

Online-Learning in Social Networks

— Report One —

Shanshan Fu, Oh Boon Eng, Ani Petrosyan, Magdalena Sadowska, Shahrokh Shahi, Yunxin Zhou
{sf2715, beo12, ap7614, ms915, ss6515, yz5115}@ic.ac.uk

Supervisor: Dr. Alessandra Russo
Course: CO530/533, Imperial College London

8th May, 2016

1 Introduction

Today, with around 3 billion people using some form of the Online Social Network (OSN)[1], the difficult challenge of preserving one's privacy whilst still fulfilling their social needs is almost inevitable. In order to decrease such difficulty, OSN platforms often allow the uploaders to define a target audience of the information they post. However, some of the members of the preselected group may choose to re-share the information, which could lead to unexpected privacy breaches[2]. As a result, the uploaders might not always anticipate the negative consequences of their online activities and engage in actions that they may later regret[3, 4].

The aim of this project is to develop a web plugin based on the research conducted by Imperial College London. The study proposes a Learning to Share product, in which a parametric Markov chain (PMC), a chain with unknown transition probabilities, is utilised to evaluate the privacy risk and social benefit of sharing an information with each friend or a member of a predefined target audience.

This report describes initial steps and methodology taken to develop a product based on the research and consists of the following sections. Section 2 provides the description of the project. Section 3 lists the core and additional requirements set by the research group. Section 4 discusses the development approach, techniques, system boundaries and proposed timeline of the project.

Being currently the most widely used OSN that provides a variety of user interactions[5, 6], Facebook will be considered as the prominent case study in this project.

2 Design

The proposed Learn to Share approach consists of two fundamental aspects, being social benefit and privacy risk of sharing information. Social benefits is represented by positive response gained through sharing of the information, whereas the privacy risk is defined as the risk of being exposed to sharing sensitive information with an improper audience.

In order to evaluate the said privacy risk and social benefit, the research group proposes a web plugin that dynamically monitors interactions between the user and group members, categorising them into one of the predefined groups based on their potential privacy violation and displays an appropriate warning message, if necessary.

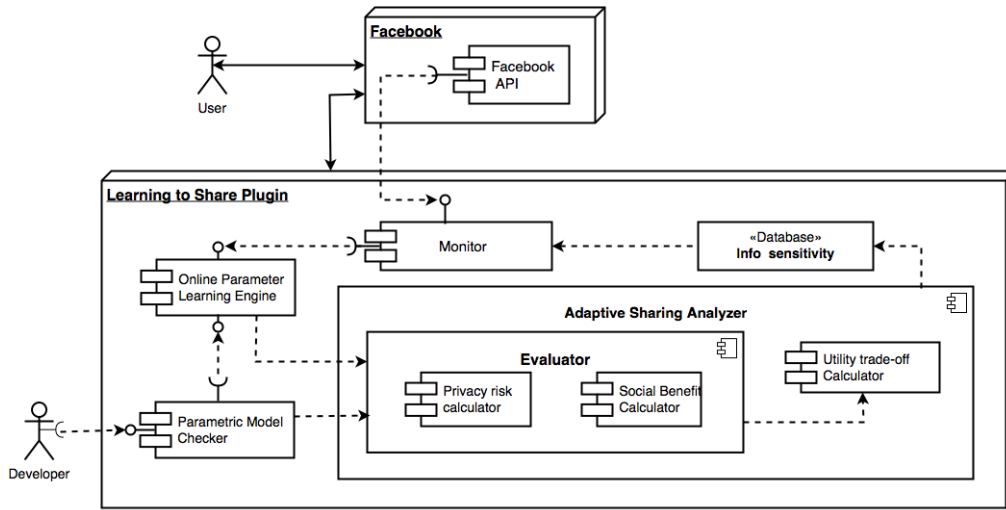


Figure 1: Component Diagram

Consequently, the most essential components that will make up the web plugin are as follows (Figure 1). The first component is Facebook API, in which the user logs in into Facebook and inputs the information along with the target audience. The API also allows the user to specify the social benefit and privacy risk thresholds, as well as the level of sensitivity. Besides, each piece of information, together with its corresponding level of sensitivity, is stored in the database. The Parametric Model Checker then takes two essential parameters as inputs: a parametric Markov chain model and the user's sharing preference for the social benefit requirement, which can be obtained through probabilistic calculation tree logic (PCTL) formula. The latter one is an algebraic expression produced using an open-source probabilistic model checker, PRISM. The Monitor keeps track of the interactions on the user's posts, which will be used in the learning model. The Online Parameter Learning Engine updates the model parameters. The Adaptive Sharing Analyzer then uses the updated parameters that are used to calculate the privacy risk and social benefit. The calculated values are then compared with the user- specified thresholds to estimate the trade-off, which is then used to determine if the information is safe to be shared with the selected audience.

Furthermore, Figure 2 depicts more detailed sequence of activities of the web plugin.

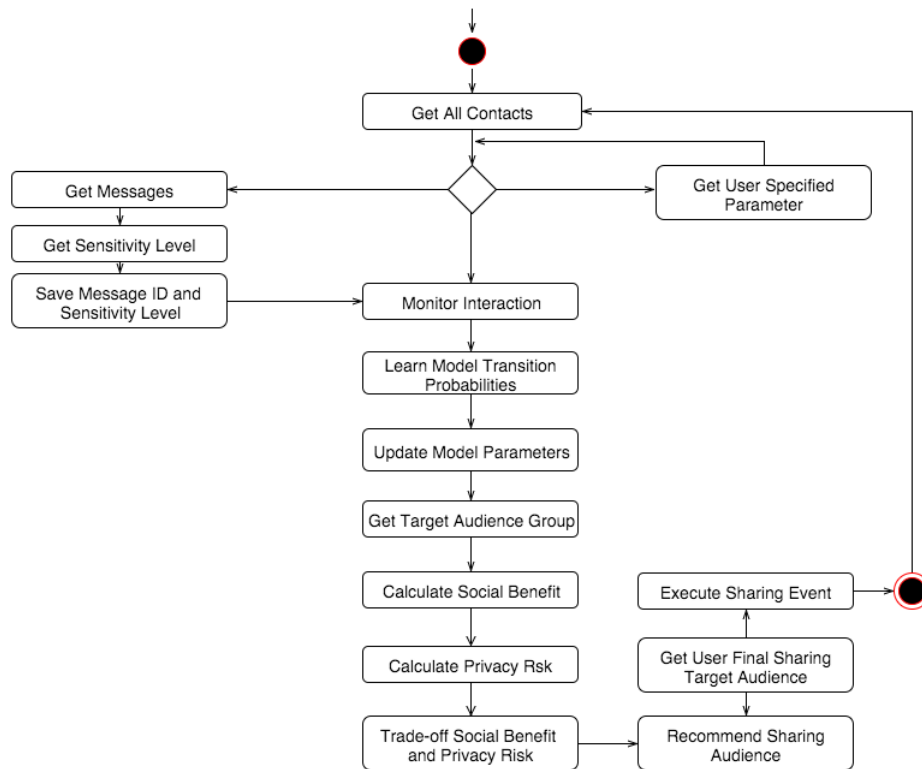


Figure 2: Activity Diagram

3 Specifications

3.1 Requirements

Main high level essential requirements of the web plugin proposed by the research group include the creation of a component, that is responsible for monitoring the social interaction between the user and each friend, as well as adding an automatic manager responsible for recommendation to the user to select a proper audience group. Table 1 demonstrates the low level specifications derived from the aforementioned requirements by applying the simplified version of van Lamsweerde's KAOS goal-oriented model [7], questioning how and why a particular requirement is applied. Table 1 is grouped by a particular component of the plugin and includes a relevant risk levels.

Table 1: Specifications

	Component	Description of Specifications	Risk*
Minimum Specifications	Facebook API	Creating an interface to get permission from user through face-book API	L
		Creating a user interface that allows the user to input the level of sensitivity as well as the privacy risk and social benefit thresholds	L
	Database	Creating a database in form of a hash table to store information and sensitivity level	L
	Monitor Module	Keeping track of interactions between the user and selected friends	H
	Parametric Model Checker	Integrating of existing parametric Markov chain model with the application	L
	Online Parameter Learning Engine	Integrating of existing code that updates the parameters of the parametric Markov chain model	L
	Adaptive Sharing Analyzer	Creating a privacy risk and social benefit calculator	L
		Verifying if the user specified sharing preferences are met	L
		Calculating utility trade-off between social benefit and privacy risk	H
		Identifying the potential malicious friends and report them to the user	L
Possible Extensions	Database	Development of a database in place of hash table storing relevant data	L
	Online Learning Model	Automate the generation of the online learning model for any online social network based on the activity diagram on the network	H
	Level of Sensitivity	Explore machine learning methods and algorithms to improve determination of level of sensitivity	H
	Parametric Model Checker	Automate producing the algebraic expression through PRISM for other social networks	H
	Plugin Support	Extension of the plugin to support multiple browsers and exploring the possibility of support on mobile devices	H

*Risk index is categorised as high (H) or low(R), depending on its impact and occurrence. The following were considered in the classification: human, i.e. lack of development experience, software, being bugs with chosen tools and frameworks, and schedule, meaning inability to complete given specifications within given time framework.

Additionally, the non-functional specifications that need to be taken into consideration are performance and security. As far as performance is concerned, the application of concurrency will be implemented. The security, on the other hand, will be met by a strict implementation of Facebook API, as well as a thorough testing of the plugin.

Initially, the essential requirements of the web plugin were concerned with monitoring social interactions within a user's post in a group. However, after application of the said KAOS goal model, it became obvious that monitoring of the user's post in a group would not be feasible. More specifically, during the research concerning how to implement such an aspect, it was discovered that selecting the target audience of shared information is not allowed, therefore making it impossible for the web plugin to enable the exclusion of malicious members. That discovery led to a significant change in requirements from the group setting to monitoring of the Facebook Timeline activities, which introduced a new probabilistic model, as described in Section 2.

4 Methodology

4.1 System Boundaries

Main components of the programs backend service will be developed using Java programming language. The reasoning behind such is the lack of system dependency of the language as well as a widely available multithreading library needed for development of the Monitor component[8]. Moreover, the existing code for the Online Learning Model was initially written in Java, therefore incorporation of such will be more efficient and less complex.

The API fetching and connection to the server as such will be developed using either JavaScript, or by implementation of a JavaScript parser due to strict language requirements of the web plugin. The multi-browser support, as well as other frontend and backend development, testing and deployment tools and frameworks are not yet decided upon and will be researched further in the following weeks.

The possible challenge that could arise is the lack of previous experience in Java, as well as a number of solutions that could be adopted to develop a cross-browser plugin. To ensure the refactoring of the code is kept to minimum, thorough research of the languages and available tools will be conducted prior any development takes place.

4.2 Development technique

Due to schedule constraints and lack of previous software development experience, the development process will take place iteratively[9]. Such approach allows for creation of mini-projects, which appears to be less risky than a spiral model in the instance where additional features would not be possible to implement.

Moreover, the project will be adopting the Agile ideology[10] with Scrum framework. The reasoning behind such is more flexible model of Scrum over XP, and a greater suitability for smaller, as in this instance, development teams. Another appealing feature of Scrum framework that will help with the development of plugin is an opportunity of daily meetings, which would allow to discover potential issues early. Also, Scrum will enable a more efficient development of the plugin by more structured process through division of work into tickets. To ensure the group maintains the agile development structure throughout the process, the use of Trello[11] software for the ticketing purposes, and Asana[12] for the project management will be implemented.

As far as the version control system is concerned, the distributed approach as opposed to centralized will be adopted due to the flexibility it offers and the option to work on several features simultaneously[13]. Although it is believed that distributed model could be more difficult to maintain and prone to merge conflicts[13], previous experience with the Git-based software, along with daily meetings should reduce such risks to minimum. Moreover, GitLab was decided upon over Mercurial and other distributed VCSs, due to reliability, popularity and an excellent logging capability[14]. Additionally, the project repository will be hosted at the Imperial College GitLab server, providing with an opportunity of immediate support in case of unexpected issues with the system.

4.3 Proposed Schedule

Due to selected iterative model, the scheduling strategy consists of the division of the project development into one research phase and four goal-oriented phases[?]. Each phase is inclusive of the technical group meeting for the ticketing purposes, as well as the ongoing testing and documentation. Moreover, a follow-up meeting with the research group will take place after the completion of each phase to ensure all the requirements were met as expected.

The proposed schedule is subject to possible changes based on the outcome of the meetings or general alterations of requirements. The development process as such will be split into one week Sprints.

4.4 Division of work

All the features and tasks will be available to select from between all group members, as opposed to assigning a particular area of expertise per developer.

Furthermore, each group member will be selected as a product owner of a different phase to ensure that all the relevant tickets would be created and developed. Also, the phase owner will assign a person responsible for adding to and updating documentation. A User Experience, a Quality Assurance Tester and documenter will be assigned on the rolling bases in each phase.

The reasoning behind our approach is to ensure that every group member gains experience in a variety of roles and areas.

Table 2: Division of Product Ownership

Group Members	Responsibility
Boon Eng Oh, <i>Group Leader</i>	Overall Product Ownership
Shahrokh Shahi	Phase Zero
Magdalena Sadowska	Phase One
Shanshan Fu	Phase Two
Yunxin Zhou	Phase Three
Ani Petrosyan	Phase Four

References

- [1] eMarketer. 2014. *Internet to Hit 3 Billion Users in 2015*. [ONLINE]. Available at <http://www.emarketer.com/Article/Internet-Hit-3-Billion-Users-2015/1011602>[Accessed 23 January 16]
- [2] Xu, H., Teo, H.H., Tan, B.C. and Agarwal, R., 2009. The role of push-pull technology in privacy calculus: the case of location-based services. *Journal of Management Information Systems*, 26(3), pp.135-174.
- [3] Wang, Y., Leon, P.G., Chen, X. and Komanduri, S., 2013. From Facebook regrets to facebook privacy nudges. *Ohio St. LJ*, 74, p.1307.
- [4] Wang, Y., Norcie, G., Komanduri, S., Acquisti, A., Leon, P.G. and Cranor, L.F., 2011, July. I regretted the minute I pressed share: A qualitative study of regrets on Facebook. *In Proceedings of the Seventh Symposium on Usable Privacy and Security* (p. 10). ACM.
- [5] Statista. 2016. *Leading social networks worldwide as of January 2016, ranked by number of active users (in millions)*. [ONLINE]. Available at <http://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users/>[Accessed 26 January 16].
- [6] Yang, M., Yu, Y., Bandara, A.K. and Nuseibeh, B., 2014, September. Adaptive sharing for online social networks: a trade-off between privacy risk and social benefit. *In Trust, Security and Privacy in Computing and Communications (TrustCom), 2014 IEEE 13th International Conference on*(pp. 45-52). IEEE.
- [7] Van Lamsweerde, A. and Letier, E., 2004. From object orientation to goal orientation: A paradigm shift for requirements engineering. *In Radical Innovations of Software and Systems Engineering in the Future* (pp. 325-340). Springer Berlin Heidelberg.
- [8] Oracle. *The Java Language Environment*. [ONLINE] Available at: <http://www.oracle.com/technetwork/java/intro-141743.html#318> [Accessed 27 January 16].
- [9] Chang, S.K., 2002. *Handbook of Software Engineering & Knowledge Engineering: Fundamentals*. (Vol. 1). World Scientific.

- [10] Sutherland, J., Schwaber, K., Scrum, C.C.O. and Sutherl, C.J., 2007. *The scrum papers: Nuts, bolts, and origins of an agile process*.
- [11] Trello.[ONLINE] Available at: <https://trello.com/>. [Accessed 01 February 16].
- [12] Asana.[ONLINE] Available at: <https://app.asana.com/>. [Accessed 28 January 16].
- [13] Hardeep, S., 2013, *Designing, Engineering, and Analyzing Reliable and Efficient Software*.
- [14] Van Baarsen, J. 2014. *GitLab Cookbook*. Packt Publishing Ltd.