Simulating Users in a Social Media Platform Using Multi-Agent Systems

Daniel Pérez and Estefanía Argente

Valencian Research Institute for Artificial Intelligence (VRAIN),
Universitat Politècnica de València, Camino de Vera s/n, 46022 Valencia, Spain
dapregar@vrain.upv.es; esarvil@vrain.upv.es

Abstract. The massive use of social media makes it increasingly easy to find highly sensitive information about almost anyone on the Internet. Despite the efforts of social media platforms to provide their users with tools to manage their privacy, these are proving insufficient due to their complexity. For this reason, it has been considered necessary to develop a software tool based on a multi-agent system to help users to improve and correct their bad behavior by using automation mechanisms and transmitting the information in a natural way for them, replicating the behavior of a human being. The aim of our work is to implement a multi-agent system where agents interact organically with each other and with human users on the PESEDIA social network, so that they can support user in a non-intrusive manner, using paternalistic techniques through actions available on the social network.

Keywords: Social Media Platform · multi-agent · privacy · simulation

1 Introduction

Over the last decade Social Media Platforms (SMPs) have experienced a spectacular increase in terms of their number of users. From Facebook, with 1.26 billion active users daily (as of September 2019) [9], through Twitter [11], with 145 million active users every day in the same period, to the most recent Instagram, with approximately 500 million active users daily (as of September 2017) [10], just to mention some of the most popular ones, it can be inferred that social media are undoubtedly an integral part of our daily lives.

As a result, users have gradually lost their fear of this type of technology, showing themselves more and more willing to pour into them personal information, such as their thoughts or ideologies, chores, opinions, conflicts, etc. This trend is even more notable among the younger strata of society [14], who have grown up with SMPs and therefore do not have the prejudices shown, initially, by people who have been able to experience the popularization of the Internet, where privacy was highly valued, driven by fear of the unknown.

Deciding when to disclose confidential information has always been difficult, especially for those who join and publish content on SMP [17]. The main motivations for disclosing private information at social media are sharing information with friends and acquaintances, using it as a means of storing information,

keeping up with trends, as well as a means of fostering narcissism and showing popularity [19]. The preservation of privacy is essential in human relationships, but we often consent to unreliable software that collects, stores and processes our data, often not knowing how this information will be protected or who will have access to it [15].

In the SMPs, users often reveal a lot of personal information in their profiles, in order to appear more sympathetic and friendly to others [7]. Users should learn the inherent privacy rules of the social network, so that they know how much they need to disclose in order to be socially accepted. Generally, this learning is achieved through continued use of the site [13, 18], e.g. participants with private profiles tend to have friends who also have private profiles on Facebook.

Although there are many campaigns to raise awareness of the risks of social media, practical training should be provided on the proper use of personal data, privacy, as well as greater attention to the content of messages and how these messages can involve content risks for other users [2]. In training actions, it is important to bear in mind that a person learns only 40% of what he observes and hears, but 80% of what he lives or discovers on his own[1]. Thus, only practical training, based on "learning by doing", in which the individual is fully involved, can ensure a high success rate. So it will be very useful to develop secure SMPs that integrate simulation scenarios, allowing users practice so that they become aware, in a safe environment, of the needs for better control of their privacy.

For this reason, the PESEDIA - AI4PRI (Artificial Intelligent agents for Privacy Aware in Social Media) research project[2] was proposed with the didactic objective of teaching and helping correcting all these erratic behaviours to its users. In this project, a multi-agent system (MAS) oriented to social media will be developed to address the problem of privacy, offering each user a personalized agent to help them make decisions regarding the performance of actions that may involve a risk to their privacy. Likewise, this project will provide a controlled environment where virtual agents and human agents co-exist and where various situations that potentially compromise user privacy in the context of social networks can be simulated. The aim is to facilitate the learning of the good use of social networks in a totally practical and immersive way.

Models, techniques and technologies of multi-agent systems are interesting for the study of social networks and the development of models based on them. An example of the use of MAS and social media is found in Franchi and Poggi's work [6], where the use of a distributed MAS is proposed for the creation of a social network where the information of each user is managed by an agent. The proposal focuses on suggesting friendship connections based on the user's profile. In Kökciyan and Yolum's work [12] an agent-based social network is proposed, where the agents manage the privacy requirements of the users and create privacy agreements with the agents. The agent checks the current status of the system to detect and resolve privacy breaches before they occur.

[1] Source: National Training Laboratories, 1977.
[2] https://pesedia.webs.upv.es/

Throughout this project, one of the biggest problems we have faced is trying to solve the way we show the users information about their activity on the SMP. Currently, the way to communicate it to them consists on showing them different panels and informative pop-ups. This way, despite being practical, has been proven to be insufficient through different experiments carried by our research group. There are many studies [3–5, 8] that shown that a human being tends to give more credibility and importance to the information when they perceive it is another human being who is transmitting it to them, even modifying their behaviour when they perceive they are speaking with an artificial intelligence.

In order to improve the way in which we communicate to the users of the PESEDIA SMP the information necessary to fulfil the educational objectives of our project, we have decided to integrate a series of intelligent agents that simulate the behaviour of a human being. In this work we present our first results derived from the design and the implementation effort carried out to integrate a multi-agent system inside PESEDIA.

The aim of our proposal is to implement a multi-agent system where agents interact organically with each other and with human users on the PESEDIA social network, with the future objective of providing them with the necessary tools so that they can support users is a non-intrusive manner, using paternalistic techniques through the tools available on the social network such as comments, "I like it", private messages, etc., which have been proven to be effective [20].

The rest of the document is structured as follows. Section 2 provides a brief description of the PESEDIA social network. Section 3 gives an overview of the components and interactions of our proposed system. Section 4 shows examples of the implementation of some of the numerous actions that agents can carry out within our SMP, together with the results obtained from their execution. Finally, Section 5 details different proposals for future work and the conclusions of the paper.

2 Pesedia

PESEDIA[3] is a social network, based on the open source platform Elgg[4], with an educational purpose, whose main objective is to teach its users the importance of their personal information and the dangers of not correctly managing the scope of the content published on the social network.

PESEDIA's target population is mainly adolescents, as they are the sector most unprotected against the dangers of the Internet. In recent years, several seminars have been held within the Summer School of our university, in which 13 and 14 year-old students have used PESEDIA and, through game techniques, have learned to use the social network, as well as to understand the involvement and impact of each interaction they make. These sessions have also served to collect anonymous information with which to generate models that help understand the relationship that children have with issues related to privacy, their

[3] https://pesedia.webs.upv.es/
[4] https://elgg.org/

emotional state and how all this influences when generating content, how they modify their behaviour patterns as knowledge of the platform increases, etc.

Up to now, PESEDIA included a series of agents responsible for advising and warning children when they were going to publish sensitive content, through text analysis, emotions and argumentation techniques. These agents were intended to help users correct the content they post on social networks, following up individually to achieve an improvement over time.

3 Overview of our Proposal

This section provides an overview of the proposed system, detailing its components, its communication mechanisms and the technology required.

Figure 1 shows the system components together with their communication flow. The workflow of the system is divided into four different parts (numbered in the image). Next, we briefly describe these four parts:

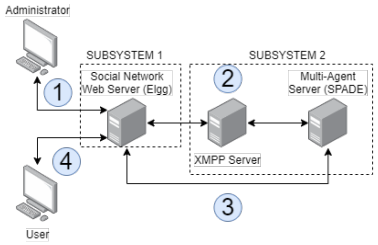

Fig. 1. System Components and Communication flow.

1. The administrator of the SMP can manage the multi-agent system through a web plugin (named here as subsystem 1) that includes both the interfaces and the communication mechanisms needed to make the link between the SMP and the multi-agent system effective. The administrator connects via web through the browser included in his/her device (PC, mobile, tablet, etc.) to the administration panel of the SMP, so as to activate the plugin provided through the corresponding menu.
   Once the plugin is activated, the administrator will see a new option within the administration menu (named "Multi-agent Settings"), and, when accessing this new option, he will be able to connect the SMP with the MAS (see Figure 3). To do this, the web plugin asks the multi-agent system its status to know if it is available and, if so, it shows the administrator a new window where he can see the system status, as well as the different possible operations: add and remove agents, disconnect, etc.
2. The second part of the workflow includes the protocols and communication channels used by the web plugin and the multi-agent system (named here

as subsystem 2). It deals with one of the communication mechanisms used between the SMP and the MAS - a full diagram detailing the protocols used is available in Figure 2). As we have developed our multi-agent system on the SPADE[5] platform, and this platform uses the XMPP (Extensible Messaging and Presence Protocol) protocol[6] as a communication mechanism for the agents, we have thought it convenient to use it as well.

In the proposed MAS we have a coordinating agent, named *master agent*, that is in charge of managing the rest of agents (which will normally represent SMP users) that we introduce in the system. Both the web plugin and each of the agents in the multi-agent system have their own JID (Jabber ID) identifier. Using this identifier they are able to send, receive and interpret the different messages sent between the different actors in the system.

Furthermore, when the administrator performs any operation that may need to receive or send information to the MAS - check its status, for example - the web plugin is in charge of sending the request, using its own JID, to the associated JID of the master agent. This master agent then processes the message received, inform the other agents of the MAS of the relevant information for them and respond to the web plugin accordingly. The web plugin processes the response and updates the necessary information on the screen so that the administrator can know the result of the operation.

3. The third part of the system workflow deals with the other communication mechanism used between the SMP and the MAS. Each time the master agent notifies the other agents of an event of their interest, or when they themselves consider it, they should establish a communication channel that allows them to carry out the necessary operations to meet their objective.

   Elgg offers the programmer the possibility of implementing functions accessible through an API, as a native way of managing communication with external systems. Therefore, we have found it reasonable that the agents use this mechanism to interact with the SMP and to be able to both extract and persist information from it without having to go through the master agent. To achieve this they consult, through the API, the endpoint (a specific web address) designed for this task. Once they get the API response, they can perform the necessary operations to try to reach their goal and, if needed, they can persist results in the SMP using another endpoint. For interacting with the API, agents must indicate the user assigned to them in the SMP.
4. The fourth part of the system's workflow is concerned with solving the way the SMP (specifically the web plugin) sends information to the MAS so that it is aware of changes in the website based on the interaction of users and other agents. Every time the agents persist information in the SMP, it is likely that this information must be published so that the real users of the social network or other agents in the system become aware of it.

   Both when the agents publish the content through the API, and when the users do it from the pertinent options available within the SMP, it is neces-

[5] http://spade.gti-ia.dsic.upv.es/index.php
[6] https://xmpp.org/

sary that such content is visible to them. To solve the *agent* → *user* communication (understanding by "user" a real user or another agent) we have several functions exposed inside the API to allow the information persistence. This information, once in the database, is treated by Elgg and the framework itself facilitates its visibility based on what has been provided.
In order to solve the *user* → *agent* communication (understanding "user" as a real user or another agent) we use Elgg's tools that allow us to capture the different events associated to the actions that occur within the social network. When one of these events is triggered we proceed to send via XMPP (as explained in step 2) the information to the master agent, and this one is in charge of spreading it to the other agents as it considers appropriate.

Figure 2 shows in a conceptual manner the communication protocols used in our proposed system, based on the information above.

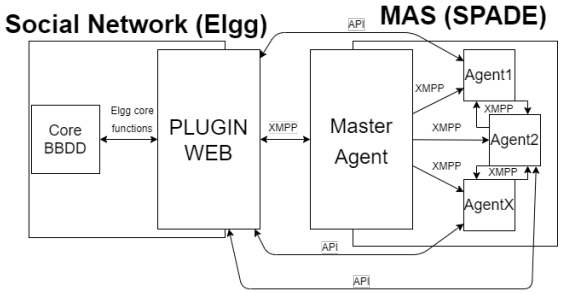

Fig. 2. Communication protocols used between the SMP and the MAS.

A reasonable doubt that may arise to the reader is the need to use two different communication protocols instead of sticking to only one of them. All these reasons are detailed below:

- **Reactivity and proactivity of the agents**: the nature of the agents implies that they can act on their environment and react to external stimuli. If we only used Elgg's API there would be no way to inform the agents of what is happening on the social network unless they decide to look into it. This problem could be solved by using only XMPP, but choosing this solution could generate new problems, which are detailed below.
- **Saturation in communications and modification of the servers' structure**: having determined that in case of opting for a single communication channel, XMPP is the only alternative, the need to know how this affects the different servers involved arises. First of all, it would be necessary for the server where the social network is hosted to be constantly listening for possible XMPP calls. To do this, it would be necessary to modify the server and add a socket in which to host the XMPP client responsible of managing

these communications, which would move it away from its original design and could lead to bad practices and future unwanted errors. On the other hand, the XMPP server in charge of managing the communications would have twice the workload because it would also have to attend the requests coming from the agents, what could cause an overload and require more hardware resources than the desired ones.

- **Web server overload**: the web server should only serve content and we must ensure that our solution does not force an additional workload. As we have seen in the previous point, a socket would allow to host an XMPP client that would attend the agents' requests. However, this socket would not be a traditional web socket and, instead of being open waiting for HTTP calls, it would have associated an XMPP agent that would be waiting in a loop for XMPP requests with a persistent connection, forcing to dedicate constant resources for a task that may not be necessary. This use of resources would obviously penalize the performance of the web server, already subject to a large workload due to the great amount of events that it has to handle with the normal use of the social network.

4 Validation

In order to validate the proposal, two different activities have been implemented.

The first activity replicates the process of registering a new user in the SMP, which requires the creation of new users in PESEDIA and also new agents within the multi-agent system. Currently, at the time of creation, the administrator will be able to choose between creating agents of two different types: extraverted or introverted. Extraverted agents will be willing to make as many friends as possible, whereas introverted agents will be shyer and less active in making friends. As future work, we would like to include other types of agents, such as egocentric, reserved, etc., that represent the different kinds of users that you would normally face within a social network.

To ensure that the subsystems can understand each other when communicating instructions, it has been necessary to develop a lexicon that both can interpret. For this purpose, a series of fixed structures, functions and parameters have been used which, once formatted using JSON notations, allow the desired instructions to be transferred unequivocally between both systems.

To add new agents, the administrator must access the corresponding screen (shown in Figure 3) once the relevant plugin is activated in Elgg. Inside the screen, he must configure the number of users and the percentage distribution of types of agents that he wants to make.

Next, our web plugin creates the users in the social network using an Elgg's extension, named hypeFaker [7] plugin; and it also requests the MAS (specifically to the master agent) to create the corresponding agents, using a JSON message like the one displayed in Figure 4.

[7] https://github.com/hypeJunction/hypeFaker

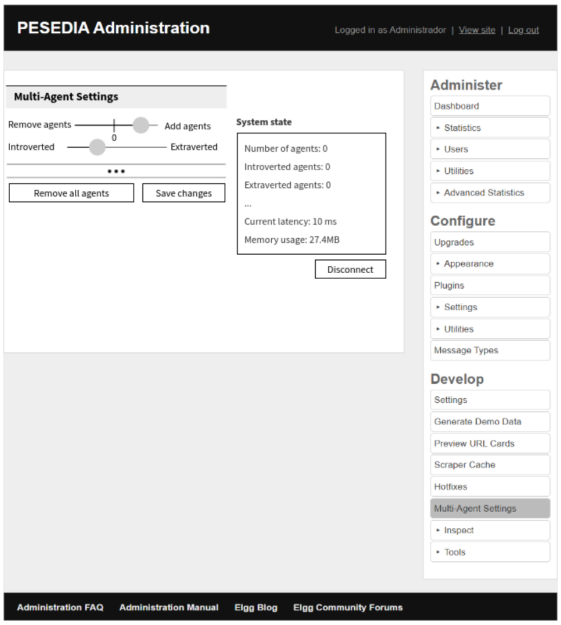

Fig. 3. Multi-Agent Settings Screen.

After receiving the petition, the master agent proceeds creating the requested agents thanks to SPADE's functionality and passing them the required information to enable them to communicate freely with PESEDIA. Once created the agents start their behaviours as expected.

For the purpose of this paper, we have decided to also include the action of adding friends. This action in particular is quite interesting because it requires to perform certain instructions both within the multi-agent system, and the SMP. Additionally, this action can be started by the agents instead of the users, demonstrating how the agents can be proactive if needed. The following formula has been designed so that agents can decide whether they want to add new friends or not:

$$fDesire = min(100, max(iDesire * 25 + tPersonality * 8.33 - (nFriends * opposite(1 - nFriends * 0.3) + rBehaviour), 0))$$

where:

- **fDesire**: the final desire to perform the action. Its value is within [0-100].
- **iDesire**: the initial desire to perform the action, depending of the type of user we are representing, used to represent how active the agent should be in the SPM. Possible values: 1 (low-willing to do this action), 2 (medium-willing to do this action) or 3 (high-willing to do this action).

```
{
  "operation": "mas_operation_add_agents",
  "additional_parameters": {
    "personalities_and_number": [
      {"type": "INTROVERTED", "number": 1},
      {"type": "EXTRAVERTED", "number": 1}
    ],
    "pesedia_ids": [1203, 1204],
    "user_names": ["John Doe", "Jane Doe"]
  }
}
```


Fig. 4. Example of a JSON sent by PESEDIA to the MAS with the information required in order to create 2 new agents, as specified by the administrator.

- **tPersonality**: user personality type. Possible values: 1 (introverted), 2 (neutral), 3 (extraverted).
- **nFriends**: the number of friends the agent already has. Since humans tend to have threshold numbers regarding their number of friends, this factor must be implemented in the agents too. Initially agents do not have any friend, and this value is increased every time they add a new friend inside the SMP.
- **rBehaviour**: a random factor within the range [-5, 5]. Human behaviour is not always predictable and the agents' behaviour should not be either.

Since the final value should be in the range [0%-100%] - being 0% not willing to perform the action and 100% keenly disposed to do it - a set of weights has been assigned to each parameter. The most influential is the initial desire, which can represent a value as high as 75%, having a possible final weight of 25%, 50%, and 75%, increasing accordingly with the desired activity of the agents. The personality factor is the second one, with a maximum value of 25% and possible values of 8.33%, 16.66%, 24.99%. The number of friends is used to lineally reduce the willing to perform the action. Finally the random number adds a small variation of $\pm 5\%$. Since using only the function can provide a value outside of the desired range, we have restricted it using minimum and maximum functions.

Figure 5 displays the evolution of the fDesire value after several executions, using a sample of 100 agents each time (50 introverted and 50 extraverted). As expected, the introverted agents lose their interest faster than the extraverted agents and, consequently, they will end up with a lower number of friends.

Once an agent has decided that he wants to add a new friend (i.e. $fDesire > threshold$), the agent asks PESEDIA for the current user list, via API, in order to decide which user he will send the friendship request, from amongst the ones that are not already his friends. Then he sends his petition once again via API and waits for the acceptance or rejection. Once informed of the result, he proceeds, adding the user to his friend list if accepted, or adding the user to his rejection list if pertinent, to avoid asking this user for friendship in the future.

Additionally, the agents are able to decide if they want to accept a friend request using the same formula. In this case PESEDIA will inform the master agent that a user (may the user be a human or another agent) is trying to add one of the agents as a friend. The master agent will inform the corresponding

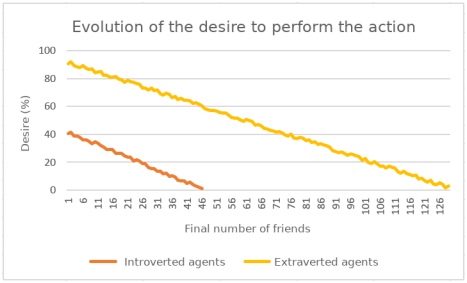

Fig. 5. Example of evolution of the $fDesire$ value.

agent who will decide either to add the user or not, informing the social network via API with the appropriate response.

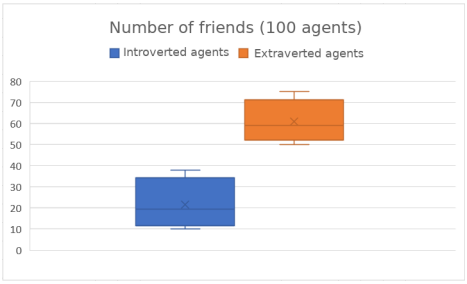

Fig. 6. Experiment result with 50 Extraverted agents and 50 Introverted agents.

Figure 6 displays the results obtained after a clean execution (in which PESEDIA has not any previous information nor users), using 100 agents (50 introverted and 50 extraverted), in an isolated agent environment. The expected results of this test considered that, usually, every extraverted agent would end up being friend of every other extraverted agent and some of the introverted ones, while the introverted ones have more probability of ending up as friends with extraverted agents mainly. In order to corroborate if our hypothesis is correct we have also run an smaller experiment with 2 extraverted agents and 10 introverted ones. The results of this test are shown in Figure 7.

5 Conclusions

This document proposes a framework that allows to populate a social network using a MAS, with agents that replicate behaviors based on human behavior patterns. The objective of the agents is to help transferring to the human users of the social network information about their behaviors, that may be harming their privacy, in a non intrusive (e.g. dialogues or pop-up windows) manner.

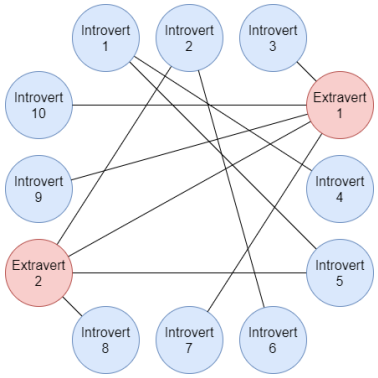

Fig. 7. Experiment result with 2 Extraverted agents and 10 Introverted agents.

Additionally, an initial approach to the proposed system has been developed using the Elgg social network development framework and the XMPP-based multi-agent system framework SPADE. Different experiments have been carried out to validate that the behaviour of the agents is accurate using the PESEDIA social network as a test environment. The aim of this work was to test how the agents interact within the environment of the SMP. It is our intention to deploy this system in the next summer school seminars in order to determine how its usage may impact the teenager's behaviours and patterns of usage.

Being now this infrastructure in place, new research opportunities arise. Firstly, the possibility of designing privacy-aware agents customized for each user, so that they can help them in a direct way to learn good practices in relation to privacy and to correct bad habits through the use of simulation scenarios powered by the agents. In this privacy-aware agents we will integrate the soft-paternalism techniques [1] and argumentation system [16] already developed in our research group. Secondly, it is possible to study in depth the aspect of behaviour simulation. This work mentions how the personality to be emulated can influence the behaviour of the agents, so it might be possible to design different models of agents based on psychological profiles. Finally, it is also possible to design agents that, based on the content posted by the users of the social network and their reactions, try to determine the social norms of the network.

6 Acknowledgments

This work has been funded thanks to the Spanish Government through project TIN2017-89156-R and predoctoral contract PRE2018-084940.

References

1. Alemany, J., del Val, E., Alberola, J., García-Fornes, A.: Enhancing the privacy risk awareness of teenagers in online social networks through soft-paternalism mecha-

nisms. International Journal of Human-Computer Studies **129**, 27–40 (2019)
2. Argente, E., Vivancos, E., Alemany, J., García-Fornes, A.: Educando en privacidad en el uso de las redes sociales. Educ. Knowl. Soc. **18**(2), 107–126 (2017)
3. Bell, L., Gustafson, J.: Interaction with an animated agent in a spoken dialogue system. In: 6th European Conf. on Speech Communication and Technology (1999)
4. Cassell, J., Thorisson, K.R.: The power of a nod and a glance: Envelope vs. emotional feedback in animated conversational agents. Appl. Artif. Intell. **13**(4-5), 519–538 (1999)
5. Cerrato, L., Ekeklint, S.: Different ways of ending human-machine dialogues. Proc. Embodied Conversational Agents (2002)
6. Franchi, E., Poggi, A.: Multi-agent systems and social networks. In: Handbook of Research on Business Social Networking: Organ., Manage., and Technol. Dimensions, pp. 84–97. IGI Global (2012)
7. Hollenbaugh, E.E., Ferris, A.L.: Facebook self-disclosure: Examining the role of traits, social cohesion, and motives. Comput. Hum. Behav. **30**, 50–58 (2014)
8. Hubal, R.C., Fishbein, D.H., Sheppard, M.S., Paschall, M.J., Eldreth, D.L., Hyde, C.T.: How do varied populations interact with embodied conversational agents? findings from inner-city adolescents and prisoners. Comput. Hum. Behav. **24**(3), 1104–1138 (2008)
9. Inc., F.: Facebook reports third quarter 2019 results (2019 (accessed January 27, 2020)), https://investor.fb.com/investor-news/press-release-details/2019/Facebook-Reports-Third-Quarter-2019-Results/default.aspx
10. Inc., F.: Instagram for business (2019 (accessed January 27, 2020)), https://www.facebook.com/business/marketing/instagram
11. Inc., T.: Twitter q3 '19 investor fact sheet (2019 (accessed January 27, 2020)), https://s22.q4cdn.com/826641620/files/doc_financials/2019/q3/Q3_19_InvestorFactSheet.pdf
12. Kökciyan, N., Yolum, P.: Priguardtool: A tool for monitoring privacy violations in online social networks. In: AAMAS. pp. 1496–1497 (2016)
13. Lewis, K.: The co-evolution of social network ties and online privacy behavior. In: Privacy online, pp. 91–109. Springer (2011)
14. Madden, M., Lenhart, A., Cortesi, S., Gasser, U., Duggan, M., Smith, A., Beaton, M.: Teens, social media, and privacy. Pew Research Center **21**, 2–86 (2013)
15. Patkos, T., Flouris, G., Papadakos, P., Bikakis, A., Casanovas, P., González-Conejero, J., Figueroa, R.V., Hunter, A., Idir, G., Ioannidis, G., et al.: Privacy-by-norms privacy expectations in online interactions. In: 2015 IEEE Int. Conf. on Self-Adaptive Self-Organizing Systems. pp. 1–6 (2015)
16. Ruiz Dolz, R.: An argumentation system for assisting users with privacy management in online social networks (2019)
17. Spottswood, E.L., Hancock, J.T.: Should i share that? prompting social norms that influence privacy behaviors on a social networking site. J. Comput-Mediat. Comm. **22**(2), 55–70 (2017)
18. Stutzman, F., Kramer-Duffield, J.: Friends only: examining a privacy-enhancing behavior in facebook. In: Proc. of the SIGCHI Conf. on human factors in computing systems. pp. 1553–1562 (2010)
19. Waters, S., Ackerman, J.: Exploring privacy management on facebook: Motivations and perceived consequences of voluntary disclosure. J. Comput-Mediat. Comm. **17**(1), 101–115 (2011)
20. Wisniewski, P.J., Knijnenburg, B.P., Lipford, H.R.: Making privacy personal: Profiling social network users to inform privacy education and nudging. Int. J. of Human-Computer Studies **98**, 95–108 (2017)