What's Hot in Software Engineering Twitter Space?

Abhishek Sharma, Yuan Tian, and David Lo
School of Information Systems
Singapore Management University
{abhisheksh.2014,yuan.tian.2012,davidlo}@smu.edu.sg

***Abstract*—Twitter is a popular means to disseminate information and currently more than 300 million people are using it actively. Software engineers are no exception; Singer et al. have shown that many developers use Twitter to stay current with recent technological trends. At various time points, many users are posting microblogs (i.e., *tweets*) about the same topic in Twitter. We refer to this reasonably large set of topically-coherent microblogs in the Twitter space made at a particular point in time as an *event*.**

In this work, we perform an exploratory study on software engineering related events in Twitter. We collect a large set of Twitter messages over a period of 8 months that are made by 79,768 Twitter users and filter them by five programming language keywords. We then run a state-of-the-art Twitter event detection algorithm borrowed from the Natural Language Processing (NLP) domain. Next, using the open coding procedure, we manually analyze 1,000 events that are identified by the NLP tool, and create eleven categories of events (10 main categories + "others"). We find that external resource sharing, technical discussion, and software product updates are the "hottest" categories. These findings shed light on hot topics in Twitter that are interesting to many people and they provide guidance to future Twitter analytics studies that develop automated solutions to help users find fresh, relevant, and interesting pieces of information from Twitter stream to keep developers up-to-date with recent trends.

***Index Terms*—Twitter; Event Detection; Exploratory Study; Categorization**

I. Introduction

Twitter is currently the most popular microblogging service in the world. Apart from using Twitter to connect with friends and family, people also use Twitter daily to share news and knowledge and discover latest information and updates about various topics of interest. Recent studies have found that software developers also use Twitter for their personal, as well as professional pursuits. Singer et al. [1] survey 271 GitHub developers and interview 27 of them to better understand their Twitter usage. They find that software developers use Twitter quite extensively in their professional activities. Developers use Twitter to stay aware of the latest software trends and practices, to extend their software knowledge by learning new stuffs and to maintain relationships with fellow software developers.

In this work, we extend Singer et al.'s study by investigating *events* in software engineering Twitter space. An event corresponds to a set of topically-coherent microblogs that are shared by many Twitter users at a point in time. It can be viewed as a popular *trending topic* in the software engineering Twitter space happening at a particular point in time. By studying events, we can discover *hot* topics that interest many developers. Different from Singer et al.'s work which focuses on getting insight of developer's use of Twitter by interviewing developers, this work analyzes contents of tweets about popular topics that developers generate. Moreover, while Singer et al. reported that developers use Twitter to get up-to-date with the latest trends and consume knowledge, this work drills deeper by investigating the kinds of popular trends and knowledge that get disseminated widely.

To find events, we first monitor a set of 79,768 users who are potentially interested in software development. We collect microblogs that are generated by these users over an 8 month period which amounts to 48,889,030 microblogs in total. Next, we need to identify software engineering related tweets among these 48.9 millions tweets. Unfortunately, this will require a prohibitive amount manual effort since automated solutions still cannot identify software engineering tweets with high accuracy (c.f., [2], [3]). To make the identification of software engineering related tweets practical, in this work we only study microblogs that mention each of the following popular programming languages, i.e., C#, Java, Python, Scala and Ruby. We leave the study of other software engineering related tweets as future work. We then apply a state-of-the-art Twitter event detection algorithm [4] on each of the five sets of microblogs to find events related to each of these programming languages.

We sort the identified events based on their popularity (i.e., number of tweets involved in the event), and selected the top 200 events for each of the programming language. These 1,000 events are then manually analyzed using the open coding procedure [5], [6] to create event categories. We then investigate three research questions: 1) What are some hot software engineering related events in Twitter space? 2) What are the categories of software engineering related events in Twitter space? 3) How hot is each event category?

Our study is the first step towards a deeper understanding of tweets that interest developers. A good understanding of interesting tweets will guide our future work on the construction of a recommendation system that can highlight fresh, relevant, and interesting pieces of information from the Twitter stream to keep developers up-to-date with recent trends and gain new knowledge. Such a solution will address challenges that prevent developers from using Twitter, e.g., information overload, etc. [1].

The contributions of this work are as follows:

1) We are the first to investigate events or trending topics that appear in the software engineering Twitter space.
2) We perform an open coding procedure on 1,000 events to group them into categories, and answer three research questions that shed light to topics that interest many people in software engineering Twitter space.

The structure of the remainder of the paper is as follows. In Section II, we describe the methodology that we follow in this exploratory study. We describe findings of our study in Section III. In Section IV, we discuss related work, and finally we conclude and mention future work in Section V.

II. Methodology

Figure 1 shows the methodology we follow in our study which contains 3 major steps: Twitter Data Extraction, Event Identification, and Open Coding.

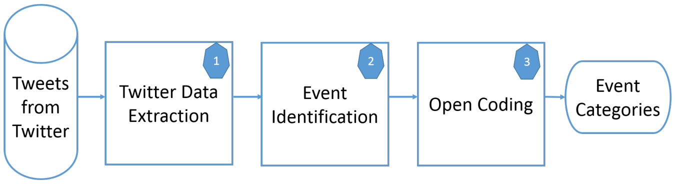


Fig. 1. Empirical Study Methodology

In Step 1, we identify a set of 79,768 Twitter users who are potentially interested in software development. This identification was done following the methodology used in our previous work [7], [2]. We start with a seed set of 100 users who are well-known in software development[1] and include other users that follow or are followed by at least five of the seed users. We then periodically crawl all tweets of these users for an 8 month period between September 2012 to April 2013 and we collect in total of 48,889,030 tweets. From these tweets, we identify 5 sets of tweets; each set consists tweets that mention one of the following 5 programming languages: C#, Java, Python, Ruby and Scala. After this keyword filtering process, we have sets of C#, Java, Python, Ruby, and Scala tweets of sizes 27,102, 117,385, 54,862, 104,528 and 35,634 respectively. At the end of this step, we order tweets in each set based on the time they were posted.

In Step 2, for each series of tweets for a programming language extracted in Step 1, we identify events in them by running a state-of-the-art Twitter event detection technique by Diao and Jiang [4]. Diao and Jiang design a unified model which combines a topic model to represent user interest, a dynamic non-parametric model to represent events, and a probabilistic matrix factorization component to capture the relationship between events and topics. The unified model tries to separate personal from event-related tweets, identify tweets that belong to the same event, and penalize long-term events since most events are short-lived. The precision of Diao and Jiang's technique has been shown to be high (precision@5=100% and precision@30=90%). After processing each of the 5 series of tweets, the technique produces a ranked list of the events and we take the top-200 events sorted based on their sizes (i.e., the number of tweets in the event). At the end of this step, we have a set of 1,000 events that we need to group into categories.

In Step 3, we follow the open coding procedure [5], [6] to generate event categories. Open coding is performed in three iterations. In the first iteration, each event and tweets contained in it is read and a short code (i.e., description) is assigned to it. In the second iteration, the codes are analyzed to create higher-level concepts by merging similar codes together. In the final iteration, the concepts are analyzed to create a small set of categories. After the categories have been identified, the categories are sorted based on how "hot" they are (i.e., how many events belong to each category). The first author performs most of the labeling work, while the second and third authors review the first author's work.

III. Results

Our study aims at answering the following three research questions. This knowledge can be used to build an automatic recommendation system which can find interesting software engineering related events from Twitter stream.

A. *RQ1: What are some hot software engineering related events in Twitter space?*

By answering this question, we want to highlight some of the hot or popular software engineering events for the time period we consider. Table I shows sample hot events found for each of the 5 languages we consider.

The example hot event for Java is a security bug that affected Java based web browsers in Jan 2013 which was shared by at least[2] 369 Twitter users. This was an important advisory to developers as well as general public to disable or not use Java based web browsers until the issue is resolved. For C#, the hot event is the release of a viral blog specifying benefits of using C# for mobile development, which was shared by at least 181 Twitter users. This blog may inspire many developers who work on mobile development to use C# rather than other languages. For Python, it is a trademark dispute which was shared by at least 382 Twitter users. This dispute was a call-to-arms for Python developers and enthusiasts to join forces in a legal battle to keep the name Python. For Ruby, it is the release of Ruby 2.0.0, an event that many Ruby developers were likely to be waiting for, and this event was shared by at least 842 Twitter users. For Scala, the joining of Rod Johnson (creator of Java Spring framework) to Typesafe Inc. (the firm mainly responsible for pushing Scala's commercial adaptation) is a hot event. This event was shared by at least 150 Twitter users. This event was an exciting news for Scala developers and it may motivate many other developers to learn and use Scala.

[1] http://noop.nl/2009/02/twitter-top-100-for-software-developers.html

[2] We only monitor a subset of all Twitter users.

TABLE I
HOT SOFTWARE EVENTS FOR EACH PROGRAMMING LANGUAGE

Language	Date of First Tweet	Event Description	Sample Tweets	Tweet Count
Java	11/01/2013	Security vulnerability found in Java	• Feds warn PC users to disable Java • security vendors warn users to disable java after zero day exploit is found	369
C#	02/01/2013	A blog specifying reasons why C# is the best language for mobile development was posted.	• post by xamarin why c# is the best language for mobile development • They've got a horse in the race, but yes. RT @xamarinhq: Eight reasons C# is the best language for mobile development	181
Python	14/02/2013	A company in United Kingdom applied to trademark "Python" for all software and services.	• unbelievable, some random software company in Europe is trying to trademark "Python" • Python trademark at risk in Europe: We need your help!	382
Ruby	24/02/2013	Ruby 2.00 was released	• Ruby 2.0.0-p0 was released • Ruby 2.0.0-p0 is released Come and get it! Boosts to language support, performance, debugging, and built in libs.	842
Scala	01/10/2012	Rod Johnson, creator of Spring Framework in Java joining Typesafe Inc.(founded by authors of Scala team)	• Excited to be getting involved with @typesafe. I love Scala more and more and it's a gr8 team with Martin, Jonas & crew & now Mark Brewer • Proud to welcome Rod Johnson (@springrod) to the Typesafe board: @typesafe #akka #scala #playframework	150

B. RQ2: What are the categories of software engineering related events in Twitter space?

In this research question we want to group software engineering events into categories. Following the open coding procedure described in Section II, we have been able to determine 11 categories of events (10 main categories + "Others") in Table II.

Tweets occurring in category *Article and Multimedia Sharing* such as: "*functional programming principles in scala starts again next monday still time to enroll*" helps interested developers by exposing them to learning resources available. For category *Technical Discussion* tweets such as: "*Scala protip lazy val is not free (or even cheap). Use it only if you absolutely need laziness for correctness, not for optimization*" can help developers in their programming activities. Tweets like: "*Feels so good @ScalaIDE: Scala IDE 3.0.0 is out With semantic highlighting, Scala debugger*" occurring in *New Releases* category can be extremely helpful for developers who are on lookout for such tool. "*Dropbox Hires Away Googles Guido Van Rossum, The Father Of Python*" is an example of a tweet in the category *News*, which the users may find interesting. Tweets such as: "*ebook dealday think python $1599 save 50% use code deal*" in the category *Product Promotions* help developers to be aware of latest deals and promotion on books and products they might be interested in. "*I'll be kicking off Build with style and rocking C# on 2.5 billion devices at @xamarinhqs #bldwin Welcome Party!*" is an example of tweet in the category *Community events*. This was used to attract and encourage software developers to join *Microsft Build Devekoper Conference*. Tweets in *security updates* category such as "*sql injection vulnerability in ruby on rails affects all versions*" help to quickly disseminate security related information to developers. Additionally, tweets such as: "*Seeking Contributors for the Facebook C# SDK*" in the *Crowdsourcing Request* category can be extremely helpful for developers looking for such opportunities. Similarly tweets that fall under the *Career* category can be helpful to software developers hunting for jobs, while tweets that fall under the *Satires* category can help developers to de-stress.

TABLE II
CATEGORIES OF EVENTS IN SOFTWARE ENGINEERING TWITTER SPACE

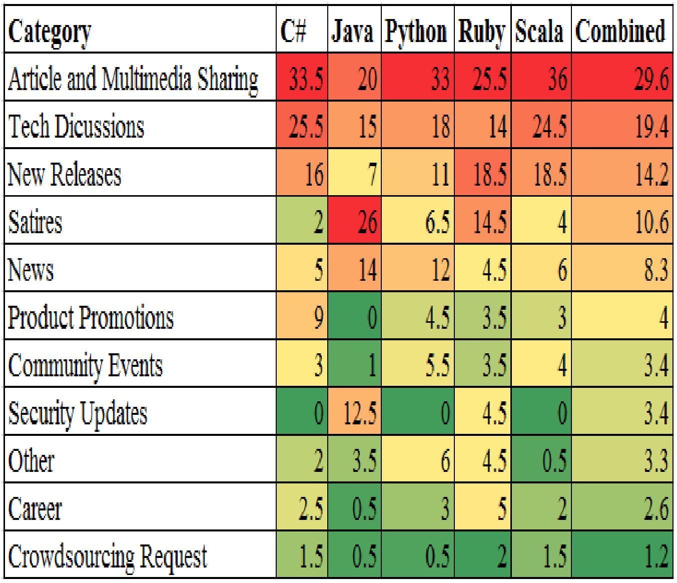
Category Name	Description
Article and Multimedia Sharing	Tweets sharing articles, blogs, tutorials, or videos related to software development.
Technical Discussions	Tweets discussing some technical issues related to software development.
New Releases	Tweets announcing the release of a new software version, tool, etc.
Satires	Tweets sharing jokes and funny quotes generally related to software bugs or issues.
News	Tweets sharing news items related to software development such as joining of a new CEO for a large software company, etc.
Product Promotions	Tweets promoting commercial books and tools related to software development.
Community Events	Tweets about conferences, coding events, anniversaries, etc.
Security Updates	Tweets about latest security issues and fixes affecting software products and frameworks.
Career	Tweets about job openings and candidates sharing their availability for hire.
Crowdsourcing Requests	Tweets requesting users to contribute to open source projects, surveys, petitions, etc. related to software development.
Others	All other events which do not fall into one of the above categories

C. RQ3: How hot is each event category?

In this research question, we analyze the popularity of each event category. For each event category, we count the total number of events in the category. We then plot a heat map showing the hottest event categories (categories with the most tweets) for each programming language and overall in Figure 2.

From the figure, we can note that the top-3 hottest event categories are: *Article and Multimedia Sharing, Technical Discussions*, and *New Releases*. To help developers keep up-to-date with recent trends, we encourage future studies to build automated solutions which are able to find, recommend, and summarize tweets that fall under the ten categories, especially the hotter ones.

Another interesting observation to note is that for the 8 month period, popular *security updates* occurred only for Java and Ruby. Java security updates are retweeted by a large number of Twitter users and one eighth of all popular Java event tweets are about security updates. Another observation is that the number of popular *community events* are higher for Python, Ruby and Scala as compared to C# and Java. Another thing to notice is that for Java, the *Satire* category is more popular than the other languages, on the other hand, no events occur in the *Product Promotions* category.



Category	C#	Java	Python	Ruby	Scala	Combined
Article and Multimedia Sharing	33.5	20	33	25.5	36	29.6
Tech Dicussions	25.5	15	18	14	24.5	19.4
New Releases	16	7	11	18.5	18.5	14.2
Satires	2	26	6.5	14.5	4	10.6
News	5	14	12	4.5	6	8.3
Product Promotions	9	0	4.5	3.5	3	4
Community Events	3	1	5.5	3.5	4	3.4
Security Updates	0	12.5	0	4.5	0	3.4
Other	2	3.5	6	4.5	0.5	3.3
Career	2.5	0.5	3	5	2	2.6
Crowdsourcing Request	1.5	0.5	0.5	2	1.5	1.2

Fig. 2. Hotness of event categories across languages (red = most popular, green = least popular). Numbers represent the ratios of the total number of tweets of an event category to the total number of tweets (in percentages).

D. Threats to Validity

Similar to other exploratory studies, there are some threats that may affect the validity of our study. First, we only study 79,768 Twitter users and their 48,889,030 microblogs which we collect over an 8 month period. Second, the event detection algorithm by Diao and Jiang [4] may wrongly identify events. Third, we only manually analyze 1,000 events using the open coding procedure. Fourth, the open coding procedure involves subjectivity and most of the labeling decisions are made by one person. Still, 48 million (microblogs) and 1,000 (events) are large numbers. Also, Diao and Jiang's algorithm is a state-of-the-art algorithm and has been shown to perform well. Furthermore, the second and third authors have reviewed the first author's labels to improve them.

IV. Related Work

In this section, we first describe closely related empirical studies on how Twitter is used by software engineers (Section IV-A). Next, we describe studies that propose domain-specific analytic solutions that help developers better use Twitter in their software development activities (Section IV-B). Finally, we highlight various studies that also analyze software engineering textual corpora (Section IV-C).

A. Empirical Study of Twitter Use by Software Engineers

Singer et al. in [1] survey 271 developers and interview 27 developers who were active on GitHub to analyse how software developers use Twitter. They find that many developers do use Twitter to keep up-to-date with the latest trends, to consume knowledge, and to network with other developers. They also report two major challenges developers face while using Twitter: information overload due to availability of too much contents, and deciding which users to follow to get interesting and relevant contents. Based on these challenges, there is a need to help developers find latest events or trends from Twitter streams that are useful to software developers. In this work, we perform an exploratory study of events that many developers care about in a 8 month period. We report 11 categories of such events and highlight the three "hottest" event categories. This finding can guide future studies to develop recommendation systems that can identify interesting information from Twitter to developers.

Bougie et al. analyze 11,679 tweets posted by 68 software developers and group the microblogs into four categories, i.e., software engineering related, gadgets, current events, daily chatter [8]. Different from Bougie et al. who target *all* tweets posted by developers, Tian et al. manually categorize 300 software-related microblogs into ten categories [9]. Different from Bougie et al. and Tian et al. we want to find and characterize hot trending events in Twitter. Wang et al. analyze 568 tweets posted by developers from the Drupal open source project [10]. They find that Drupal developers use Twitter to coordinate efforts, share knowledge, encourage potential contributors to join, etc. Recently, Tian et al. investigate the behaviors of software microbloggers in terms of their microblogging frequency, generated contents, and interactions among themselves, based on a dataset that contains more than 13 million microblogs generated by more than 42 thousands software microbloggers [7]. They find that some software microbloggers are very active in posting software-related contents (generate more than 100 software-related microblogs monthly). They also find that the community of software microbloggers is more tightly knitted than that of general microbloggers.

B. Domain-Specific Twitter Analytic Tools for Software Engineers

A number of studies have built tools that can help developers better use or benefit from Twitter in their software development activities [11], [3], [2]. Achananuparp et al. build a visualization tool to help software developers monitor tweets that are related to a particular keyword [11]. Prasetyo et al. build a SVM based tool that can identify whether a microblog is relevant or irrelevant to engineering software systems, based on a set of labeled tweets [3]. Following this study, Sharma et al. propose a novel approach named NIRMAL, which can automatically identify software relevant tweets without any labeled tweets [2]. The core of NIRMAL is a language model learned from a training corpus (i.e., set of documents), which is created from posts on StackOverflow, a programming question and answer site.

C. Text Analysis for Software Engineering

Our work is among the series of works that perform text analysis on software engineering corpora, e.g., [12], [13], [14], [15], [16], [17]. Some of these studies investigate blogs (e.g., [13]), StackOverflow posts (e.g., [12], [14], [15]), bug reports (e.g., [16]), feature descriptions (e.g., [17]), etc. Different from these works, we analyze a different source of textual data.

V. Conclusion and Future Work

Today, Twitter is one of the most popular mediums for information and resource sharing. Software developer also use Twitter a lot in their career related activities especially for remaining updated with the latest happenings, gaining new knowledge, and maintaining a community network [1]. In this work, we perform an exploratory study of events (or trending topics) in software engineering Twitter space which have attracted the interest of many developers. We collect more than 48 million tweets made by close to 80,000 Twitter users over a period of 8 months, filter them based on 5 programming language names, and identify events by running a state-of-the-art Twitter event detection algorithm [4]. We manually analyze 1,000 identified events and group them into categories using the open coding procedure. At the end, we analyze how hot each of these event categories is. Our exploratory study shows that most events that attract the interest of many Twitter users relate to: article and multimedia sharing, technical discussion, new releases, satires, news, product promotions, community events, security updates, career, and crowdsourcing request. The first three categories in particular are the hottest ones.

In the future, we plan to expand this study to include tweets about other programming languages, libraries, software development methodologies, etc. in order to gain a good understanding of features of noteworthy tweets in the software engineering Twitter space. Our eventual goal is to build a recommendation system that can identify tweets that interest many developers (e.g., tweets that fall into categories identified in this study) to help developers keep up-to-date with recent trends and learn new knowledge from Twitter stream. Such a solution will help solve challenges that prevent developers from using Twitter, e.g., information overload, etc. [1], potentially resulting in an increased adoption of Twitter to improve software development activities.

Acknowledgement. We would like to thank Qiming Diao for sharing with us the source code of the Twitter event detection technique used in this work [4]. This research is supported by the Singapore National Research Foundation under its International Research Centre @ Singapore Funding Initiative and administered by the IDM Programme Office, Media Development Authority (MDA).

References

[1] L. Singer, F. M. F. Filho, and M. D. Storey, “Software engineering at the speed of light: how developers stay current using twitter,” in *36th International Conference on Software Engineering, ICSE ’14, Hyderabad, India - May 31 - June 07, 2014*, 2014, pp. 211–221.

[2] A. Sharma, Y. Tian, and D. Lo, “Nirmal: Automatic identification of software relevant tweets leveraging language model,” in *Software Analysis, Evolution and Reengineering (SANER), 2015 IEEE 22nd International Conference on*. IEEE, 2015, pp. 449–458.

[3] P. K. Prasetyo, D. Lo, P. Achananuparp, Y. Tian, and E.-P. Lim, “Automatic classification of software related microblogs,” in *ICSM*, 2012, pp. 596–599.

[4] Q. Diao and J. Jiang, “A unified model for topics, events and users on twitter,” in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013*, 2013, pp. 1869–1879.

[5] A. Strauss and J. Corbin, *Basics of Qualitative Research: Grounded Theory Procedures and Techniques*. SAGE Publications, 1990.

[6] J. Saldana, *The Coding Manual for Qualitative Researchers*. SAGE Publications, 2013.

[7] Y. Tian and D. Lo, “An exploratory study on software microblogger behaviors,” in *Mining Unstructured Data (MUD), 2014 IEEE 4th Workshop on*. IEEE, 2014, pp. 1–5.

[8] G. Bougie, J. Starke, M.-A. Storey, and D. M. German, “Towards understanding twitter use in software engineering: preliminary findings, ongoing challenges and future questions,” in *Proceedings of the 2nd international workshop on Web 2.0 for software engineering*, 2011.

[9] Y. Tian, P. Achananuparp, I. N. Lubis, D. Lo, and E.-P. Lim, “What does software engineering community microblog about?” in *MSR*, 2012.

[10] X. Wang, I. Kuzmickaja, K.-J. Stol, P. Abrahamsson, and B. Fitzgerald, “Microblogging in open source software development: The case of drupal and twitter,” *Software, IEEE*, 2013.

[11] P. Achananuparp, I. N. Lubis, Y. Tian, D. Lo, and E.-P. Lim, “Observatory of trends in software related microblogs,” in *ASE*, 2012.

[12] A. Barua, S. W. Thomas, and A. E. Hassan, “What are developers talking about? an analysis of topics and trends in stack overflow,” *Empirical Software Engineering*, vol. 19, no. 3, pp. 619–654, 2014.

[13] D. Pagano and W. Maalej, “How do open source communities blog?” *Empirical Software Engineering*, vol. 18, no. 6, pp. 1090–1124, 2013.

[14] X. Xia, D. Lo, X. Wang, and B. Zhou, “Tag recommendation in software information sites,” in *Proceedings of the 10th Working Conference on Mining Software Repositories, MSR ’13, San Francisco, CA, USA, May 18-19, 2013*, 2013, pp. 287–296.

[15] S. Wang, D. Lo, B. Vasilescu, and A. Serebrenik, “Entagrec: An enhanced tag recommendation system for software information sites,” in *30th IEEE International Conference on Software Maintenance and Evolution, Victoria, BC, Canada, September 29 - October 3, 2014*, 2014.

[16] Y. Tian, C. Sun, and D. Lo, “Improved duplicate bug report identification,” in *16th European Conference on Software Maintenance and Reengineering, CSMR 2012, Szeged, Hungary, March 27-30, 2012*, 2012, pp. 385–390.

[17] S. Wang, D. Lo, Z. Xing, and L. Jiang, “Concern localization using information retrieval: An empirical study on linux kernel,” in *18th Working Conference on Reverse Engineering, WCRE 2011, Limerick, Ireland, October 17-20, 2011*, 2011, pp. 92–96.