International Journal of Computer Science and Applications
© Technomathematics Research Foundation
Vol. 9 No. 2, pp. 70 - 87, 2012

E-Mail Prioritization using Online Social Network Profile Distance

SHALIZ REZAEE
School of Computing, Blekinge Institute of Technology
SE–371 79 Karlskrona, Sweden
shalizrezaee@yahoo.com

NIKLAS LAVESSON
School of Computing, Blekinge Institute of Technology
SE–371 79 Karlskrona, Sweden
Niklas.Lavesson@bth.se

HENRIC JOHNSON
School of Computing, Blekinge Institute of Technology
SE–371 79 Karlskrona, Sweden
Henric.Johnson@bth.se

Online Social Networks (OSNs) provide new ways for people to communicate with one another and to share content. OSNs have become quite popular among the general population. Their rapid growth has raised concerns about privacy and security on the one hand but increased efficiency in solving everyday online tasks on the other hand. Many predict that the OSNs of today provide a glimpse of the future Internet infrastructure. Whether or not that will be true is difficult to say but what is certain is that privacy, integrity, and security issues related to the world wide web will be as important then as now. This article presents a method for improving efficiency and security of E-mail applications by focusing on one of the core concepts of OSNs; the user profile, which both includes private and public information that the user shares to different parties and the social interaction between users. We present a generic method for comparing user profiles, by measuring the distance between user profile or user vectors and an algorithm that uses this method for prioritizing the E-mail inbox. We report on an experimental case in which the proposed method is used in conjunction with social information from Facebook to demonstrate the applicability as well as to motivate its theoretical foundation.

Keywords: E-Mail; Prioritization; Online Social Networks; User; Distance.

1. Introduction

The fact that that the future Internet architecture is reliable and trustworthy is of great importance for user-to-user communication. As the web becomes more interactive, several new ways exist to facilitate communication and meeting people. Most of the communication today is performed using E-mail conversation. The existence of unwanted messages has inspired the proposed research to restore E-mail trust between users who are socially linked together through Online Social Networks (OSNs). Due to the radical shift in the number of users in the social networks worldwide, there exist new opportunities to keep in contact as compared to the traditional web sites. The web is empowered with a new mode to adopt the real life social relationships into the digital world. Unlike the traditional web, which revolves around information, documents, and web items, the concept of OSNs revolves around individuals, their connections and common interest-based communities. OSNs are useful when it comes to keeping in touch with friends, relatives and colleagues. Millions of users are using OSNs on a daily basis in order to make new contacts, start research collaborations, perform information sharing and even to conduct political campaigns. Some OSNs are used for professional purposes, such as XING and LinkedIn, where it is possible to discover new business connections. Other OSNs are more friendship-oriented and are primarily used for communication, news feeds, entertainment and photo and video sharing. Notable examples of such OSNs are: Facebook, Orkut and MySpace. These networks provide new and interesting ways to communicate, share, and meet on the Internet. OSN have gone a long way towards mapping social connections between friends and peers in the real world. However, they are currently inadequate at adding social context to user-to-user communication [Tran *et al. (2010)*].

E-mails has been (more or less) reliable throughout most of its entire long history. However, in recent years we have observed that E-mail as a form of communication has been dominated by unimportant, unwanted, and even malicious messages. Therefore, we argue that the combination of social information (friendship intensity, interactions, geographical information, number of likes, wall posts, and so on) from OSNs and data analysis could help the users obtain a more reliable and socially-aware means of communication. The task of distinguishing trusted from untrusted messages automatically becomes much more complex. Thus, the question is how to automatically assign different levels of trust to E-mail messages arriving in the inbox. Our goal is to let the human user teach a computer system to prioritize E-mails by using social interaction data between the recipient and senders in conjunction with weights that can be adjusted automatically on the basis of user feedback.

1.1. *Aims and Scope*

The aim of this paper is to build upon a new perspective by which OSN user profiles and OSN applications can be analyzed. The core idea, which was first presented in a recent work [Lavesson and Johnson (2011)], is to provide a means for measuring the distance between user profiles and the distance between a specific user profile and a number of OSN applications of interest. In addition, we use the concept of personal spheres, which are essentially bounding boxes that enclose a certain user profile and other user profiles and external applications that are based on similar notions (for example, of: trust, privacy, or interests). In the presented study, we will show how the previously presented perspective and methods can be used to solve a completely different problem than what was studied in the original work. The perspective and method are first re-iterated in a general, theoretical context. We then report on an experimental case in which we apply the method to measure distances between users in one of the major OSNs, by leveraging social information about user interaction, in order to prioritize the E-mail inbox. The main contribution of this article is the presentation and analysis of an E-mail prioritization algorithm that is based on interactive user feedback and extraction of the social interaction data between recipient and sender.

1.2. *Outline*

Section 2 presents the background, terminology, and related work. Section 3 then describes the methodology of how to represent profiles in a metric space and to compare profiles. Section 4 reports on an experiment performed using social information extracted from Facebook, in which the proposed method is used in conjunction with Least Mean Squares to rank E-mail senders according to the level of trust between the recipient and the senders. Finally, Section 5 features analysis and discussions and Section 6 concludes the article and gives some pointers to future work.

2. Background

The major OSNs share four common traits: (1) they generally contain a large number of users, (2) the users have developed trust between each others and share similar interests, (3) it is easy for a user to register and create a personal profile, and (4) it is easy to develop new applications for users to accept and be spread among the trusted friends. Adding a friend then involves a confirmation step and the view of the user's profile is then normally limited to Friends Only or to Friends of Friends, unless the user wants the profile to be visible for Everyone on the web.

In accordance to other studies [Crescenzo and Lipton (2009)] [Zhang *et al. (2010)*], we believe that one of the major aspects of OSNs is the ability for a group or an individual to seclude information about themselves and thereby reveal themselves selectively. What content is considered private differ between individuals and between different cultural groups. This fact has been discussed by several social network theorists [Granowetter (1973)] [Granowetter (1983)] that highlight the relevance of social network relations regarding depth and strength. Privacy is in some situations related to anonymity, or the wish to remain unnoticed or unidentified in the public domain. It also calls for the possibility to hide information about the participation in the OSN or elsewhere. As large scale social networks become more common, the amount of information that is stored about users and user interaction makes it interesting to investigate possibilities to leverage this social information to make the online tasks and online experience in and outside of the OSN more efficient and enriching.

This type of social information, whether it consists of user information or user interaction statistics, has been leveraged in a number of widespread and successful web services. For example: Amazon recommends suitable products to its customer based on their previous purchases and their endorsements or reviews of a product, many newspapers provide personalized summaries of news based on their readers previous reading habits, massive multiplayer games such as World of Warcraft let players connect in a virtual world where they can see each other, communicate, and exchange items.

It turns out that many traditional types of communication, such as E-mail, can be made more efficient and secure by leveraging information from OSNs. In this article, we investigate the use of social information as a basis for creating an intelligent decision support tool for E-mail users. Most such users today have come to accept the fact that this type of service come with some strings attached and not all of them are good: one of the most negative aspects of E-mail as a means for communication is the amount of spam, scam, and uninteresting multiple recipient messages that regular users have to cope with. In this work, we rely on the user interaction habits displayed in a major OSN to personalize the prioritization of incoming E-mail.

2.1. *Terminology*

A *user profile*, in the OSN context, is a collection of personal data associated with a specific user. A user profile can store the user's interests, gender, birthday, religious beliefs, and other characteristics of the user. This information can then be exploited by systems, applications or other users in the OSN or, as we shall see, by non-OSN web applications. In Section 3 we elaborate on the concept of user profiles and we will show how it is possible to compared and visualize them.

The difficulty of defining *trust* is that different people have different explanations of trust that may bear various perceptions depending on the context [Deutsch (1973)]. First we must clarify our notation of trust. We use a widely cited definition of trust [Gambetta (2000)]:

Trust is a particular level of the likelihood with which a user evaluates that another user will perform a particular action, before he can monitor such action and in a context in which it affects his own action.

In this definition, the level of likelihood indicates that there exist levels of trust between users. Furthermore, in order to avoid inconsistency, we need to agree that higher values are better. In other word the higher the trust value, the more trusted the person is. Trust in this article refers to the attitude one user has toward another user rather than attitude toward a system or toward oneself.

Interpersonal ties are defined as connections between people, in which information is carried. Interpersonal ties, is often divided in three varieties: strong, weak, or absent. The strength of an interpersonal tie is a linear combination of the amount of time, the emotional intensity, the intimacy, and the reciprocal services, which characterize each tie [Granowetter (1973)].

After a user has joined an OSN, they are prompted to identify others in the network with whom they have a relationship. Depending on which OSN the label for these *friendship* differs: Friends, Contacts or Fan. Most OSNs require a bi-directional confirmation of the friendship. The one-direction ties are normally called as Fans or Followers. However, the term Friends can be misleading since the the social connection does in some situation not necessarily mean friendship and the reason why people connect are varied. Therefore, an interesting research area is the development of new methods to classify friendship and how to measure friendship intensity [Johnson *et al. (2011)*] in OSNs.

The *Facebook Platform* provides a set of tools that enable third party developers to integrate with the OSN users. This could be through applications on Facebook.com or external websites and devices. Facebook applications have two core components: a homepage and a profile box. Developers can then choose if the homepage content is proxied through Facebook or isolated in an iframe. For the proxied content the Facebook Markup Language (FBML) is used [Felt and Evans (2008)].

2.2. *Related Work*

Due to the increased use of OSNs, there is a growing number of studies that focus on using social network data for prioritization (scoring) of messages in order to filter unwanted messages in E-mail systems. The difference between each approach has to do with the way the concept of trust is represented, computed and used. Moreover, different ideas are used to apply social informatics to E-mail systems. In TrustMail, which is a prototype E-mail client, an approach is proposed that makes use of OSN reputation ratings to attribute different scores to E-mails [Golbeck and Hendler (2004)]. The actual benefit of this system is that, by using social network data, it identifies potentially important and relevant messages even if the recipient does not know the sender [Golbeck and Hendler (2004)]. The approach of combining whitelists and social networks is proposed in Reliable Email (RE) [Garriss *et al. (2006)*]. RE exploits social relationships between E-mail senders and recipients to accept potentially relevant and important messages by automatically broadening the users whitelist among socially connected users. Ostra utilizes the existing trust relationship among users to charge the senders of unwanted messages and thus block spam [Mislove *et al. (2008)*]. SoEmail considers the trust as an integral part of networking rather than working alongside of an existing communication system [Tran *et al. (2010)*]. SoEmail leverages social network data to rate the messages. The key feature of SoEmail is that instead of directly connecting the sender and the recipient, messages are routed through existing friendship links.

Additionally, a considerable amount of trials have been conducted to extract trust from OSNs. It has been suggested that users build up social connections with others who have similar alternatives [Abdul-Rahman and Hailes (1998)]. Therefore, they proposed a model in which trust is calculated based on recommendations from similar minded recommenders on a specific context rather than using information from direct experience. In one proposed model [Yu and Singh (2001)], the trustworthiness of a user is calculated based on direct experience with that user as well as the belief rating of her neighbors. At the same time, another team of researchers suggested a model that trust is measured from the performance of previous direct interaction with a user as well as the direct interaction and asking trusted users to recommend other users [Esfandiari and Chandrasekharan (2001)]. Social interactions (e.g., the exchange of messages between users) have been suggested as an indicator of interpersonal tie strength [Xiang *et al. (2010)*]. As a consequence, an unsupervised model has been developed to estimate the relationship strength from the interaction activity and the user similarity in the OSN [Xiang *et al. (2010)*].

Although all of the aforementioned approaches leverage social relationships for extracting trust, the applications are not designed to be automated in the sense that the user must explicitly score other users, score messages, create whitelists or adjust the credits. Furthermore, none of the aforementioned trust models are utilized to prioritize the E-mail senders. As far as we know, there is only one other such similar approach to this work [Banks and Wu (2009)]. In both this work [Banks and Wu (2009)] and the presented study, the source data are extracted from Facebook but there is a basic difference between the approaches: in the former [Banks and Wu (2009)], the trust score between each pair of users is assigned via a human contributed survey. In contrast, our method lets the user provide feedback about the ranking of E-mail senders and we then employ an automatic means for estimating weights that allows a more personalized ranking of E-mails by automatic means in the future. A potential advantage with the latter approach is that users precisely rate senders via rating E-mails without needing a mechanism to map the answers to the user preferences. Keeping in mind that the prioritization process of E-mails does not depend on the content of messages rather it based on senders and their level of trust with recipients. To the best of our knowledge, such a method has not been presented earlier.

3. Method

We will now elaborate on a new perspective concerning OSN user profiles by presenting a method for comparing user profiles and for determining whether or not an OSN application or an OSN user should be trusted by a particular user. First, we describe how user profiles can be defined in a metric space, which enables user profile comparison. Second, we show how this profile comparison can be conducted by measuring the distance between user profiles in the metric space. Third, we introduce the concept of personal spheres within the context of OSN user profiles and describe how even OSN applications can be transformed to the same metric user profile space and subsequently compared with a user's profile and his or her personal sphere.

To summarize, we present methods for measuring profile-to-profile distance or profile-to-application distance. The former can be used, for example, to determine which OSN friends to share media and information with and the latter can be used to determine whether or not the permission requirements of an OSN application conform to the privacy settings of one's user profile.

3.1. *Representing Profiles in Euclidean Space*

An OSN user profile can be quite complex and can, for example, include: basic personal information, contact information, marital status, date of birth, educational history, but it may also include or be associated with: the list of OSN friends, the list of blocked applications and users, or any other type of information or personalized setting. Ultimately, an OSN user profile is essentially just a list of values, where each value, or set of values, is associated with a parameter. On a higher level of abstraction, of course, sets of parameters can be associated with different parameter families or groups. Consider, for example, that parameters could be organized into the following two groups: user data and privacy settings. Elaborating on this example, each parameter group could be further divided into sub groups out of which the first example group of ours could include the following sub groups: image data, text data, video data, and sound data.

Let P represent the user profile space. A user profile can be defined as a tuple, $p \in P$, of n elements, $e_1, \ldots, e_n$, where each element, e_i, represent a profile parameter setting, i. For our purposes, it is sufficient that elements of p can represent either discrete or continuous variables, meaning that the profile could include information such as: date of birth (continuous), gender (discrete), general privacy level (discrete) and so on.

3.2. *Profile Distance*

By using the aforementioned profile definition, any profile, $p_j \in P$, can be represented by a point, j, in the real coordinate system, more specifically, the n-dimensional Euclidean space. Moreover, the distance between two profiles, p_j and p_k, can thus be computed using the Euclidean distance metric:

$$d(j,k) = \sqrt{(j_1 - k_1)^2 + \cdots + (j_n - k_n)^2} = \sqrt{\sum_{i=1}^{i<n} (j_i - k_i)^2}$$

However, our method is not limited to the use of Euclidean distance for measuring profile-to-profile distance. It is possible to enhance the usability of profile distance measurement by introducing a concept we like to denote the personal sphere: conceptually, we could describe this sphere as a combination of pieces of knowledge, experiences, psychological perspectives, preferences, and so on, that collectively describe an individual or a group of similar individuals. Add to this, that individuals outside of one's personal sphere may be regarded as different or dissimilar to one's own character. In fact, the sphere could define anything from a stratum of society to a particular field of activity. However, for our purposes, the personal sphere could be seen as enclosing similar user profiles. Similar, in this context, refers to the notion that the two user profiles share essential features, e.g., with respect to privacy settings or age group or that the level of social interaction between the two corresponding users is high.

3.3. *The Personal Sphere*

Using previous definitions, we are able to define the personal sphere as an ordinary sphere (or a hypersphere if $n > 3$) in the mathematical sense, using a central point that is equal to the profile point in question, say p_j, and a reasonable radius, r. We say that a user profile, p_k, is similar to p_j if it satisfies the following inequality and thus is included in the personal sphere of p_j:

$$r \leq \sqrt{(k_1 - j_1)^2 + \cdots + (k_n - j_n)^2}$$

We provide a simple example visualization of user profile distances and personal spheres in Figure 1. For cases where $n > 3$, it is also possible, and perhaps more feasible, to make use of multiple radii and a more general bounding box instead of a bounding sphere. Alternatively, the n-dimensional space can be transformed into a two-dimensional or three-dimensional space using multi-dimensional scaling in order to visualize the personal sphere.

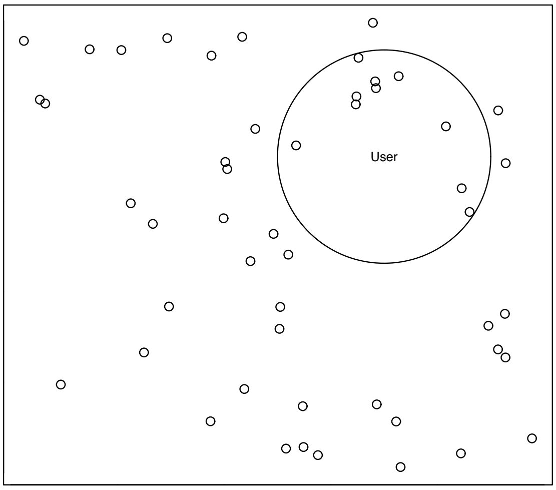

Fig. 1. A two-dimensional visualization, obtained using multi-dimensional scaling, of a personal sphere around a particular user depicted together with multiple friends (small circles).

3.4. *User Profiles and Applications*

The aim of this paper, as mentioned earlier, is two-fold: we elaborate on a way to measure the distance between different user profiles but, perhaps more importantly, we address the question of how to quantify how well other OSN users conform to the personal sphere of a specific OSN user. This is done by establishing a way to measure the distance between a user and his or her OSN friends. However, it is also possible to use the same methodology to measure the distance between an OSN application and an OSN user in order to determine whether the application is suitable for the user. This suitability test can be used for a range of purposes, for example when there is a need to determine how well an application's permission requests conform to a user's notions of privacy.

In the general case, an OSN service or an OSN application gives the user a certain functionality but more often than not, the application provides this functionality with some strings attached. A common argument is that, in order to personalize the service provided and to increase the social networking experience, the application may ask for certain permissions that can, for example, be associated with the extent to which private profile data is shared or whether or not the application can act on the user's behalf for certain actions.

For a majority of the large-scale OSNs available today, the permissions are directly related to settings that are customizable within the realms of the user profile. Thus, not only does the user need to customize general privacy and security settings in the user profile, the user also needs to consider the very same privacy and security aspects each time an application is to be used for the first time (and sometimes even on multiple occasions during the use of an application). In the OSNs reviewed, there is not a clear one-to-one mapping between application permission queries and user profile security and privacy settings, which means that a user can opt for a certain OSN behavior or privacy level in the user profile while unknowingly permitting conflicting behavior from an application.

In order to address this issue, we therefore let Q represent the application permission query space. Analogously to user profiles, an application permission query can be defined as a tuple, $q \in Q$, of m elements, $f_1, \ldots, f_m$, where each element, f_l, represent a required application permission, l. Again, for our purposes, it is sufficient that elements of q represent discrete variables. In fact, a Boolean variable is usually sufficient since a required application permission can be represented by a value of 1 whereas 0 refers to a permission that is not required. In order to measure the distance between the required permissions of an application and the personal sphere of a user, we need to provide a mapping (a function) between the user profile space, P, and the application permission query space, Q:

$$g : Q \to P$$

The function, g, can be quite complex and will vary substantially between different OSNs. However, theoretically, we may regard g as a general purpose mapping and it

Setting	Everyone	Friends of Friends	Friends Only	Other
My status, photos, and posts				X
Bio and favorite quotations			X	
Family and relationships				X
Photos and videos I'm tagged in			X	
Religious and political views			X	
Birthday	X			
Can comment on posts			X	
Places I check in to			X	
Contact information		X		

Table 1. Sharing settings excerpted from Facebook with example choices, marked with X, for each setting.

can thus be used to measure the distance between a user profile and an application or to find out whether an application is within the personal sphere of a user: an application permission query, $q_k \in Q$ is first mapped to P which yields $p_k \in P$. The query can now be regarded as a user profile and may thus be compared with personal spheres or other user profiles by way of distance measurement.

Turning back to our original idea on user-to-user, or profile-to-profile distance measurement, we now present a problem for which we show how to apply this type of measurement in a real-world OSN setting. The problem and proposed solution not only demonstrates the practical use of our method but also serves as a way to motivate and explain the theoretical basis of the method.

4. E-Mail Prioritization using Profile Distances

4.1. *Prioritization Algorithm*

The computational trust literature contains many different kinds of algorithms that can be categorized into two main groups; global and local trust computation algorithms. The global trust computation algorithm computes a single trust score for each individual in the network. In contrast, trust in the local trust computation algorithm is based on the attitude and preferences of individuals toward a certain user. The difference is that in the network with the global trust computation algorithm, a user always has a single score of trust. But in the network with the local trust computation algorithm each user has several trust scores depending on the viewpoints of other users in which the trust score is computed. The choice of global or local trust computation algorithm depends on the context. As has been mentioned by [Golbeck and Hendler (2004)], the global trust computation algorithm is applicable when the opinions of users are very similar. But in the context that users have very different idea about a topic, the global trust computation algorithm is not effective. In context of E-mail prioritization, the users opinions about the validity of an E-mail can be quite different. Thus, the local trust computation algorithm is a more suitable choice. To analyze how the trust score is computed based on the social information in OSNs; an algorithm (see Algorithm 1) is proposed.

Currently, Algorithm 1 computes the trust scores between users who are directly connected (friends) in the OSN. In the future improvement of the algorithm it is possible to change the algorithm in a way that it computes the trust scores between any random users. Each user in the OSN runs the algorithm individually to predict the trust scores with friends (see Algorithm 2). Algorithm 2 starts by getting an m-by-n matrix, X, that contains social information. The rows of the matrix (m) include all friends of the user and the columns (n) include the different criteria from which the trust score is calculated. In the algorithm, there is no limitation of the number of friends or the number of criteria. In addition to matrix, X, the algorithm gets an m-sized vector that is initialized with a personalized trust score for each friend. For example, assume that a user has four friends A, B, C, and D in the OSN and ranks friends based on trust. The ranked list can be used to produce artificial

Algorithm 1 Prioritization Weight Estimation

```
Require: $X$, an $n$-by-$m$ matrix where $m$ is the number of Facebook friends and $n$
    is number of criteria
Require: $U$, a vector of user IDs ordered according to feedback
Ensure: $Y$, a vector of size $m$ that denotes the actual trust
Ensure: $W$, a vector of size $n$ that will hold the estimated criteria weights
  for $i = 1 \to m$ do
    $Y[i] \leftarrow U[i]$
  end for
  $W \leftarrow (X'X)^{-1}X'Y$
  return $W$
```

Algorithm 2 E-Mail Prioritization

```
Require: $X$, an $n$-by-$m$ matrix where $m$ is the number of Facebook friends and $n$
    is number of criteria
Require: $W$, a vector of size $n$ that denotes the estimated criteria weights
Ensure: $Y'$, a vector of size $m$ that will hold the predicted trust
Ensure: $U'$, a vector of unordered user IDs
  $Y' \leftarrow XW$
  /* Order $U'$ according to the predicted trust to each user */
  $U' \leftarrow$sort$(Y', U')$
  return $U'$
```

trust scores that reflect this ranking. That is, friends might get ranks from 1 (less trustworthy) to 4 (most trustworthy). For example: $A = 2$, $B = 4$, $C = 1$, and $D = 3$. Then, the actual/artificial trust vector is [2, 4, 1, 3]. The weight for each criterion is given as another n-sized input vector, for which each element is initially set to zero. That is, W=[0,0,0,0]. The goal is to automatically predict a vector of trust scores in a way that approximates the user preferences. In order to compute a good estimation for the predicted trust vector, Y', the algorithm needs to compute an optimal set of weights for the different criteria. To do so, the algorithm uses least mean squares [Walpole *et al. (1998)*]. Finally the predicted trust vector, Y', is obtained from the inner product of the matrix, X, and the weight vector, W.

Note that the current algorithm prioritizes E-mails based on the senders, regardless of the content of the E-mails. This means that, if sender B is more trusted than sender A, all the E-mails from sender B are ranked higher than the E-mails from sender A in the inbox. Obviously, it is possible to combine a content-based filtering method with this algorithm in order to prioritize E-mails based on both the social relationship and the contents of the sent E-mail message. However, such a hybrid algorithm is considered as out of scope for this thesis. The algorithm may fail to correctly predict extreme situations in which the relationships to different friends do not correlate well with the trust towards the friends. For example, very close friends that meet each other regularly might not have intensive interaction in the OSN since their interaction is face to face or via the phone. Such a friendship might get a low trust score by the algorithm in spite of the fact that the friendship is strong. The opposite scenario can happen when one decides not to deny an acquaintance and to have some interactions to be polite while the user does not actually trust the acquaintance. However, such a limitation does not depend entirely on this algorithm rather it is an inherent part of all algorithms that are based on the users online social life since all aspects of real-world social relationships of users are not reflected in OSNs. Furthermore, the primary goal is not to recognize the negligible extreme relationships but to be able to categorize most of the friends in order to find the friends that are the least trusted to elevate recognizing potential senders of unwanted E-mails.

4.2. *Experimental Design*

As an experimental platform, Facebook is chosen because of several reasons. Firstly, Facebook is one of the most popular OSNs and boasts the largest number of users. A significant number of people sign in to Facebook daily, update their profile, share information, interact with their friends and so on. This makes Facebook a rich OSN in terms of data availability and it is up-to-date OSN in terms of frequently occurring status updates. Facebook popularity helps users to be in contact with friends, relatives, acquaintance as well as strangers. Consequently, Facebook users are highly connected to each other and therefore have several varieties of relationships with each other. Furthermore, Facebook provides an application programming interface

(API), Graph API, which enables developers to read and write data by using a SQL-style interface. It is entitled the Facebook Query Language (FQL).

In order to compute the trust score between users in Facebook, user profile data is needed. The basic idea is to extract information about user interaction, interest and background similarity with friends. It is tried to minimize the amount of data that is needed to collect to demonstrate the applicability of the method. For this purpose we decided to collect data of one Facebook user and 13 randomly drawn friends. In order to have a wide variety of social information data is divided into three different categories: interaction intensity, interest similarity, and background similarity and from each category, three criteria are selected. It is tried to choose criteria that are common to most Facebook users, and that fairly indicate the relationship quality in correlation with trust. To this end, number of wall posts, photo tagging, and, comments are selected in order to measure interaction intensity. In order to evaluate interest similarity, we opted for: number of shared likes, number of shared events, and number of mutual friends. Finally, we decided to measure background similarity based on: home town, current city, and family member. We do not claim that these criteria are in any sense optimal for measuring interaction intensity and similarity of users in OSNs. We understand that selecting another set of criteria or a different number of criteria would definitely change the priority but this fact does not matter since the user feedback is used to tune the criteria weights to restore the priority. What is important in this work is to examine the basic idea, that is, whether there is a relation between OSN user interaction and trust (at least according to the adopted definition of trust). The experiment is performed by extracting the number of wall posts, photo tags, comments, and, so on and then adding each criterion value into a new column in matrix X. Each friend-to-friend pair was assigned to a different row in X. The order of criteria, from left to right on each row was: friend id, wall post, photo tag, comments, likes, events, mutual friend, home town, current location, and family member. Matrix X used as the experimental case is shown in Table 2. A static user feedback vector, U, is used in the experiment and it is equal to [3,1,9,8,13,7,2,12,6,4,11,10,5]. In other words, we assume that the user would always give the same feedback. This vector reveals, for example, that the fifth friend has received the highest rank and the second friend has received the lowest rank. This feedback vector is used as the actual trust vector (i.e., Y=[3,1,9,8,13,7,2,12,6,4,11,10,5]) and based on vector Y, one weight for each criterion is computed (i.e., $W \leftarrow (X'X)^{-1}X'Y$). The idea is that, for a particular user, some criteria will have greater impact on the trust attributed to friends than other criteria. The goal is to estimate a set of weights that would yield a ranking that is as similar to the user's feedback as possible. In this experiment, the computed weight vector, W, is [-0.570, -6.321, 1.246, -16.488, 2.705, 0.522, 4.436, -2.533, 0.276]. Notice that for readability purpose, the values are truncated to three fractional digits but in the actual computation, raw values are used. The predicted trust vector, Y', based on the weight vector is equal to [2.425, 3.750, 8.502, 7.147, 12.627,

a	b	c	d	e	f	g	h	i	j
100	0	0	0	0	0	1	1	1	0
101	0	0	0	0	1	2	0	0	0
102	0	0	2	0	2	6	0	1	0
103	0	0	3	0	0	6	0	0	1
104	17	0	7	0	0	17	1	0	1
105	0	0	1	0	1	2	0	0	0
106	2	2	0	0	1	17	1	0	0
107	0	0	2	0	1	4	1	0	0
108	2	0	0	0	0	7	1	0	1
109	0	0	0	0	2	2	0	1	0
110	10	2	7	0	3	15	1	0	0
111	5	0	9	0	1	5	0	1	0
112	0	0	5	1	4	0	1	0	0

a	Friend id
b	Wall post
c	Photo tag
d	comment
e	likes
f	event
g	Mutual friend
h	hometown
i	Current location
j	Family member

Table 2. The X matrix

4.996, 2.232, 11.722, 7.227, 3.922, 10.767, 11.151, 5.000]. This vector implies, for example, that the fifth friend has received the highest rank and the seventh friend has received the lowest rank. Finally the friend IDs, are sorted in vector U' according to the predicted trust vector. In order to evaluate the correctness of the predicted trust vector, there is a need to compare the friends ranks based on the user feedback with the friends ranks based on the predicted trust (i.e., comparison of vector U with vector U'). The goal is to predict the friends ranks as close as possible to the user feedback. Thus, it is decided to compute the prediction error based on the number of displacement that is needed to convert vector U' to vector U. Moreover, 100 random feedback vectors are generated and based on these random vectors, weight and predicted trust vectors are computed as described earlier. Then via an algorithm (see Algorithm 3), the error of each predicted trust vector is computed by comparing vector U with U'. Finally an average error was computed from all 100 error computations. This average error is compared with the error of the predicted trust vector based on actual feedback. In addition, the worst case scenario

Algorithm 3 Prediction Error Computation

```
Require: $U$, a vector of user IDs ordered according to actual trust
Require: $U'$, a vector of user IDs ordered according to predicted trust
  $e \leftarrow 0$
  for $i = 1 \rightarrow m$ do
    $c \leftarrow 0$
    $j \leftarrow 1$
    while $U'[i]! = U[j]$ do
      $j \leftarrow j + 1$
    end while
    $c \leftarrow \|i - j\|$
    $e \leftarrow e + c$
  end for
  return $e$, the prediction error
```

is considered in which the values of the predicted trust vector are arranged in the inverse order in comparison with the actual trust vector in terms of vector indexes. This means that, the related vectors U and U' are arranged in the inverse order in terms of vector indexes. Consequently, each friend ID in vector U' needs to have the maximum number of displacements to be arranged in the same order as vector U. Obviously, in that case, the error will be maximized. The maximum error rate depends on the number of friends in matrix X and it can be calculated using of the following formula: assume n is the number of friends. Then, if n is an odd number the maximum number of displacements $= 1/2(n^2 - 1)$. If n is an even number, the maximum number of displacements $= 1/2n^2$.

4.3. *Experimental Results*

We measured the average error for 100 random feedbacks and the obtained result was 51.58 with SD= 11.36. If the hypothesis would be true, that the Facebook interactions and the level of trust are not correlated, the error obtained using real feedback should not be lower than the mean error of random trials. We then measured the error of the predicted trust obtained when prioritizing using the real feedback and the result is equal to 32.00 with SD= 0. As is shown in Table 3, the

Feedback	Mean Error (SD)
Feedback-based method	32.00(00.00)
Random feedbacks	51.58(11.36)
Worst case result	84.00(00.00)

Table 3. Results

error of our method is significantly lower than random feedback and is relatively low, given the small number of observations. This, we argue, suggests that inbox prioritization is achievable using the selected methodology and that trust is correlated with user interaction, at least when using the definition of trust provided in this work. This result supports the earlier suggested benefits of leveraging social informatics such as user interaction and interest similarity in OSN, in order to apply it in the context of E-mail prioritizing.

4.4. ***Analysis and Discussion***

The experimental case provided can easily be extended to include more comprehensive user profile descriptions and this extension can be made by just adding elements to the user interaction vector. Optionally, or rather, depending on the context, the user-to-user mapping function, g, can be re-defined more elaborately. No matter how many dimensions the defined user vector may have, the distance measurements and the introduction of the personal sphere will be conducted in the same manner. Thus, essentially, our method is highly generalizable within the Facebook context but arguably it is also generalizable to many of the other major OSN platforms. The perspective and method we have introduced can be used both for visualization and measurements pertaining to user profiles and applications in OSNs. Moreover, it provides the basis for automatically determining the order of E-mails in the inbox on the basis of social information extracted from Facebook. In this context, machine learning or general artificial intelligence techniques can also be used to replace LMS to create a more noise resistant solution. Additionally, the personal sphere can be employed to the provided case in order to introduce spam detection functionality: by using the trust score for distance measurement and E-mail ranking and letting E-mails be marked as spam if they are ranked lower than a chosen threshold.

Although this paper focuses on making E-mail use more efficient, there are many other domains in which our method can be applied. For example, in online dating or job finder social networks. The main functionality of such networks is to provide their users with good matches, that is, profiles that match their own profile. In the context of online dating, recent work has focused on the problem of learning user preferences [Pizzato *et al. (2010)*]. Interestingly, this work also states that the problem of inaccurate explicit user preferences is not confined to online dating but rather it is a problem for all domains in which users do not know precisely what they want or are unable to accurately specify their preferences. This very notion is unmistakably central in the presented study as well.

5. Conclusions

This paper has addressed E-mail inbox prioritization by leveraging social information extracted from modern Online Social Networks (OSNs). The task has been solved by focusing on one of their core concepts; the OSN user to user interaction.

We have elaborated on a new perspective on how to visualize and compare user profiles, namely by measuring the distance between user profiles and applications in the Euclidean space. We have also presented a way to detect when the distance is too great and this functionality can, for example, be used to introduce spam detection for the provided inbox prioritization problem or to find out whether an OSN application conforms to the privacy settings defined in a user profile. This detection is carried out by defining a bounding box (denoted in the presented study as the personal sphere) around the user profile and then checking whether users or applications reside outside the personal sphere of the particular user. We report on an experimental case in which the proposed framework and method are applied to Facebook in order to demonstrate the applicability of our approach as well as to motivate the theoretical foundation. In the context of this case, we describe how to define user profiles by extracting information about user-to-user Facebook social interaction. The analyses indicate that the proposed framework and method are feasible to use, at least in the Facebook context, and that they can provide users with an efficient means to prioritize their E-mail inboxes by leveraging social information extracted from Facebook. The analysis performed in context of the experiment indicates that Least Mean Squares is a suitable approach for estimating weights that can be used for generating a personalized Facebook friend trust indicator. This indicator is used to prioritize the E-mail inbox by ordering E-mail senders according to their trust scores. Based on a theoretical analysis of our method, we conclude that it can be adopted quite easily for use with, for example, techniques from artificial intelligence or machine learning in order to automatically determine a reasonable personal sphere radius and also to provide means for increasing robustness to noise in comparison to Least Mean Squares. However, the method proposed in this paper could also be employed for radically different reasons, for example, to assess user profile similarity in match making applications.

5.1. *Future Work*

We are planning to conduct a large-scale experiment inspired by the demonstrative experimental case featured in the presented study. The future experiment is going to feature a large population of E-mail users and trust scores will be computed to all of their Facebook friends. In addition to Least Mean Squares, a number of candidate weight estimation approaches will be evaluated and compared.

References

A. Abdul-Rahman and S. Hailes. A Distributed Trust Model. In *Proc. 1997 Workshop on New Security Paradigms*, pages 48–60, 1998.

Facebook Analytics and Advertising. http://adonomics.com.

L. Banks and S. F. Wu. All Friends are Not Created Equal: An Interaction Intensity Based Approach to Privacy in Online Social Networks. In *Proc. International Conference on Computational Science and Engineering*, 2009.

G. Crescenzo and R. J. Lipton. Social Network Privacy via Evolving Access Control. In *Proc. Fourth International Conference on Wireless Algorithms, Systems, and Applications*, pages 551–560, Springer, 2009.

M. Deutsch. The Resolution of Conflict. *American Behavioral Scientist*, 17(2), 1973.

B. Esfandiari and S. Chandrasekharan. On How Agents Make Friends: Mechanisms for Trust Acquisition. In *Proc. Fourth Workshop on Deception, Fraud and Trust in Agent Societies*, 2001.

A. Felt and D. Evans. Privacy Protection for Social Networking APIs. In *Proc. WEB 2.0 Security and Privacy*, 2008.

D. Gambetta. Can We Trust Trust? *Trust: Making and Breaking Cooperative Relations*, Electronic Edition, Department of Sociology, University of Oxford, 213–237, 2000.

S. Garriss, M. Kaminsky, M. J. Freedman, B. Karp, D. Mazieres, and H.Yu. RE: Reliable Email. In *Proc. Third Conference on Networked Systems Design & Implementation*, pages 22–22, 2006.

J. Golbeck and J. Hendler. Reputation Network Analysis for Email Filtering. In *Proc. First Conference on Email and Anti-Spam*,44, pages 54–58, 2004.

M. Granowetter. The Strength of Weak Ties. *American Journal of Sociology*, 78:1360–1380, 1973.

M. Granowetter. A Network Theory Revisited. *Sociological Theory*, 1:201–233, 1983.

H. Johnson, N. Lavesson, H. Zhao, and S. F. Wu. On the Concept of Trust in Online Social Networks. *Trustworthy Internet*, Springer, 2011.

N. Lavesson and H. Johnson. Measuring Profile Distance in Online Social Networks. In *Proc. International Conference on Web Intelligence, Mining, and Semantics*, May 2011.

A. Mislove, A. Post, P. Druschel, and K.P. Gummadi. Ostra: Leveraging Trust to Thwart Unwanted Communication. In *Proc. Fifth USENIX Symposium on Networked Systems Design and Implementation*, 2008.

L. Pizzato, T. Chung, T. Rej, I. Koprinska, K. Yacef, and J. Kay. Learning user preferences in online dating. In *Proc. Preference Learning Workshop – European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, 2010.

T. Tran, J. Rowe, and S. F. Wu. Social Email: A Framework and Application for More Socially Aware Communications. *Social Informatics*, 203–215, 2010.

R. E. Walpole, R. H. Myers, S. L. Myers, and K. Ye. *Probability and Statistics for Engineers and Scientists*, Prentice Hall, 1998.

J. Whittaker. Why Secure Applications are Difficult to Write. *IEEE Security & Privacy*, 1(2):81–83, 2003.

R. Xiang, J. Neville, and M. Rogati. Modeling Relationship Strength in Online Social Networks. In *Proc. 19th International Conference on World Wide Web*, 2010.

B. Yu and M. P. Singh. Towards a Probabilistic Model of Distributed Reputation Management. In *Proc. Fourth Workshop on Deception, Fraud and Trust in Agent Societies*, pages 125–137, 2001.

C. Zhang, Z. Jinyuan, and Y. Fang. Privacy and Security for Online Social Networks: Challenges and Opportunities. *IEEE Network*, 24:13–18, 2010.