Exam Learning dynamics: Measuring behavioral trust in social networks

Charlotte Hendrickx[1], Mathieu Parmentier[1], Olivier Renson[1] and Guillaume Buisson-Chavot[1]

[1] Université Libre de Bruxelles, Masters in Bioinformatics and Modelling

Abstract

Trust plays a crucial role in our social relationships. It enables the formation of coalitions and determines the information flow between people as well as the reliability of that information. In their article ”Measuring Behavioral Trust in Social Networks”, Adali et al. proposed a measure of the trust relationship between actors in social networks based on their behavior, hence the term ”Behavioral trust”. In this work, we will present their method, discuss their results, discuss the results we were able to reproduce and provide some further insights into the problem.

Introduction

Trust relationships shape our interactions with others. Trust creates a feeling of security between different people and determines who we interact with, with whom we share information, if we trust the information we get, what groups or coalitions we make,... It is thus a driving force in the creation of social networks. However, quantifying trust relationships can be tricky without asking people explicitly who they trust. Trust is also multifactorial; it depends on different personal parameters such as the relationship history of both people, the previous relationship they had with one another or the person's reputation.

In order to study the phenomenon of trust in social networks, it is necessary to start by simplifying the problem as to express them in an algorithmic form. Any social network is made up of actors, we will be particularly interested in the dyadic interactions between these actors, quantifying and orienting them since an actor A may trust an actor B but the reverse is not necessarily true. An interaction cannot systematically be translated into trust. Nevertheless, repeated interactions build a relationship and a relationship can result in relative trust.

In their article, Adali et al. tried to identify possible trust relationships in a dataset based on conversations on Twitter. They propose two measures of trust: the level of conversation between two agents, called *”Conversational trust”* and the number of informations that were propagated between different agents, called *”Propagational trust”*. No semantic analysis was performed here. The results are exclusively based on statistical features.

Related work

Researchers in the social sciences and computer sciences have developed various models, in particular on the emergence of trust in social networks and on trust that adjusts itself according to the reputation of the actors based on the number of interactions they make (4), (5), (1), (2). Then, the notion of trust following subjective recommendations (2) and the notion of security also based on trust (7) has been added to these models. Other researchers have given an explicit probabilistic interpretation of trust by quantifying it using their SUNNY algorithm (9). If an information is considered reliable (high confidence) then its source (the person who share the information) is reliable. All of the work described above is based on a semantic analysis. The dynamics are not taken into account.
In contrast to these works, we will describe trust using observed communication statistics, without using semantic information, and we will apply them to a dynamic network like Twitter. It is the notion of trust as a social link (8) that is conceptualized in this article (Adali et al.).

Problem description

Let us now formally define the problem. The input is a set of 3-tuples *(sender, receiver, time)* describing the communications between the agents of the network. We want to transform this input to a trust graph where nodes represent the agents (senders and receivers) and weighted edges represent the trust relationships between them.
The authors hypothesised that a trust relationship between two nodes A and B will result in certain behaviors, for example conversations. Such behaviors are not a guarantee of trust; we can have conversations with people we do not trust or converse very seldom with people we do trust. However, without analysing the content of the messages, it provides a good approximation. Another behavior is the propagation of

information from one person to another. We put the hypothesis that someone is more likely to propagate information if the source is someone he/she trusts. These events being captured by the *Propagational trust* algorithm.

Methods

In order to assess the validity of the methods presented in this article, we tried to reproduce some of their results on two datasets: a dataset coming from Twitter and a dataset of e-mails. Our code as well as the datasets and results are available here.

Conversational trust

The first algorithm the authors proposed in order to compute the trust between the agents is called *Conversational trust*. It rests on the hypothesis that the amount of conversation between two agents is a good approximation for the trust between them. The algorithm takes into account the length, the frequency and the balance of the conversations (only one agent speaks or both respond in equal amounts).

Conversational trust takes a set of 3-tuples *(sender, receiver, time)* as input. The output is a trust graph in which the nodes represent the agents and the edges represent the trust between them.

Let A and B be a pair of users and let $M = \{t_1, t_2, ..., t_k\}$ be a sorted list of times when a message was exchanged between A and B. From this set of messages M, Adali et al. constructed a set of disjoint conversations. Two consecutive messages $(t_{i+1} - t_i)$ are in the same conversation if the time between them is shorter than the average time between two messages ($\tau = \frac{t_k - t_1}{k}$) times a user-defined parameter S. Given a conversation $C = \{t_{i_1}, ..., t_{i_c}\}$, if $t_{i_c+1} - t_{i_c} < S \cdot \tau$, we add t_{i_c+1} to the conversation C, else we add it to a new conversation. Only conversations of more than two messages were considered. This thus gives us a list of conversations between two agents.

The conversational trust $T_c(A, B)$ is defined as follows:

$$T_c(A, B) = \sum_{i=1}^{l} ||C_i|| \cdot H(C_i) \tag{1}$$

$H(C_i)$ is a measure of the balance of the conversation computed using the density function where p is the fraction of messages in the conversation C_i that were sent by A:

$$H(C_i) = -p \cdot log\ p - (1-p) \cdot log\ (1-p) \tag{2}$$

This trust $T_c(A, B)$ is the weight of the edges connecting the nodes A and B in our trust graph. The trust has been normalised as to have a maximum weight of 1 and the trust of less than 0.01 have been removed.

In our implementation, we did not consider the number of conversations. All messages were considered and put in one conversation. The authors set the parameter S to 4, which did not yield conversations of more than one message thus returning no trust graph. Therefore we decided to leave out this parameter in our implementation.

Propagational trust

The second algorithm is called *Propagational trust* and is based on the hypothesis that when two agents trust each other, they will propagate the information they receive from one another to other external agents more often. This hypothesis is questionable and the propagation events are more rare than conversations. The results of the articles are based strongly on the *Propagation trust* and we could not replicate their results to that extend. However, we will present the algorithm because of its importance to the results of the article and the originality of its approach.

From the input *(sender, receiver, time)*, we first compute two sorted time lists of messages incoming to B and messages emitted by B. We then identify potential propagations as messages that came in to B (m_1) and were propagated by B (m_2) in a certain time interval (between τ_{min} and τ_{max}):

$$\tau_{min} \leq t_{m_2} - t_{m_1} \leq \tau_{max} \tag{3}$$

All potential propagations are found by finding pairs of messages that match the propagation constraint defined before. The propagations incoming from A are selected and compared to the expected number of propagations identified in a random network. This yields a list of potential propagations that are statistically significant.

Given the identified propagations between A and B, we can define two measures of trust: one based on the energy spent by agent B to propagate messages of agent A (Equation 4) and the other based on the fraction of messages of agent A, B propagated (Equation 5). With m_{AB} being the number of messages A sent to B, $prop_{AB}$ being the number of messages A sent to B that were propagated by B and $prop_B$ being the total number of messages B propagated.

$$T_p(B, A) = \frac{prop_{AB}}{prop_B} \tag{4}$$

$$T_p(B, A) = \frac{prop_{AB}}{m_{AB}} \tag{5}$$

According to Adali et al., there was no difference in the results yielded by both definitions of the propagation trust. We could not verify this because of our lack of results for this algorithm.

In our implementation, we were able to identify very few propagations and thus we did not filter the propagations that are significant compared to a random network.

Data

The first dataset we used to test the algorithms was constructed by taking all tweets emitted from a radius of 50km around Brussels on 31/12/2020 using the Twitter API. This enabled us to catch 100,000 tweets, which we filtered to only include responses between two users, thus discarding the original tweets. After this filtering, 36,924 remained, representing the conversations between two users. These tweets were then formatted as to match the paper's input; as tuples *(sender, receiver, time)*. We were limited by the free access of the Twitter API than enabled us to get only 2,500 in 15min and to access Tweets posted less than one week before. Because weren't able to collect as much data as we would have liked and in order to test the algorithms on similar data collected in another context, we decided to search for another dataset.

The second dataset we considered is the Enron e-mail dataset (6) available here. This dataset contains the e-mails of 150 collaborators of the Enron Corporation, corresponding to around half a million e-mails. 129,184 communications were identified and retreived from this dataset. These were subsequently converted to the input format of our algorithms (3-tuples).

In the original article (Adali et al.), each algorithm was tested on data from Twitter consisting of more than 2 million distinct users, which is way more than the dataset we were able to collect. We only considered the 36,924 directed messages of our dataset, compared to the 15,563,120 directed messages considered in the original dataset. The authors also included broadcast retweets, while we only identified replies to the tweets. We thus have less statistical power and were able to identify less conversations and propagations than in the original paper.

Results

Conversational trust

Network topology When applying the *Conversational trust* algorithm on the data from twitter, the trust network we constructed was composed of 5,528 nodes (3,861 senders and 3,170 receivers) and 9,757 edges.

We assessed the number of connections per node, which shows a distribution resembling the distribution expected from a scale-free network (see Figure 1). Most nodes are connected to only a few other nodes. However, some nodes are connected to as much as 53 other nodes. These nodes are highly active in the twitter community and respond to many tweets of other users.

We compared the distribution of the connections per nodes for the Twitter dataset with the connections per node averaged over 50 random networks (see Figure 2). We can see that the maximal number of connections is 7 in the random network. The topology of the network based on the Twitter data and that of the random network are thus totally different.

We also compared the results with the results averaged over 50 scale-free distributions that were made with the same number of nodes as our Twitter dataset (see Figure 3). We can see that there is way more interconnectedness in the scale-free network than in our real dataset.

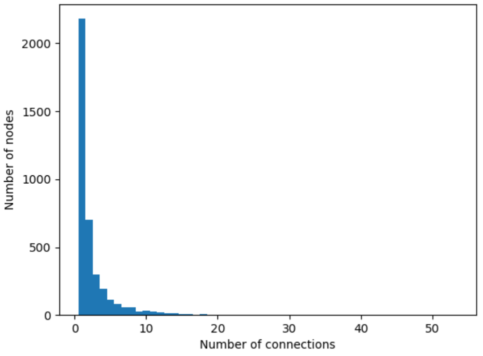

Figure 1: Number of nodes connected to x other nodes (Twitter dataset)

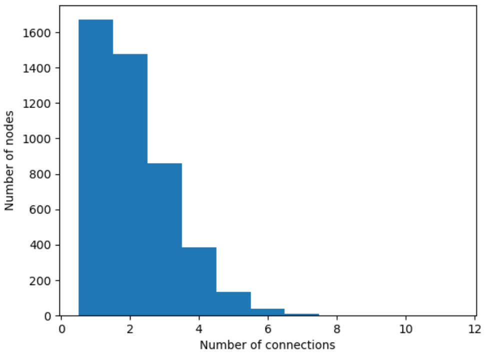

Figure 2: Number of nodes connected to x other nodes for an average of 50 random networks with the same number of nodes and edges as the networks from the Twitter dataset

For the Enron mail dataset, we were able to identify 1,966 nodes (1,650 senders and 655 receivers) and 3,998 edges. These represent people that engaged in e-mail conversations of more than two messages.

Like in the Twitter dataset (see Figure 1), we also see a scale-free like distribution of the network connections. In this dataset, we can see that the number of nodes having high connectivity is even bigger than in the Twitter data, some nodes having up to 67 connections.

Like in the Twitter dataset, the random network with the same characteristics as our dataset had way less connections (see Figure 8) while the scale-free network had a higher number as well as more highly connected nodes than the real data with less lowly connected nodes (see Figure 9).

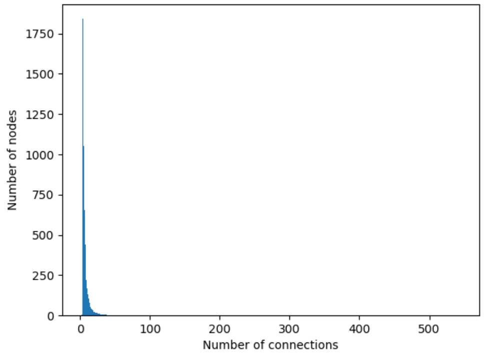

Figure 3: Number of nodes connected to x other nodes for an average of 50 scale-free networks with the same number of nodes and edges as the networks from the Twitter dataset

Distribution of the trust The two components of trust that are considered are:

- Length of the conversations
- Balance of the conversations

Figure 4 shows the distribution of the trust in the Twitter dataset. Edges with a trust equal to 0 are excluded. As expected, the distribution of the trust follows the distribution of the interconnectedness of the nodes.
The distribution of the trust values for the Enron dataset (see Figure 10) is similar to that of the Twitter dataset.

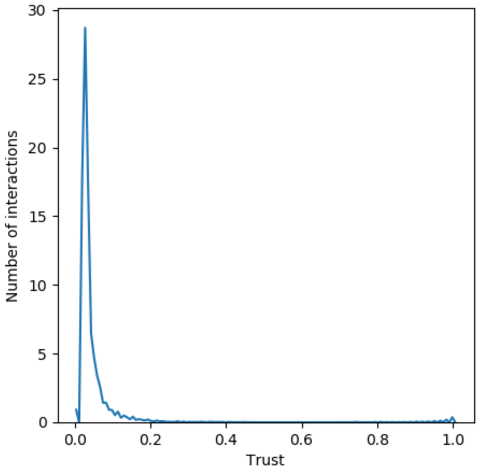

Figure 4: Distribution of the trust in the Twitter dataset (edges with $trust = 0$ are excluded)

The figure 11 shows the confidence score for different numbers of conversations with different configurations. We can see that it is the total number of messages that determine confidence. It is important to add that the two participants each in turn communicate (alternating messages). The confidence is normalised between 0 and 1 but for visualisation reasons it has not been applied to figure 11.

Trust based communities A key aspect of trust in social network is the creation of communities having a higher trust amongst them. We tried to identify such communities by applying a clustering method to the networks constructed from our dataset, as well as to a random network.

A visualization of the trust network is provided in Figure 5. All nodes with a total zero confidence were removed. The two upper graphs correspond to the data from Twitter and Enron (left and right respectively), while the lower graphs show the randomly generated trust networks that share the same characteristics (number of nodes and edges) as the Twitter and Enron datasets (left and right respectively). We can see that there is on average more trust in our data than in the randoms data. When considering the cumulative average of the trust in our data and the random network, our data shows a higher average level of trust.

As expected, not all agents are connected to each other in the networks above. There are sub-groups of ”trust”. Figure 6 shows the distribution of these different trust subgroups. As Figure 6 shows, we have a greater diversity of subgroup size for the Twitter and Enron data compared to the random data. It is interesting to note that the maximum size of a subgroup is around 3,600 nodes for random data versus around 360 for Twitter (both networks have the same number of links and nodes). A similar observation can be made with Enron data versus random data. The random data thus seem to form one big community, in which the agents are lowly connected to one another. The real datasets seem to yield more, but smaller clusters.

One possible explanation is the over-presence of spam in the Twitter and Enron data. A semantic analysis should be carried out to confirm this hypothesis. Another explanation would be that the random algorithm generates more conversations that are significant in terms of trust and in greater numbers (according to the network graph fig.5).

Propagational trust

The *Propagational trust* algorithm was also tested on our two datasets (Twitter and Enron). However, it was able to identify only a very small number of propagations (11 for the Twitter dataset). This very small size of the results didn’t enable us to discuss or draw conclusions in order to assess the validity of this algorithm.
Several explainations could be considered as to why this algorithm didn’t give the expected results. In most cases, such a result is due to a technical issue in the implementation. This is also possible here, but this is the case, we weren’t

able to identify it. In the article, the authors used the retweets as a validation of the results of the *Propagational trust* algorithm. We thus didn't consider the retweets as input of our algorithm. We tested the algorithm on two datasets and the results are highly dependant on the Twitter data. The output wasn't significant for the Enron mail data./

Discussion

This article aims at characterizing complex relationships like trust using simplistic measures like the number of messages exchanged during a conversation, the number of conversations or messages propagated between agents of the network.

We have some considerations about the hypotheses that were formulated in the article (Adali et al.). In the *Conversational trust* algorithm, there is a parameter S defining when a message is set in a new conversation or added to the present conversation. If this parameter is set too low, the conversations that will be detected will probably be short. Because conversations of less than 2 messages are not considered, only a very small number of conversations will get past this filtering stage and the conversations will probably be short, which will affect the trust value. We do not think that this parameter will lead to a better trust score.

The article on which this work is based was written in 2010. Back then Twitter was way smaller and less popular than it is today. Twitter has been used for testing and validating both the *Conversational trust* and *Propagational trust* algorithms. To validate *Propagational trust*, they used the retweets and made a distinction between the retweets to all followers and the retweets directed to some people. This functionality isn't available anymore on Twitter, which diminishes the use of their propagation trust and makes it hard for us to validate their results.

On social media, some posts containing only a small amount of information can get viral quite quickly ("buzz" phenomena, jokes, advertising, games, fake news,...). These posts are highly propagated, but the quality of the information is very low. The assumption that highly propagated information is trustworthy thus doesn't hold here. The exchange of jokes or the retweeting of some posts isn't the same as a real trust relationship. This is not identifiable without a semantic analysis of the content of the message and thus constitutes a limitation of their method.

The results in this article are also highly dependant on the social network data (Twitter in this case). We weren't able to produce any result of *Propagation trust* on the Enron dataset which is a dataset of e-mails and is thus placed in another context than that of the social networks like Twitter. The *Conversational trust* algorithm proved efficient in detecting trust relationships, but the *Propagation trust* was not and rests on disputable hypotheses. We thus doubt the usefulness of the *Propagation trust*.

Potential improvements

During the implementation, we thought about some possible improvements to the algorithm. The parameter S that separates the messages into conversations could, according to us, be suppressed. All messages of all conversations could be put into one conversation.

Another improvement would be to consider the semantic content of the messages. It would enable to identify positive messages (for example by identifying smileys) from negative ones. Another idea would be to identify question marks. These suggest an effort to sustain the conversation and an interest in the other person's thoughts, two indicators of trust.

Some social networks give access to the geo-localisation of their users. This of course poses privacy concerns, but it would enable to identify the true relationship between the users : people who are connected on social media and who co-localise know each other in real life and thus probably have a higher level of trust.

Conclusion

Trust is a complex feeling, granted after long interactions and several proves of worthiness in real life. We can trust people at different levels. All these variables make a really complex interaction out of trust. The algorithms that were presented in the article of Adali et al. (Adali et al.) are simplistic and measure interactions more than trust. The level of conversation is however a good proxy for the interaction level between people.

References

Abdul-Rahman, A. and Hailes, S. (2000). Supporting trust in virtual communities. In *Proceedings of the 33rd Hawaii International Conference on System Sciences-Volume 6 - Volume 6*, HICSS '00, page 6007, USA. IEEE Computer Society.

Aberer, K. and Despotovic, Z. (2001). Managing trust in a peer2 - peer information system. Proceedings of the Tenth International Conference on Information and Knowledge Management.

Adali, S., Escriva, R., Goldberg, M. K., Hayvanovych, M., Magdon-ismail, M., Szymanski, B. K., Wallace, W. A., and Williams, G. T. Measuring behavioral trust in social networks.

Beth, T., Borcherding, M., and Klein, B. (1994). Valuation of trust in open networks. pages 3–18. Springer-Verlag.

Buskens, V. (2002). *Social networks and trust*. The Netherlands: Kluwer Academic Publishers.

Enron Corp and Cohen, W. W. (2015). Enron email dataset.

Gray, E., marc Seigneur, J., Chen, Y., and Jensen, C. (2003). Trust propagation in small worlds. In *In Proc. of 1st Int. Conf. on Trust Management (iTrust'03*, pages 239–254.

Kelton, K., Fleischmann, K. R., and Wallace, W. A. (2008). Trust in digital information. *J. Am. Soc. Inf. Sci. Technol.*, 59(3):363–374.

Kuter, U. (2007). Sunny: A new algorithm for trust inference in social networks using probabilistic confidence models. In *In Proceedings of the National Conference on Artificial Intelligence (AAAI*.

Appendices

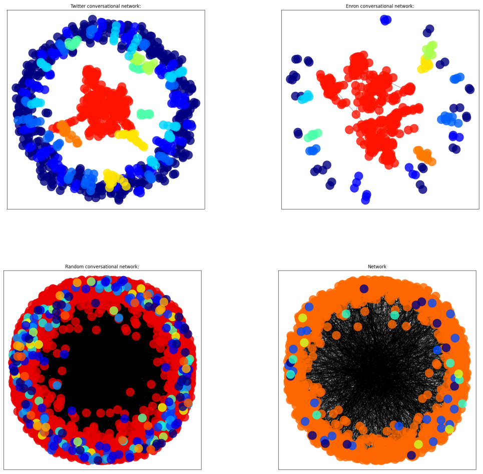
Twitter conversational network:
Enron conversational network:
Random conversational network:
Network

Figure 5: Network graph

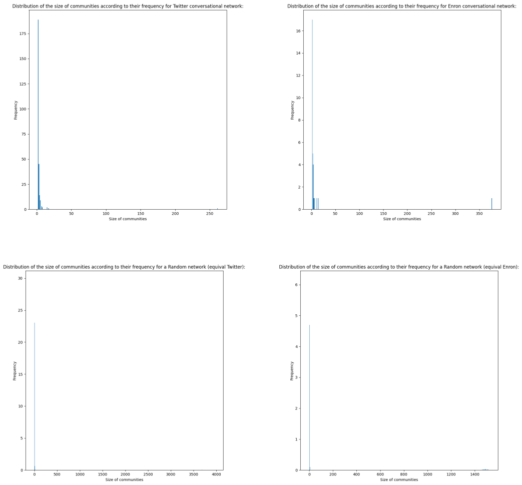

Figure 6: Distribution of cluster in the network

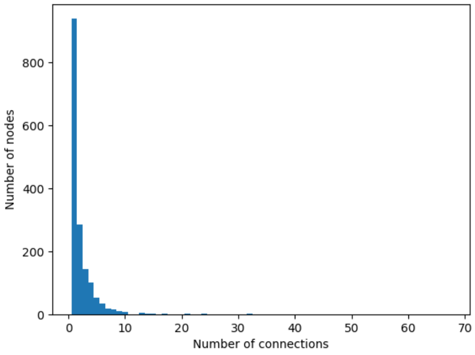

Figure 7: Number of nodes connected to x other nodes (Enron dataset)

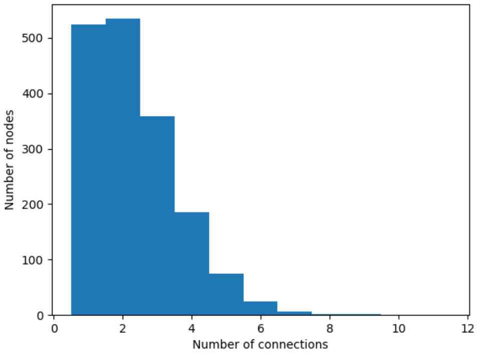

Figure 8: Number of nodes connected to x other nodes for an average of 50 random networks with the same number of nodes and edges as the networks from the Enron dataset

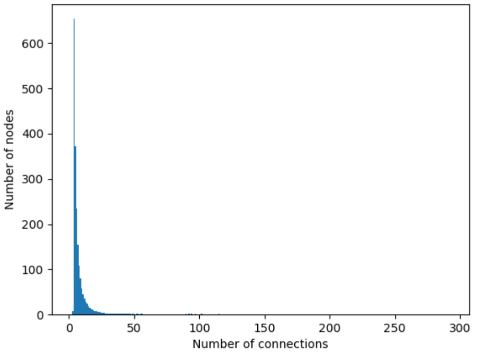

Figure 9: Number of nodes connected to x other nodes for an average of 50 scale-free networks with the same number of nodes and edges as the networks from the Enron dataset

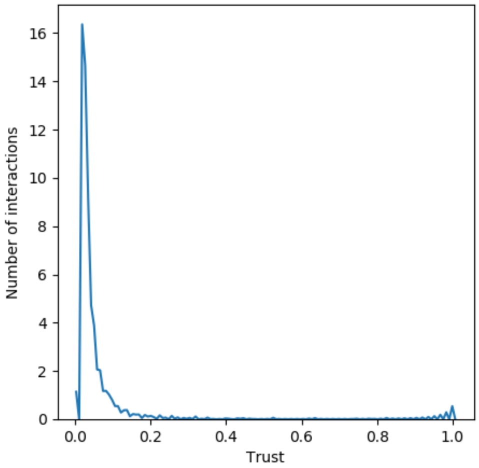

Figure 10: Distribution of the trust in the Enron dataset (edges with $trust = 0$ are excluded)

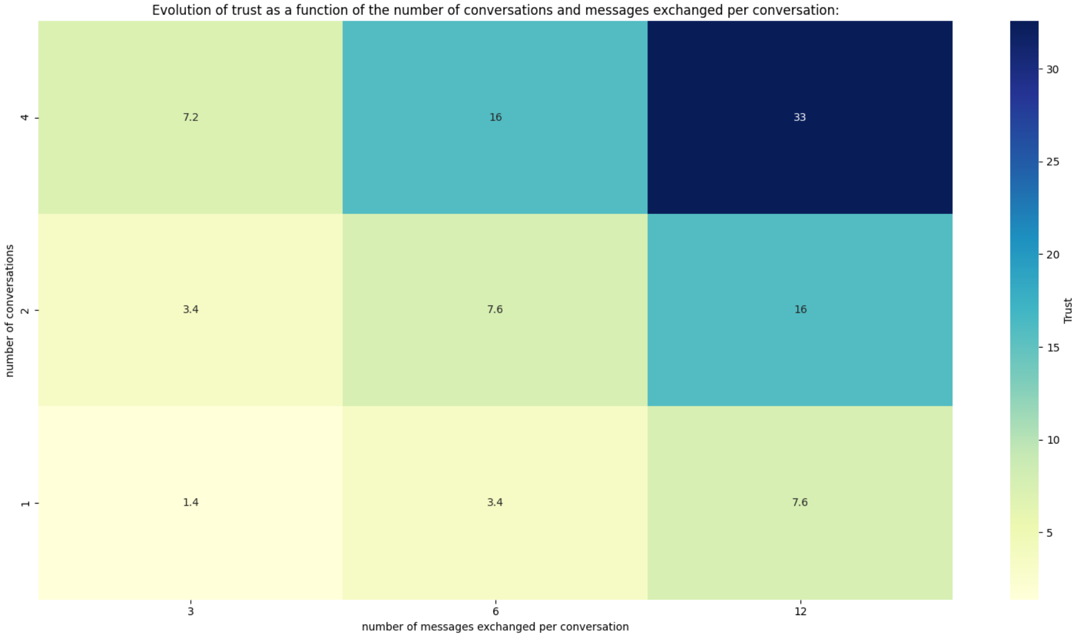

Figure 11: Evolution of trust as a function of the number of conversations and messages exchanged per conversation (x axis: number of messages exchanged per conversation, y axis: number of conversations)