Journal of Information Processing Vol.26 1–5 (Jan. 2018)

[DOI: 10.2197/ipsjjip.26.1]

Regular Paper

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

Kotori Hieda' Yu Mingzhe Yasutaka Kamei

Received: March 4, 2016, Accepted: August 1, 2016

Abstract: Stack Overflow (SO) [1] is one of the most popular question and answer website among software developers. There are many tags in Stack Overflow. If you ask about Python and tag that post with "Python", people interested in Python can follow that tag and see your post. Many questions and answers are received every day for Stack Overflow. We explored how their interest shifted from how they use tags. We classified 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 table "Posts" which includes about 42 million posts from Stack overflow and table "Users" where there are about 9 million rows of user information [2]. Analysis of stack overflow revealed that: (1) There were some historical events in IT such as the launch of new tools and the termination of services when there were characteristics in the transition of magnet value and sticky value. (2) The types of tags that are classified do not change much.

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

1. Introduction

Pew Research Center [3], an organization that studies problems in 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 are defined as *magnet states* and states where the high proportion of the population who continues to live in the same state since birth are defined as *sticky states*. For example, 86% of Nevada's population migrated from other states so Nevada state is quite *magnet*. Through such a survey, you can find the tendency how American citizens move.

For many developers, it is important to know the changes in the interests of other developers because popularity among many people should have advantages. Developers always want to work with convenient and easy-to-use tools. To make a project better, excellent developers need to be interested in the project over the long term.

According to the definition of magnet and sticky to be defined later, we classified tags used for stack overflow into magnet tags and sticky tags. In addition, we define strong magnetic and strong sticky *attractive*, strong magnet weak sticky *stagnant*, weak magnet strong sticky *fluctuating*, weak magnet defines a weak sticky *terminal*. By classifying like this, we analyzed what we know from the popularity of tags. We examined tags' magnets and sticky values by classifying them as the tags *programming language*, *framework*, and *environment*. We also compared the news and history of software 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.

(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 new definition to apply to topics in Stack Overflow as follows:

Magnet and Sticky in Stack Overflow

Questions in Stack Overflow are composed of the content of the question, answers to the questions and comments [4], that are

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

Fig. 1 Example of the merge of tags belonging to analogous subjects

call Posts in the database of SOTorrent. Each question has one or more tags that separate the question into different topics. Simultaneously, posts in 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 the question. We also define the activity of asking or answering questions on some topic as a discussion of the topic. For example, a classical question in Stack overflow has three tags like java, apache, and Linux which is asked by user A and answered by user B, and C, so A, and C are participants of topic java, apache, and Linux.

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

Sticky 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 Let us explain how we calculate magnet and stick values for some topics belong to a major category as an example. There are 6 questions (a, b, c, d, e, f) and 7 users (A, B, C, D, E, F, G), the Last Activity Date of question a, b, c is during 2017 and question d, e, f is during 2018. The registration date of user A, B, C, D is during 2017 and the registration date of user E, F, G is during 2018 [5].

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

To calculate the sticky metric, on topic 1, there are three users participate in the discussion in 2017 (A, B and C) but only one of them also participate in the discussion in 2018 (A). Hence, the sticky value of project 1 is 1/4. on topic 2, there are 2 users participate in the discussion in 2017 (B and C) but only one of them also participate in the discussion in 2018 (B). Even though new users E F G participate 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.

	User	2017	2018	Magnet 2017	Magnet 2018	Sticky 2018
Topic1	A	●	●	4/4	2/3	1/4
	B	●	●			
	C	●	●			
	D	●				
	E					
	G					
Topic2	B	●	●	2/4	3/3	1/2
	C	●	●			
	E		●			
	F		●			
	G					
Topic3	A	●	●	2/4	2/3	2/2
	B	●	●			
	C		●			
	E		●			
	F		●			

Fig. 2 Example of Magnet and Sticky values definition

We merge tags belonging to analogous subjects into one topic,for example tag "python-2.7" and tag "python-3.6" are merged into topic "python". So we get 3 topics in this case. Of course different version number suffixes are the most common examples of analogous tags. In our observations, analogous tags that also need to be merged include derivatives of the same technology on different platforms, special tools in a certain tool family, or a combination of a technology with its commonly used library, etc. For example, 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."

In our exmaple which is presented in Figure 1 and Figure 2, question a has tag 1.0, question b has tag 1.1 and question f has tag 1.2 , which means according to our merge rule, they all belong to topic 1. Simultaneously, question c has tag 2.0 and tag 3.1, which means it belongs to topic 2 and 3 at the same time. Therefore, queston d belongs to topic 2 and question e belongs to topic 3.

3. Dataset

In this paper, we analyze the Stack Overflow dataset provided by Sebastian Baltes et al [6] called SOTorrent.

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 20 different tables which store not only data from official SO data dump but also data extracted from the original official SO data dump.

In this paper, we only analyze the data from table *Posts* which includes about 42 million posts from Stack overflow and table *Users* where there are about 9 million rows of user information

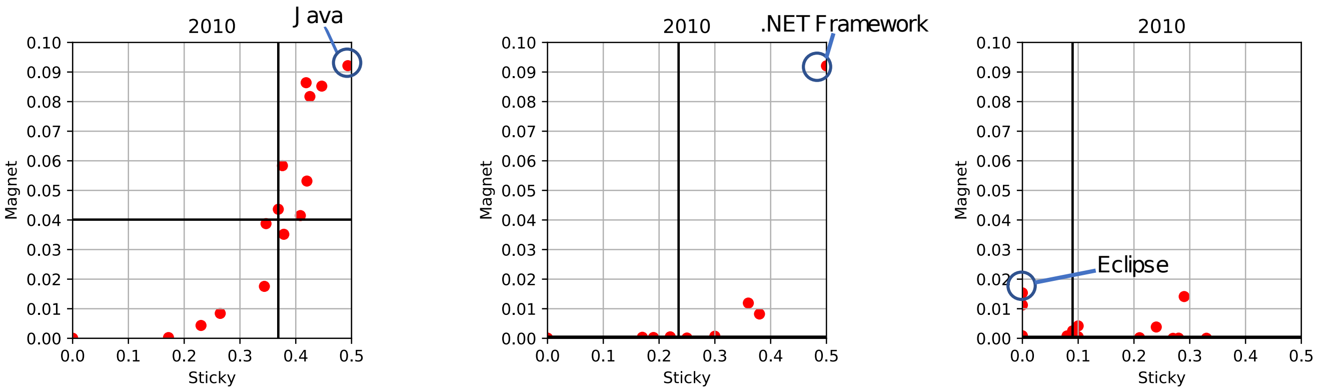

Fig. 3 Distribution of Magnet and Sticky values in Prosgraming Language, Framework and Environment

from July 2008 to September 2018 and we pay attention to users, tags, and time information of questions.

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

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

We have calculated magnet and sticky values as defined in section 2. And we plotted the magnet value on the vertical axis and the sticky value on the horizontal axis. We classified the plotted points into 4 quadrants.

Attractive: Tag with the high magnet and sticky value. By knowing attractive tags, we can find out what the developers are interested in.

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

Stagnant: Tag with the low magnet and high sticky value. These tags are difficult to attract new users but maintain existing users.

Terminal: Tag with the Low magnet and low sticky value. This tag can neither attract new users' developers nor keep them interested.

In this paper, the median of the magnet and sticky values for each year is used for the threshold of the quadrant because the median value is not much affected by outliers. As we showed the sticky value definition in section 2, the sticky value depends on the number of tag users in that year and the following year. So in order to answer RQ1, we got 9 years' worth of sticky value from 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 year is too low, the sticky value will be too small. Therefore, in order to remove noise, we decided thresholds for each topic and when the magnet and sticky value is less than the threshold value, it is set to 0. We did not investigate all the tags at once, but analyzed it in three categories. That is because it makes it easier to find features.

We did not analyze all the tags at once, but divided them into three categories for analysis. It is because it is easy to find the feature. The 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), and 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). We chose these tags based on Stack Overflow's survey of over 100,000 developers in 2018. We focused on tags used by more than 5% of developers who answered the questionnaire. The year when StackOverflow released these data is May 2008. Since the sticky value is calculated from the difference between the previous year and the current year, the distribution chart of magnet and sticky value according to three categories in 2010 is shown as the first year for which yearly data of sticky value can be obtained.

Manual analysis: From Figure 3, .NET Framework is a high magnet and sticky value. It can be said that it is very good to be a high magnet and sticky value. This is because the magnet value of .NET that was on the net is high because beginner developers can develop somewhat advanced software. Are there many reasons why the .NET Framework's sticky value is high reasons most conveniently? It is the foundation system for building applications. Java has long been popular and attractive as it is one of the most famous programming languages in the world. Eclipse especially attracted people in 2010, so the magnet value is high.

Quantitive results: Figure 3 shows a quadrant plot of the magnet and sticky values of the 2010 framework, programming language, environmental tag from the left. From the values on the vertical axis and the horizontal axis in Figure 3, we can see that the magnet value is lower than the value of sticky. In other words, it is more common that the tag used in the overflow is not a magnet but a tag. This is also in the investigation of the Pew Research Center mentioned earlier. Like citizens are more likely to spend more time living on the same land than to change houses, it is eas-

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
Java	Attractive	Attractive	Attractive	Attractive	Attractive	Attractive	Attractive	Attractive	Attractive
JavaScript	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	Fluctuating	Fluctuating	Fluctuating	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

ier for developers to continue developing about the same content. —

Tags with high magnet value are easy to use even for beginners. A tag that is familiar from old days like Java has a high magnet value and sticky value and is attractive. Also from the eclipse example, the trend of tags can often be affected by something unrelated to itself.

(RQ2) How do magnet and sticky values change over time?
Approach: From 2010 to 2018, we calculated the probability that the tags move quadrants from one year to the following year. For example, there are six Attractive tags in 2010. Of the six 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%. Table 2 shows the average of this transition probability from 2010 to 2018.

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.

Quantitive results: Table 2 shows the proportion that the vertical axis is in that quadrant for some years up to 2009-2018, the horizontal axis is that year old. From the table, it can be seen that the ratio of tags that do not move the quadrants from the previous year to the following year is the highest in any field of programming language, framework, environment. Since the tags have hardly changed from any quadrant to *, once the tags have become popular to a certain extent, the users of the tags have not been significantly reduced since then. It also shows that once tags have become less popular, it will be difficult to become popular again.

Table 1 shows that 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. Table 2 shows that it is difficult for tags to make large changes in quadrants.

5. Conclusions

Whether it's a programming language or a program framework or an operating system, keeping the community alive and attracting more people to participate in discussions is critical to its development. Especially on the stack overflow, the world's largest program Q&A platform, having more questions and answers on a topic means that customers of the product are more likely to solve their own problems, which is even more tedious than that developers rack their brains to write a lengthy development document or Q&A This paper applied the magnet and sticky population concepts to a set of topics in Stack Overflow. We find that:

1. The number of topics that people participate in is 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 did ten years ago.

2. Under their respective major categories, the most popular topics are still very popular after ten 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 some reference for enterprises to choose their own main technology stack, and can also be used as a reference for computer science students to learn new technologies, because it (1) predicts the trend of computer technology in the next few years, (2) points out which technologies are easier to access and the questions can be easier to get answers to.

References

[1] S. M. Nasehi, J. Sillito, F. Maurer, and C. Burns, "What makes a good code example?: A study of programming q&a in stackoverflow," in *2012 28th IEEE International Conference on Software Maintenance (ICSM)*. IEEE, 2012, pp. 25–34.

[2] S. Baltes, C. Treude, and S. Diehl, "Sotorrent: Studying the origin, evolution, and usage of stack overflow code snippets," *arXiv preprint arXiv:1809.02814*, 2018.

[3] C. for Community and C. E. Economic Development University of Wisconsin-Extension, "Community & economic development update," 2011, https://myemail.constantcontact.com/News-from-the-Center-for-Community—Economic-Development.html?soid=1104293309477&aid=FnOfZtbhFgo.

[4] 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.

[5] 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.

[6] S. Baltes, C. Treude, and S. Diehl, "Sotorrent: Studying the origin, evolution, and usage of stack overflow code snippets," 2019.