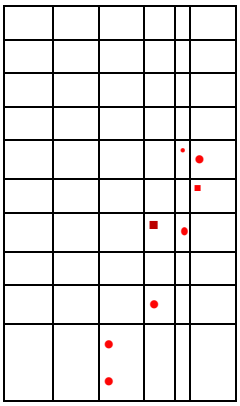
3. Dataset

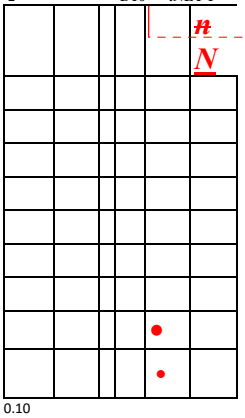
~~In this paper, we~~We analyze the ~~Stack Overflow~~SO dataset (~~SOTor rent~~SOTorrent) provided by Sebastian Baltes et al. [4]. SOTorrent is an open dataset based on the official SO data dump. "SOTorrent provides access to the version history of SO content at the level of whole posts and individual text or code blocks~~,~~." The dataset ~~includes~~consists of 20 different tables ~~which store not only~~stored in data ~~from~~on official SO data dump ~~but also~~and data extracted from the original official SO data dump.

~~In this paper~~ However, we only analyze the data from table *Posts* ~~which includes about~~of approximately 42 million posts from ~~Stack overflow~~SO and table *Users* ~~where there are about~~of approximately 9 million rows of user information from July 2008 to September 2018. We focus on users, tags, and time ~~information~~ of questions. Moreover, we consider users who ask or answer questions in SO. Those who comment or like/dislike questions or answers are excluded from the statistics.

~~In this paper, we consider a user to be one who asks or answers questions in Stack Overflow. Those who comment or like/unlike~~

Comment [A4]: Remark: Please consider providing a suitable citation at this instance. These texts are quoted because they are lifted here: https://ieeexplore.ieee.org/document/8595215





3

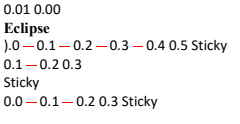


Fig. 3 –Distribution of Magnet and Sticky values in ~~Prosgraming~~Programming Language, Framework and Environment
~~on questions or answers are not counted in the statistics.~~

4. –Study Results

~~We set research results and faced two questions against these results. We discuss the questions based on the results.~~

Comment [A5]: Remark: Note that the table may have been distorted because of file conversion. Please insert the file before submission.

Comment [A6]: Remark: Note that the table may have been distorted because of file conversion. Please insert the file before submission.

In this section, we provide answers to these questions:

4.1 ~~(~~(RQ1) What are ~~typical~~the values of magnet and sticky in Stack Overflow?

Approach. We ~~have~~ calculated magnet and sticky values as defined in Section 2. We plot the magnet value on the vertical axis and the sticky value on the horizontal axis. We classify the plotted points into four~~4~~ quadrants.

Attractive: ~~Tag~~Tags with ~~the~~a high magnet and sticky value. By ~~knowing attractive~~ understanding the tags, we can ~~find out what~~discover the interests of developers ~~are interested in~~.

Fluctuating: ~~Tag~~Tags with ~~high~~a higher magnet and ~~low~~lower sticky value. This tag attracts people, but it is short-term. Excellent developers will ~~not continue to~~ be ~~interested~~disinterested.

Stagnant: ~~Tag~~Tags with ~~the low~~a lower magnet and ~~high~~higher sticky value. These tags are difficult to attract new users, but it can maintain existing users. **Terminal:** ~~Tag~~Tags with the ~~Low~~lower magnet and ~~low~~lower sticky value. This tag can neither attract new ~~users'~~ developers nor keep them interested.

~~In this paper, the~~The median of the magnet and sticky values for each year is ~~used~~ for the quadrant threshold ~~of the quadrant~~ because the median ~~value~~ is ~~not much affected~~unaffected by outliers. ~~As we showed~~From the sticky value definition in Section 2, the sticky value depends on the number of tag users in ~~that~~a year and the following year. ~~So in order to~~To answer RQ1, we got 9 ~~years'~~years' worth of sticky ~~value from~~values based on the information on the number of tag users from 2009 to 2018. The sticky value must depend on the number of new tag users, but~~,~~ if the number of new tag users in ~~the target~~a year is too low, the sticky value will be too small. ~~Therefore, in order to~~To remove noise, we ~~decided~~fix thresholds for each topic ~~and when.~~ When the magnet and sticky value is less than the threshold value, it is set to 0.

We did not analyze all the tags at once, but divided them into three categories for analysis. The selected categories ~~we selected~~ and their contents are:

• programming languages (assembly, Bash, C, C #, C ++,
CSS, Go, HTML, Java, JavaScript, PHP, Python, Ruby, SQL, Swift, TypeScript)
• frameworks ~~(~~.(.NET Framework, Angular, Cordova, Django, Hadoop, Node.js, React, Spark, Spring, TensorFlow, Torch, Xamarin)
• environment (Android Studio, Atom Editor, Eclipse, Emacs, IntelliJ, IPython, Jupyter, NetBeans, Notepad++, PhpStorm, PyCharm, RStudio, RubyMine, Sublime Text, TextMate, Vim, Visual Studio, Visual Studio Code, Xcode)
~~PyCharm, RStudio, RubyMine, Sublime Text, TextMate,~~
~~Vim, Visual Studio, Visual Studio Code, Xcode)~~

We chose these tags based on ~~Stack Overflow's~~SO's survey of over 100,000 developers in 2018 ~~^~~,. We focused on tags used by more than 5% of developers who answered the questionnaire.

Results: Figure 3 shows a quadrant plot of the magnet and sticky values of the 2010 framework, programming language, and environmental tag1[2]. ~~We can see,~~ revealing that the magnet value is lower than the ~~value of~~ sticky. ~~This is~~ value. The results are similar to the findings ~~with the investigation~~ of the PRCew Research Center. ~~Like~~For example, the U.S. citizens are ~~more~~ likely to ~~spend more time living on~~live in the same ~~land~~house than to change houses.~~;~~ it is easier for developers to continue developing ~~about~~ the same content.

Summary. Tags with high magnet value are easy to use even for beginners. A tag ~~that is~~ familiar ~~from old days like~~to Java has a high magnet value and sticky value, and it is more attractive.

4.2 ~~(~~(RQ2) How do magnet and sticky values change over time?

Approach: From 2010 to 2018, we calculated the probability ~~that~~of the tags ~~move~~moves quadrants from one year to the following year. For example, there ~~are~~were six Attractive tags in 2010. Of the six Attractive tags ~~that were Attactive~~ in 2010, ~~there are~~ five ~~that~~ were Attractive the following year ~~as well~~. Therefore, the transition probability from
Attractive to Attractive for 2010 - 2011 is 5/6 or 83.3%.

Quantitative results: Table 2 shows that the proportion ~~that~~of the vertical axis is in ~~that~~the quadrant for ~~some years up to~~ 2010~~–~~2018, the horizontal axis is ~~that~~the year old. From the table, ~~we can find that~~ the ratio of tags is the highest for those that do not move the quadrants from the previous year to the following year ~~is the highest~~ in any ~~field of~~ programming language, framework, and environment. Since the tags have hardly changed from any quadrant to 2, once the tags have become popular to a certain extent, the users of the tags have not significantly reduced. This shows that once tags have become less popular, it may continue to be unpopular.

~~*[1] https://insights.stackoverflow.com/survey/2S18~~

~~*[1] https://insights.stackoverflow.com/survey/2S18~~

Manual analysis: Table 1 shows the transition of each tag quadrant in the framework, revealing how the tags move in the quadrant. Xamarin is an interesting example. Xamarin is an API for Android and iOS developed with C #. Thus, it is difficult to develop an application without having knowledge of both applications. Developing Android and iOS apps in Windows and Visual Studio requires a lot of programming knowledge and is not good for beginners. When Xamarin was launched, it attracted the attention of many developers owing to its efficiency. However, when beginners ask questions on sites such as SO, its popularity gradually declines owing to its application difficulty for beginners.

Similarly, React is a Facebook JavaScript library that builds the web application user interface efficiently. Riact was first launched on Facebook's news feed in 2011 and on Instagram in 2012. It was an open source at the JSConf US on May 2013. Social networking services (SNSs) such as Facebook and Instagram became popular around 2015–2016 when React changed from Terminal to Floating. Therefore, it seems that the popularity of the Framework changed based on its application on the popular SNSs.

Summary: If tags were the popular tool, their popularity would decline if they were difficult to use. Even if they were not a popular tool and they turn out to be an efficient tool, they will be popular among developers.

Comment [A7]: Remark: Note that the corresponding text has been heavily edited. Please verify whether the edits convey the desired meaning.

[1²] — We choose the year 2010 because it is the first year for which yearly data of sticky value can be obtained.
[2²] —We choose the year 2010 because it is the first year for which yearly data of sticky value can be obtained.

Table 1 –Quadrant Transition of Framework 2010 - 2018

Programing Language	2010	2011	2012	2013	2014	2015	2016	2017	2018
assembly	Terminal	Terminal	Terminal	Terminal	Terminal	Terminal	Terminal	Terminal	Terminal
Bash	Terminal	Terminal	Terminal	Terminal	Terminal	Terminal	Terminal	Terminal	Terminal
C	Fluctuating	Terminal	Terminal	Terminal	Terminal	Terminal	Terminal	Terminal	Terminal
C#	Attractive	Attractive	Attractive	Attractive	Attractive	Attractive	Attractive	Attractive	Attractive
C++	Attractive	Attractive	Attractive	Attractive	Attractive	Stagnant	Stagnant	Stagnant	Stagnant
CSS	Terminal	Stagnant	Stagnant	Terminal	Terminal	Stagnant	Stagnant	Terminal	Terminal
Go	*	*	Terminal	Terminal	Fluctuating	Fluctuating	Fluctuating	Fluctuating	Fluctuating
HTML	Attractive	Attractive	Attractive	Attractive	Stagnant	Stagnant	Stagnant	Stagnant	Stagnant
J ava	Attractive	Attractive	Attractive	Attractive	Attractive	Attractive	Attractive	Attractive	Attractive
J avaScript	Attractive	Attractive	Attractive	Attractive	Attractive	Attractive	Attractive	Attractive	Attractive
PHP	Attractive	Attractive	Attractive	Attractive	Attractive	Attractive	Attractive	Attractive	Attractive
Python	Attractive	Fluctuating	Fluctuating	Attractive	Attractive	Fluctuating	Fluctuating	Attractive	Attractive
Ruby	Terminal	Fluctuating	Fluctuating	Fluctuating	Fluctuating	Fluctuating	Terminal	Terminal	Terminal
SQL	Stagnant	Stagnant	Stagnant	Stagnant	Stagnant	Stagnant	Stagnant	Stagnant	Stagnant
Swift	*	*	*	*	Terminal	F luctuating	F luctuating	F luctuating	Fluctuating
TypeScript	*	*	*	Terminal	Terminal	Terminal	Fluctuating	Fluctuating	Fluctuating

Table 2 Average Quadrant Transition rate **Language**

	Attractive	Fluctuating	Stagnant	Terminal	*
Attractive	92.3	3.9	3.9	0	0
Fluctuating	9.4	75.0	0	15.6	0
Stagnant	0	0	90.6	9.4	0
Terminal	0	9.8	4.6	85.6	0
*	0	0	0	45.8	54.2

Framework

	Attractive	Fluctuating	Stagnant	Terminal	*
Attractive	84.6	4.2	11.3	0	0
Fluctuating	0	70.2	14.3	15.5	0
Stagnant	12.5	0	87.5	0	0
Terminal	0	10.4	0	85.4	4.2
*	0	26.7	0	6.7	66.7

Environment

	Attractive	Fluctuating	Stagnant	Terminal	*
Attractive	80.9	2.1	17.0	0	0
Fluctuating	6.3	84.7	0	9.1	0
Stagnant	11.3	0	80.0	8.8	0
Terminal	0	20.8	17.5	61.7	0
*	10.0	13.3	3.3	15.0	58.3

been significantly reduced since then. It also shows that once tags have become less popular, it will be difficult to become popular again.
Manual analysis: Table 1 shows the transition of each tag quadrant in the framework. From this table we can see how the tags move in the quadrant. Here we turn to Xamarin as an interesting example. Basically, Xamarin is an API for Android and iOS that can be called from C #, so you can not develop an application without knowing both details. This means that if you start developing Android and iOS apps with Windows and Visual Studio you will need a lot of knowledge, which is not good for beginners. When Xamarin first appeared, it attracted attention as a tool that developers can efficiently develop. However, on sites where beginners often ask questions, such as StackOverflow, its popularity seems to have gradually declined due to its use difficulty.
About Riact, this is a JavaScript library from Facebook. It aims to build the user interface of web application efficiently. It was first used on Facebook's news feed in 2011 and in 2012 on Instagram. It was open sourced at JSConf US in May 2013. SNSs such as FaceBook and Instagram have begun to become popular around 2015-2016, when React has changed from Terminal to Floating. Therefore, it seems that the popularity of Framework that builds it to popular SNS has also changed.
Summary: Even if it was a tool that was initially popular, its popularity would decline if it was difficult to use. Even if it was a tool that was not as popular as it was born, It turns out that as the content using the tool becomes popular, the tool also becomes popular

5. Conclusions

Whether it is
Critical development of a programming language or, a program framework, or an operating system, keeping depends on their ability to keep the community alive and attractingattract more people to participate in discussions is critical to its development. Especially on . SO, the stack overflow, the world'sworld's largest program Q&A platform, having morehas enough questions and answers on a topic means that customers of the product are more likely to solve their ownusers' problems, which is even more tedious than that developers rack their brains to write a lengthy development document or Q&A. This paperstudy applied the magnet and sticky population concepts to a set ofexplore topics in Stack Overflow. We find
that:
1SO. The number of results show that the numbers of participating topics that people participate in is are exploding with the development and popularity of computer technology. Even the most popular themes cannot attract the high percentage of people involved in the discussion like what they that did tennot attract people's attention 10 years

ago.
2. Under their now attract a large number of participants. Under respective major categories, the most popular topics are still very popular after tenten years, and only a small number of languages or frameworks can stand out and become one of the most popular topics.
3. This research can provide someprovides a reference for enterprises to choose their own main technology stack, and. It can also be used as a reference for computer science students to learn new technologies, because it. The study (1) predictscan predict the trend of computer technology in the next few years. and (2) points out whichcan identify easier technologies are easier totto access the questions and answers.

References

[1] –C. for Community and C. E. Economic Development University –of -Wisconsin-Extension, —"“Community –& – economic development –update-,", — 2011, ~~https://myemail.constantcontact.com/~~ https://myemail.constantcontact.com/ News-from-the-Center-for-Community---Economic-Development. html?soid= 1104293309477\&aid=FnOfZtbhFgo.
[2] K. Yamashita, Y. Kamei, S. McIntosh, A. E. Hassan, and N. Ubayashi, "“Magnet or sticky? measuring project characteristics from the perspective of developer attraction and retention-,", *Journal of Information Processing,* vol. 24, no. 2, pp. 339-348, 2016.
[3] X. Liu and H. Zhong, "“Mining stackoverflow for program repair-,", in *2018 IEEE 25th International Conference on Software Analysis, Evolution and Reengineering (SANER)*. –IEEE, 2018, pp. 118-129.
[4] S. Baltes, C. Treude, and S. Diehl, "“Sotorrent: Studying the origin, evolution, and usage of stack overflow code snippets-,", 2019.