RAMS

Reliability, Availability,
Maintainability, and Safety

This is the Title of my Thesis

Your Name

August 2014

PROJECT / MASTER THESIS

Department of Production and Quality Engineering

Norwegian University of Science and Technology

Supervisor 1: Professor Ask Burlefot

Supervisor 2: Professor Fingal Olsson

Preface

Here, you give a brief introduction to your work. What it is (e.g., a Master's thesis in RAMS at NTNU as part of the study program xxx and...), when it was carried out (e.g., during the autumn semester of 2021). If the project has been carried out for a company, you should mention this and also describe the cooperation with the company. You may also describe how the idea to the project was brought up.

You should also specify the assumed background of the readers of this report (who are you writing for).

Trondheim, 2012-12-16

(Your signature)

Ola Nordmann

Acknowledgment

I would like to thank the following persons for their great help during ...

If the project has been carried out in cooperation with an external partner (e.g., a company), you should acknowledge the contribution and give thanks to the involved persons.

You should also acknowledge the contributions made by your supervisor(s).

O.N.

(Your initials)

Summary and Conclusions

Here you give a summary of your your work and your results. This is like a management summary and should be written in a clear and easy language, without many difficult terms and without abbreviations. Everything you present here must be treated in more detail in the main report. You should not give any references to the report in the summary – just explain what you have done and what you have found out. The Summary and Conclusions should be no more than two pages.

You may assume that you have got three minutes to present to the Rector of NTNU what you have done and what you have found out as part of your thesis. (He is an intelligent person, but does not know much about your field of expertise.)

Contents

Chapter 1

Approaches for Detecting Robots in Social Media

Management Summary

This is the management summary blablabla mhmhm...

1.1 Introduction

Software robots are often called bot. <- muss hier irgendwie rein

1.2 Definition and History of Social Bots

This section will introduce the term social bot formally and give a short overview about the beginning and the development of this topic.

In order to be able to discuss social media bot detection, we need a clear understanding of what social bots actually are. For that, we use the definition given by Ferrara et al. in their article The Rise of Social Bots:

> "A social bot is a computer algorithm that automatically produces content and interacts with humans on social media, trying to emulate and possibly alter their behavior." [6]

The root of of social bots, or just bots how we will sometimes call them here as well, can probably be found in the Turing test, developed by Alan Turing in 1950 [10]. It involves three parties, two of which are human and one is a computer program. While one human is having a conversation with the software, it is the task of the other human to identify the program. If he is not able to do so, the software is passing the Turing test. This led to the development of a lot of so called chatbots, which just aimed to appear as human as possible in a conversation.

A rather famous and often cited example for such a chatbot is ELIZA, introduced by Joseph Weizenbaum 1966 in [13]. It mimicked a psychotherapist and showed that – at least some kind of – communication between a human and a computer is possible.

Since then, a lot of things have changed. Today, bots are a lot more than bare entertainment or proof of concept. With the triumph of the Internet and especially social networks like Facebook and Twitter, the possible use cases for social bots have increased dramatically. While they were initially mostly used to simply post content, today they are able to credibly interact with each other and even humans [3, 7]. As we will see in the next section, nowadays bots are used to spread messages, for marketing and a lot more.

1.3 Why is Bot Detection in Social Media Important?

- Information flood -> need to get the message through
- influence political mood (smoke screening in Dissecting a Social Botnet: Growth, Content and Influence in Twitter)
- marketing
- false information (boston marathon: cassa 2013)
- seem fame
- stock exch..
- https://sysomos.com/inside-twitter/most-active-twitter-user-data 32% of tweets by bots!

—> section "engineered social tampering" and following in the rise of social bots!!

+ Key Challenges in Defending Against Malicious Socialbots [2]

1.4 Social Bot Detection Approaches

In this chapter we want to introduce several techniques for detecting social bots. Based on Ferrara et al. [6] we distinguish three detection approach classes.

The first category of detection approaches is based on social network information. They are also called graph-based, since they map users and their relations into a graph and then try to identify bots in the hereby obtained social network by means of graph theory.

Afterwards we will discuss crowd-sourcing based social bot detection approaches. They use actual humans to detect bots, assuming that the human ability to notice details in communication will make this an easy task.

The last category we want to elaborate on are detection approaches based on machine learning. Mechanisms that make use of this approach try to observe behavioral patterns that are typical for social bots. Since these patterns are encoded in so called patterns, this approach is also known as feature-based [6].

In the following sub sections, we will go into detail about each of these three approaches and illustrate them using real detection systems.

1.4.1 Based on Social Network Information

A term that is often used in combination with detection of bots by using social networks is sybil or the sybil attack. It was presented as a thread to distributed systems by John R. Douceur in [5]. In the specific context of social media platforms when conducting a sybil attack, an attacker creates a large amount of fake identities in a system to the point where these identities make up a considerable fraction of the system's whole user base. When this is achieved, the attacker can influence the whole system and control its contents to a certain degree. A sybil, sybil node or sybil account is therefor simply one of the fake entities, or, depending on the attack architecture, just a social bot. It is not hard to see that social bot detection can, more specifically, be viewed as a defense against the sybil attack.

The general proceeding of social network based bot detection approaches is rather simple. They map the users base of the social platform they aim to defend into a social graph, where a node is corresponding to a user and an edge between two nodes exists if there is a specific kind

of relationships between the two respective users on the platform. The nodes can be hereby be distinguished in sybil nodes, respectively bots, and non-sybil nodes, respectively legitimate users. The goal of the detection approach is now, to identify whether a given node is a sybil or not [11].

There are a number of proposed social network based sybil detection schemes like for example SybilGuard [14], SybilInfer [4] or SumUp [9]. While they all have different assumptions and use varying algorithms to achiever their goal, Viswanath et al. show in [11], that, at a high level, all of them work by them same principles.

Basically they can be viewed as graph partitioning algorithms, which partition a given graph into multiple disjoint subgraphs. As already mentioned above, ideally two subgraphs are assembled, one that contains only sybil nodes and one that contains only non-sybil nodes. Since a clear distinction is often hard to make, the approaches basically assign a rank to each node and decide afterwards, depending on several parameters, which ranks are classified as sybil and which as non-sybil. It is thereby obvious, that the ranking algorithm is crucial for the whole scheme. Though, of course, the ranking algorithms for the different detection schemes are differing, they generally have in common, that they base their rating on how tightly connected the respective node is to a known trusted node. Thus, they work by detecting local communities of nodes. In other words closely connected groups of nodes [11].

It is not hard to see, that these algorithms are therefor easy to deceive. If an attacker is able to establish so called attack edges, connections between his sybil nodes and non-sybil nodes which are connected to a trusted community, it gets significantly harder to identify the bots. A common assumption is that these attack edges are hard to create [14], which means, that legitimate users tend not to establish social network connections to social bots. However, Boshmaf et al. show in [1] that this assumption is to be questioned. They tried to infiltrate the social media platform Facebook with a large number of sybil accounts and tried to establish connections to real users. The average acceptance rate of their relationship requests came to about 20% and could be increased to 80% depending on how many indirect relationships between the sybil and the user already existed [1].

A well known example for a bot detection approach that is based on social network information is the Facebook Immune System (IMS) [8]. This system aims to defend the social media

platform Facebook and its users not only against sybil attacks but also to prevent spam, malware distribution, phishing and so on. To achieve this goal, it takes a lot more actions than the above described general approach for social network based detection schemes. The IMS runs checks on every action performed on Facebook in realtime in order to give attackers as less time as possible to accomplish there goals and to react. It classifies these actions according to predefined policies and makes it thereby possible to judge them and the corresponding users.

An example for this could be a newly created Facebook account, that sends a lot of friendship requests. These are used in Facebook to establish relationships between users. A legitimate user starts sending friendship requests usually mainly to people he knows and vice versa, that is, people that are likely to accept his friendship request. If a lot of these requests are now declined, this could be a indication for the system that the sending user might not be legitimate. The IMS also makes heavy use of machine learning, which we will discuss later on, and generates training data automatically in order to adapt to the fast changing attacks and user behavior [8].

1.4.2 Based on Crowd-Sourcing

A rather straight forward approach for social bot detection is based on crowd-sourcing. In contrast to the above described social network information based approaches, a connection between bots and legitimate users is not a problem for schemes based on this approach – at least not a direct one. The basic idea of crowd-sourcing based social bot detection is, to engage actual human workers to study user profiles and subsequently to decide whether it belongs to an actual user or a sybil.

In [12], Wang et al. presented a study on the effectiveness of this approach and introduce a sample for a crowd-sourcing based social bot detection system. For their tests they use sample data from Renren, China's most popular social media platform, Facebook US and Facebook India. They subdivide their human investigators, or simply testers, in expert workers, and crowd-sourced workers. Each of the testers is presented a number of social media profiles and has to decide, whether it is a real one, or a sybil. While the experts achieve a detection rate at about 90%, the crowd-sourced workers perform not as good individually. If their single votes are aggregated and used for a majority decision though, the results can significantly increased. If this is done for the expert workers, their results can be increased even more, too. A very desirable

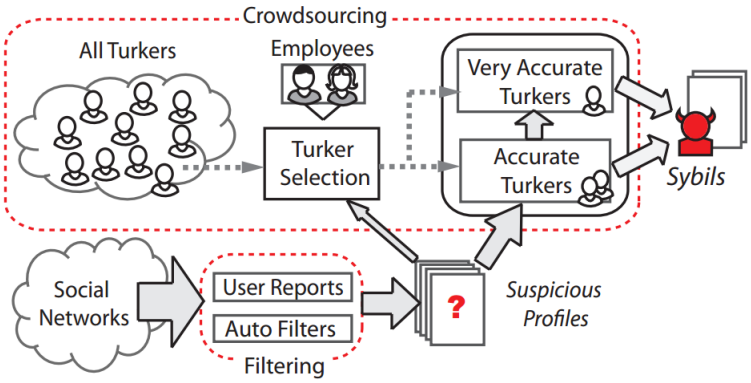


Figure 1.1: A crowd-sourced sybil detection system introduced by Wang et al. [12].

result of their studies is also that the false positive rate, that is the amount of profiles that are falsely classified as sybils, is in all groups very close to zero. This ensures, that the probability that legitimate users are accused to be social bots, what will probably offend them, is very low.

Wang et al. conclude, that it is very hard for sybil creators, to assemble social bots respectively profiles that are able to pass a "social turing test" and that crowd-sourcing based approaches can perform very well. They proceed with introducing a general practical system that is illustrated in figure 1.1 [1].

The system is working like this: The social network respectively social media platform that is to be defended generates suspicious profiles. This can either happen through explicit reports from users or through automatically applied filters that detect abnormal behavior. The so obtained profiles are then checked by crowd-sourced workers, which are subdivided in accurate and very accurate. To establish this differentiation, some of the suspicious profiles are mixed with some profiles that are known to be sybils. They are presented to all workers and their results allow a classification and to filter out unreliable workers. In the actual checking process, the profiles are first presented to the accurate workers which make a majority vote. If their decision is not clear, the profiles are presented to the very accurate workers which once again make a majority vote. The authors claim, that a system like this is very reliable and cost effective.

Though such a social bot detection approach based on crowd-sourcing seems very promising at first sight, several problems become obvious on closer examination. First of all, this ap-

[1] Turkers are in this case the former mentioned crowd-sourced workers.

proach works optimal if every newly created profile can be reviewed in the above described way. However, young social media platforms will probably not be able or willing to pay for this bot detection scheme. A factor that may influence this could be also, that those platforms probably do not have huge issues with the social bot problem in their early stages. Once the platform has grown an the issue arises though, their is usually already a huge user base that has to be reviewed retrospectively. This is not optimal and may take a considerable amount of time.

Another point that mus be taken into consideration is that this approach might not be suitable for all types of social media platforms. While it might be well suited for platforms like Facebook or Renren, where profile pages can be customized a lot and contain usually plenty of information, other networks, like for example Twitter could be less appropriate. Generally it can be said that this detection approach is highly based on the information given in the users' profiles. If the information contained in the profiles is low, human investigators will probably not be able to distinguish as precisely as in the tests of Wang et al.

The last and probably most difficult issue that has to be mentioned is connected to this. Human investigators that are charged with distinguishing between bots and legitimate users have to have access to the profiles of the users. While those profiles are sometimes publicly available anyways, detailed profiles, which are more interesting for this approach, are, as discussed above, usually only visible for certain users. It is not hard to see the privacy issue that arises here, especially when keeping in mind, that the investigators are – at least to a great extent – only crowd-sourced workers, that can not be supervised as easily as ordinary employees [6].

1.4.3 Based on Machine Learning Methods

synchrotrap, copycatch -> clustering

wie misuse based ids!

1.5 Summary and Outlook

Chapter 2

Equations, Figures, and Tables

The content of Chapter 2 will vary with the topic of your thesis. This chapter only gives guidance to some technical aspects of LaTeX.

Remark: If you want a shorter chapter or section title to appear in the Table of Contents and in the headings of the chapter, you just include the short title in square brackets before the title of the chapter/section. Example:

```
\section[Short Title]{Long Title}
```

.

2.1 Simple Equations

Mathematical symbols and equations can written in the text as λ, $F(t)$, or even $F(t) = \int_0^t \exp(-\lambda x)\,dx$, or as displayed equations

$$F(t) = \int_0^t \exp(-\lambda x)\,dx \tag{2.1}$$

The displayed equations are automatically given equation numbers – here (2.1) since this is the first equation in Chapter 2. Note that you can refer to the equation by referring to the "label" you specified as part of the equation environment.

You can also include equations without numbers:

$$F(t) = \sum_{i=1}^{n} \binom{n}{i} \sin(i \cdot t)$$

More Advanced Formulas

Long formulas that cannot fit into a single line can be written by using the environment `align` as

$$F(t) = \sum_{i=1}^{n} \sin(t^{n-1}) - \sum_{i=1}^{n} \binom{n}{i} \sin(i \cdot t) \tag{2.2}$$

$$+ \int_0^{\infty} n^{-x} e^{-\lambda x^t} \, dt \tag{2.3}$$

In some cases, you need to write ordinary letters inside the equations. You should then use the commands

```
\textrm  and/or \mathrm
```

The first command returns the normal text font and will be scaled automatically, while the second command will be scaled according to the use.

$$\mathrm{MTTF} = \int_0^{\infty} R_{\mathrm{avg}}(t) \, dt$$

Please consult the LaTeX documentation for further details about mathematics in LaTeX.

Definitions

If you want to include a definition of a term/concept in the text, I have made the following macro (see in `ramsstyle.sty`):

☞ **Reliability**: The ability of an item to perform a required function under stated environmental and operational conditions and for a stated period of time.

Figure 2.1: This is the logo of NTNU (rotated 15 degrees).

When text is following directly after the definition, it may sometimes be necessary to end the definition text by the command

`\newline`

I have not included this in the definition of the `defin` environment to avoid too much space when there is not a text-block following the definition.

2.2 Including Figures

If you use pdfLaTeX (as recommended), all the figures must be in pdf, png, or jpg format. We recommend you to use the pdf format. Please place the figure files in the directory **fig**. Figures are included by the command shown for Figure 2.1. Please notice the "path" to the figure file written by a *forward* slash (/). You should not include the format of the figure file (pdg, png, or jpg) – just write the "name" of the figure.

Each figure should include a unique *label* as shown in the command for Figure 2.1. You can then refer to the figure by the *ref* command. Notice that you can scale the size of the figure by the option `scale=k`. You may also define a specific width or height of the figure by replacing the `scale` options by `width=k` or `height=k`. The factor `k` can here be specified in mm, cm, pc, and many other length measures. You may also give `k` as a fraction of the width of the text or of the height of the text, for example, `width=0.45\textwidth`. If you later change the margins of the text, the figure width will change accordingly. As illustrated in Figure 2.1, you may also rotate the figure – and also do many other things (please check the documentation of the package `graphicx` – it is available on your computer, or you may find it on the Internet).

In LaTeX all figures are floating objects and will normally be placed at the top of a page. This is the standard option in all scientific reports. If you insist on placing the figure exactly where you

Table 2.1: The degree of newness of technology.

Experience with the operating condition	Level of technology maturity		
	Proven	Limited field history or not used by company/user	New or unproven
Previous experience	1	2	3
No experience by company/user	2	3	4
No industry experience	3	4	4

declare the figure, you may include the command `[h]` (here) immediately after `\begin{figure}`. If you will force the figure to be located either at the top or bottom of the page, you may alternatively use `[t]` or `[b]`. For more options, check the documentation.

Large figures may be included as a *sidewaysfigure* as shown in Figure 2.2:[1]

2.3 Including Tables

LaTeX has a lot of different options to include tables. Only one of them is illustrated here.

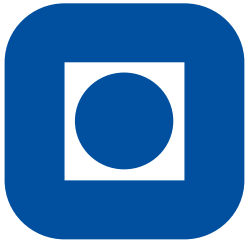
Remark: Notice that figure captions (Figure text) shall be located *below* the figure – and that the caption of tables shall be *above* the table. This is done by placing the `\caption` command beneath the command `\includegraphics` for figures, and above the command `\begin{tabular*}` for tables.

2.4 Copying Figures and Tables

In some cases, it may be relevant to include figures and tables from from other publications in your report. This can be a direct copy or that you retype the table or redraw the figure. In both cases, you should include a reference to the source in the figure or table caption. The caption might then be written as: *Figure/Table xx: The caption text is coming here [**?**].*

In other cases, you get the idea from a figure or table in a publication, but modify the figure/table to fit your purpose. If the change is significant, your caption should have the following format: *Figure/Table xx: The caption text is coming here [adapted from **?**].*

[1] You can use a similar command for large tables.



NTNU – Trondheim
Norwegian University of
Science and Technology

Figure 2.2: This is the logo of NTNU.

2.5 References to Figures and Tables

Remember that all figures and tables shall be referred to and explained/discussed in the text. If a figure/table is not referred to in the text, it shall be deleted from the report.

2.6 A Word About Font-encoding

When you press a button (or a combination of buttons) on your keyboard, this is represented in your computer according to the *font-encoding* that has been set up. A wide range of font-encodings are available and it may be difficult to choose the "best" one. In the template, I have set up a font-encoding called UTF-8 which is a modern and very comprehensive encoding and is expected to be the standard encoding in the future. Before you start using this template, you should open the Preferences ->Editor dialogue in TeXworks (or TeXShop if you use a Mac) and check that encoding UTF-8 has been specified.

If you use only numbers and letters used in standard English text, it is not very important which encoding you are using, but if you write the Norwegian letters æ, ø, å and accented letters, such as é and ä, you may run into problems if you use different encodings. Please be careful if you cut and paste text from other word-processors or editors into your LaTeX file!

Warning

If you (accidentally) open your file in another editor and this editor is set up with another font-encoding, your non-standard letters will likely come out wrong. If you do this, and detect the error, be sure *not* to save your file in this editor!!

This is not a specific LaTeX problem. You will run into the same problem with all editors and word-processors – and it is of special importance if you use computers with different platforms (Windows, OSX, Linux).

2.7 Plagiarism

Plagiarism is defined as "use, without giving reasonable and appropriate credit to or acknowledging the author or source, of another person's original work, whether such work is made up of

code, formulas, ideas, language, research, strategies, writing or other form", and is a very serious issue in all academic work. You should adhere to the following rules:

- Give proper references to all the sources you are using as a basis for your work. The references should be give to the original work and not to newer sources that mention the original sources.
- You may copy paragraphs up to 50 words when you include a proper reference. In doing so, you should place the copied text in inverted commas (i.e., "Copied text follows ..."). Another option is to write the copied text as a quotation, for example:

  > Birnbaum's measure of reliability importance of component i at time t is equal to the probability that the system is in such a state at time t that component i is critical for the system.
  >
  > **?**]

Chapter 3

Summary and Recommendations for Further Work

In this final chapter you should sum up what you have done and which results you have got. You should also discuss your findings, and give recommendations for further work.

3.1 Summary and Conclusions

Here, you present a brief summary of your work and list the main results you have got. You should give comments to each of the objectives in Chapter 1 and state whether or not you have met the objective. If you have not met the objective, you should explain why (e.g., data not available, too difficult).

This section is similar to the Summary and Conclusions in the beginning of your report, but more detailed—referring to the the various sections in the report.

3.2 Discussion

Here, you may discuss your findings, their strengths and limitations.

3.3 Recommendations for Further Work

You should give recommendations to possible extensions to your work. The recommendations should be as specific as possible, preferably with an objective and an indication of a possible approach.

The recommendations may be classified as:

- Short-term
- Medium-term
- Long-term

Appendix A

Acronyms

FTA Fault tree analysis

MTTF Mean time to failure

RAMS Reliability, availability, maintainability, and safety

Appendix B

Additional Information

This is an example of an Appendix. You can write an Appendix in the same way as a chapter, with sections, subsections, and so on.

B.1 Introduction

B.1.1 More Details

Bibliography

[1] Y. Boshmaf, I. Muslukhov, K. Beznosov, and M. Ripeanu. The socialbot network: when bots socialize for fame and money. In *Proceedings of the 27th Annual Computer Security Applications Conference*, pages 93–102. ACM, 2011.

[2] Y. Boshmaf, I. Muslukhov, K. Beznosov, and M. Ripeanu. Key challenges in defending against malicious socialbots. In *Presented as part of the 5th USENIX Workshop on Large-Scale Exploits and Emergent Threats*, 2012.

[3] Y. Boshmaf, I. Muslukhov, K. Beznosov, and M. Ripeanu. Design and analysis of a social botnet. *Computer Networks*, 57(2):556–578, 2013.

[4] G. Danezis and P. Mittal. Sybilinfer: Detecting sybil nodes using social networks. In *NDSS*. San Diego, CA, 2009.

[5] J. R. Douceur. The sybil attack. In *International Workshop on Peer-to-Peer Systems*, pages 251–260. Springer, 2002.

[6] E. Ferrara, O. Varol, C. Davis, F. Menczer, and A. Flammini. The rise of social bots. *arXiv preprint arXiv:1407.5225v3*, 2015.

[7] T. Hwang, I. Pearce, and M. Nanis. Socialbots: Voices from the fronts. *interactions*, 19(2):38–45, 2012.

[8] T. Stein, E. Chen, and K. Mangla. Facebook immune system. In *Proceedings of the 4th Workshop on Social Network Systems*, page 8. ACM, 2011.

[9] D. N. Tran, B. Min, J. Li, and L. Subramanian. Sybil-resilient online content voting. In *NSDI*, volume 9, pages 15–28, 2009.

[10] A. M. Turing. Computing machinery and intelligence. *Mind*, 59(236):433–460, 1950.

[11] B. Viswanath, A. Post, K. P. Gummadi, and A. Mislove. An analysis of social network-based sybil defenses. *ACM SIGCOMM Computer Communication Review*, 40(4):363–374, 2010.

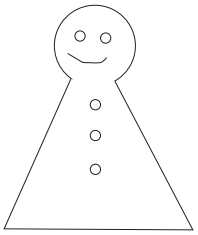
[12] G. Wang, M. Mohanlal, C. Wilson, X. Wang, M. Metzger, H. Zheng, and B. Y. Zhao. Social turing tests: Crowdsourcing sybil detection. *arXiv preprint arXiv:1205.3856*, 2012.

[13] J. Weizenbaum. Eliza—a computer program for the study of natural language communication between man and machine. *Communications of the ACM*, 9(1):36–45, 1966.

[14] H. Yu, M. Kaminsky, P. B. Gibbons, and A. Flaxman. Sybilguard: defending against sybil attacks via social networks. In *ACM SIGCOMM Computer Communication Review*, volume 36, pages 267–278. ACM, 2006.

Curriculum Vitae

Name:	**Your Name**
Gender:	Female
Date of birth:	1. January 1995
Address:	Nordre gate 1, N–7005 Trondheim
Home address:	King's road 1, 4590 Vladivostok, Senegal
Nationality:	English
Email (1):	your.name@stud.ntnu.no
Email (2):	yourname@gmail.com
Telephone:	+47 12345678



Your picture

Language Skills

Describe which languages you speak and/or write. Specify your skills in each language.

Education

- School 1
- School 2
- School 3

Computer Skills

- Program 1

- Program 2
- Program 3

Experience

- Job 1
- Job 2
- Job 3

Hobbies and Other Activities