

Improving the Recommendation Accuracy for Cold Start Users in Trust-Based Recommender Systems

Abdelghani Bellaachia, Deema Alathel*

The George Washington University, 2121 Eye Street NW; Washington, DC 20052; American.

* Corresponding author. Tel.: 202-994-8166; email: atheld@gwmail.gwu.edu

Manuscript submitted October 5, 2014; accepted May 31, 2015.

doi: 10.17706/ijcce.2016.5.3.206-214

Abstract: Recommender systems have become extremely popular in recent years due to their ability to predict a user's preference or rating of a certain item by analyzing similar users in the network. Trust-based recommender systems generate these predictions by using an explicitly issued trust between the users. In this paper we propose a recommendation algorithm called Averaged Localized Trust-Based Ant Recommender (ALT-BAR) that follows the methodology applied by Ant Colony Optimization algorithms to increase the accuracy of predictions in recommender systems, especially for cold start users. Cold start users are considered challenging to deal with in any recommender system because of the few ratings they have in their profiles. ALT-BAR reinforces the significance of trust between users, to overcome the lack of ratings, by modifying the way the initial pheromone levels of edges are calculated to reflect each edge's associated trust level. An appropriate initialization of pheromone in ant algorithms in general can guarantee a proper convergence of the system to the optimal solution. ALT-BAR's approach allows the ants to expand their search scope in the solution space to find ratings for cold start users while exploiting discovered good solutions for the sake of heavy raters. When compared to other algorithms in the literature, ALT-BAR proved to be extremely successful in enhancing the prediction accuracy and coverage for cold start users while still maintaining fairly good results for heavy raters.

Key words: Ant colony optimization, artificial agents, bio-inspired algorithms, recommender systems, trust.

1. Introduction

Due to the massive amount of information available to users online, there has been a strong demand to personalize a user's experience on the web and thus to supply the correct information to the right user. Such an adaptive environment is commonly provided by Recommender Systems (RS), which suggest to users the items that may be of interest to them [1]. One of the most common techniques applied in those systems are Collaborative Filtering (CF) techniques, which base their recommendation of items to the active user on items highly rated by similar users [2]. Although such techniques are simple to apply but they greatly suffer from their inability to provide good recommendations for cold start users. Researchers have been trying to devise ways to overcome this problem, such as allowing the users to express their level of trust in one another to improve the accuracy of predictions. Some of the major attempts in the literature include Massa et al.'s MoleTrust [3] and Golbeck's TidalTrust [4]. MoleTrust outperforms traditional CF techniques in trust-based recommender systems (TBRS) by using the explicit trust between the users instead of the similarity values. In [5] we proposed our bio-inspired T-BAR algorithm, which follows the Ant Colony Optimization (ACO) computational model [6], and it proved to enhance the overall accuracy and

coverage of recommendations in TBRS especially for heavy raters as opposed the results obtained by other algorithms.

We propose in this paper to further utilize the weights associated with edges (i.e. trust between users) within our algorithm, Averaged Localized Trust-Based Ant Recommender (ALT-BAR), to increase the ants' exploration of the solution space guided by the level of trust between users to overcome the lack of ratings for cold start users.

2. Related Background

2.1. Trust-Based Recommender Systems

Unlike traditional recommender systems (RS) where the system predicts item ratings for users based on their similarity to other users in the network, Trust-Based Recommender Systems (TBRS) rely on the concept of trust between users, rather than similarity, based on the belief that people tend to have high confidence in the opinion of users that they know or trust as opposed to depending on the opinion of *unknown* people found through a RS [7]. In TBRS, users are usually given the power to explicitly express the degree of trust they have in one another.

A web of trust (WOT) refers to the group of users in the network for which an active user has expressed trust towards, i.e. friends of the active user. An interesting characteristic of trust in such systems is that its value can be propagated within the network (creating a chain of trust) allowing items to be recommended not only from a user's WOT, but also from other users trusted by members of the WOT, i.e. friends of friends [8]. Trust propagation exposes the active user to a wider range of users in the network that can contribute to the process of item recommendation.

In traditional RSs that are based on CF techniques, the system input usually consists of item ratings provided by users in the network. However, TBRSs require also the list of each user's trusted *friends* along with their associated trust levels.

2.2. Ant Colony Optimization Algorithms

Ant Colony Optimization (ACO) is a family of algorithms that falls under the category of swarm intelligence (SI) algorithms [9]. ACO is based on a probabilistic approach for solving optimization problems by mimicking the behavior of ants within a colony when they are foraging for food. Typically, optimization problems are represented by a network consisting of nodes and edges. Usually weights are associated with those nodes and/or edges. Just like real ants, artificial ants in an ACO algorithm are dispatched over several iterations from one node in the network (that represents the nest) in order to search the solution space for *good* solution(s) to the problem (which are usually nodes or paths representing *good* food sources) [10]. While searching for a solution, an ant updates the pheromone levels on crossed edges to increase the probability of other ants crossing *good* edges and decrease their probability of choosing not-so-good ones. In this manner, the collaborative behavior of the individual ants allows the system to move from an unstable stage in early iterations where no good path (solution) is better than another to a stable stage where certain path(s) emerge as being the best ones to represent the solution to the problem.

2.3. Trust-Based Ant Recommender

Our trust-based ant recommender (T-BAR) [5] was the first successful application of an ACO algorithm to a TBRS to enhance the overall accuracy and coverage of recommendations. Within our system, the active user represents the nest and the users with a rating for the target item are considered food sources. The *good* solutions in this case are the nodes that represent users with a rating for the target item.

In T-BAR, at each step within an iteration, an ant k located at user x calculates the probability of crossing the edges connecting to a user y in x 's WOT then eventually crosses the edge that yielded the highest

probability. The probability p_{xy}^k is calculated as:

$$p_{xy}^k = \frac{(\tau_{xy})^\alpha (\eta_{xy})^\beta}{\sum_{z \in WOT_x} (\tau_{xz})^\alpha (\eta_{xz})^\beta} \quad (1)$$

where τ_{xy} is the pheromone level on the edge xy , η_{xy} is the trust T_{xy} expressed by user x towards user y , and α and β are influence parameters. The independent ants stop constructing their individual solutions within an iteration either when a certain search depth d is reached or when they reach a user that does not have a WOT.

T-BAR follows a two-level pheromone update model in which the pheromone levels on edges are updated on a local level and on a global level. The *local pheromone update* occurs when an ant k crosses an edge xy by which the pheromone level on that edge τ_{xy} is adjusted as follows:

$$\tau_{xy} = (1 - \rho) \cdot \tau_{xy} + \rho \cdot \tau_{xy}^0 \quad (2)$$

where ρ is the pheromone evaporation coefficient and τ_{xy}^0 is the initial pheromone level on xy calculated locally as [11]:

$$\tau_{xy}^0 = \frac{1}{\sum_{z \in WOT_x} T_{xz}} \quad (3)$$

The *global pheromone update* is carried at the end of each iteration after all ants finish constructing their solutions. In this step each constructed path's quality is assessed by calculating its *path trust* PT_k [3], [5] and the best paths P^{best} are selected as the ones satisfying the criteria $PT_k \geq PT_{threshold}$. The pheromone level on edges belonging to P^{best} are globally updated as:

$$\tau_{xy} = (1 - \rho) \cdot \tau_{xy} + \rho \cdot PT_k \quad (4)$$

At the end of the last iteration the system should have converged into the best solutions (paths) and that's when the predicted rating of an item i for the active user x , denoted by r_i^x is calculated as:

$$r_i^x = \frac{\sum_{u \in P^{best}} r_i^u}{n(u)} \quad (5)$$

where $n(u)$ refers to the number of *good* users u found on paths in P^{best} and have a rating for the target item i .

T-BAR's advantage over other algorithms that were proposed in the literature [3], [4] is that it considers all users with a rating for the target item along a path rather than using the item rating provided by the target user at the end of each path. Although all algorithms within the context of TBRs suffer from their inability to properly deal with cold start users, however T-BAR achieves overall significantly better results especially for heavy raters.

3. Proposed Averaged Localized Trust-Based Ant Recommender

Our proposed Averaged Localized Trust-Based Ant Recommender (ALT-BAR) is a variation of T-BAR that is altered to specifically solve the problem of cold start users. Just like T-BAR, ALT-BAR applies a dynamic approach based on ACO algorithms' probabilistic methodology to predict item ratings for users in a TBRs.

3.1. Importance of Initial Pheromone Level in ACO Algorithms

The initial pheromone level τ_0 in ACO algorithms is usually set using a constant or a pre-calculated value obtained by running a quick suboptimal algorithm such as the nearest-neighbor algorithm [10], [12] resulting in all edges having the same initial pheromone level. An ideal value for τ_0 should be one that is as close as possible to the average pheromone level that is expected to be deposited by an ant k on an edge in one iteration [13]. Thus, the value should be carefully selected because choosing a very small value will cause a slow convergence, which may result in not allowing the system to reach the optimal solution in a timely manner. On the other hand, using a large value for τ_0 will result in a fast convergence of the system in early iterations to suboptimal solutions and therefore not allowing the ants to explore other possible paths.

3.2. Rationale behind Proposed Algorithm

Trust plays a major role in any algorithm to be applied to TBRs. One of the reasons behind T-BAR's success is the fact that it greatly incorporates trust in its recommendation process by interleaving it with popularities of users and aspects of similarities between users in the search process, which allowed the system to find *good* users with *good* quality [5]. In other words, the quality of *good* users reached through T-BAR was high due to its ability to find users that have many items in common with the active user *and* that have a high trust level.

T-BAR had great results especially for heavy raters, which would make sense when considering the previously discussed reasons behind its success since heavy raters have rated many items and therefore reinforces the quality of *good* users found. However, the same cannot be said for cold start users and that would explain T-BAR's inability to perform well for them.

In order to overcome this problem, we decided to give trust a bigger role in guiding the ants in their search process without impacting their ability to perform well for other users. We saw a room for improvement in the way the initial pheromone level is calculated since T-BAR initializes all edges within a WOT using the same value without taking into consideration the individual differences between the edges in terms of trust (3). Therefore, we opted to reinforce trust whenever an ant needs to initialize the pheromone level on edges.

The effect of the introduced change in the pheromone initialization will impact:

- 1) The probability p_{xy}^k (1) because the initial pheromone level on an edge xy determines the initial probability of crossing that edge and since the edges within WOT_x will not have the same initial pheromone value then we expect edges with higher trust levels to have higher initial pheromone levels and thus a higher probability of being crossed.
- 2) The local pheromone update of τ_{xy} (2) because it involves the initial pheromone level to determine the amount to be deposited. If all edges within WOT_x have the same initial level then we would expect pheromone to increase at the same rate on those edges. But if the initial pheromone levels reflect their associated trust then pheromone will accumulate faster on edges with high initial pheromone levels.

3.3. ALT-BAR Algorithm

ALT-BAR follows the same methodology proposed in [5] to predict ratings for unseen items for the active user in TBRs. In ALT-BAR we propose to increase the influence of trust between users in the pheromone initialization step which would allow the effect be reflected in other aspects of the system without having to

alter much, as explained in the previous section. In ALT-BAR we assign each edge xy within WOT_x a different initial pheromone level τ_{xy}^0 that would reflect its associated trust level T_{xy} when compared to other edges in the neighborhood.

The initialization task is assigned to the individual ants where each ant initializes the pheromone level on edges within a WOT_x upon their first encounter in the system. Before an ant k calculates the probability of crossing an edge xy it has to check whether that edge has been initialized or not. If not, then the ant would utilize the locally available information within WOT_x to calculate τ_{xy}^0 as follows:

$$\tau_{xy}^0 = \frac{T_{xy}}{n(u_x) \cdot \sum_{z \in WOT_x} T_{xz}} \quad (6)$$

where $n(u_x)$ refers to the number of users in WOT_x . The reason behind using $n(u_x)$ is to average the initial pheromone levels within each WOT_x by its number of users and thus to avoid system fluctuations due to using high initial pheromone levels which would prevent the system from converging to the optimal solutions [9].

4. Experimental Evaluation

4.1. Dataset and Evaluation Metrics

Our proposed algorithm ALT-BAR was tested on the Epinions dataset since it is one of the few publically available datasets that allow access to both user ratings and explicitly issued trust values between the users. The dataset has 49,290 users who rated 139,738 unique items at least once. The users have expressed 487,181 trust statements. *Cold start users* comprise more than half of the users in the dataset and they are identified as being the users who each rated less than 5 items. *Heavy raters* on the other hand are the users who rated more than 10 items each. We also consider the category of *opinionated users* which refers to users that rated 5 or more items and whose standard deviation is greater than 1.5.

The *leave-one-out* technique was used to test ALT-BAR's ability to accurately predict item ratings. Our algorithm's performance is compared to the results obtained from running a basic CF algorithm that uses the Pearson Similarity and to Massa's MoleTrust algorithm [3]. It is worth noting that many researchers in the literature opt to compare their work to Massa's since it is one of the major techniques applied to TBRS [14]-[17].

The results of our empirical evaluation were assessed in terms of the *Mean Absolute Error* (MAE) along with the *ratings coverage* (RC) [18] which determines the algorithm's ability to generate a prediction for a hidden rating (regardless of the accuracy of the prediction). A major drawback in the MAE is that it does not take into consideration the percentage of cold start users or heavy raters in the system, which results in the MAE having the same weight for all user categories and thus in the case of Epinions the error for heavy raters would shadow the one for cold start users [3]. Therefore, the *Mean Absolute User Error* (MAUE) would give a better perspective of the results since it averages the MAE by the number of users in each category. In the same manner, the *users coverage* (UC) reflects the percentage of users in the dataset for which the algorithm was able to provide with at least one prediction.

4.2. Experimental Results

We compare the results obtained from running ALT-BAR on the Epinions dataset against several algorithms: *CF* which is a basic CF algorithm using the Pearson Similarity [19], *MT* which is Massa's MoleTrust algorithm [3], and *T-BAR* which is our basic trust-based ant recommender [5].

By comparing the MAE of the different algorithms (Table 1 and Fig. 1) alongside their RC (Table 2 and Fig. 2) we can see that T-BAR outperforms the other algorithms in terms of overall ratings accuracy and coverage. However since the results obtained for heavy raters can heavily affect the overall MAE of the algorithms, a quick glance at Tables 3 and 4 would give a better insight on how those algorithms rank against each other regardless of the number of users within each category. By considering the MAUE and UC in Fig. 3 and Fig. 4, the results show how our proposed algorithm ALT-BAR achieved a good overall balance by reaching a MAUE ~ 0.6 and a UC of 71% for all users.

Table 1. The MAE of the Algorithms across Different User Categories

Views	Algorithm			
	CF	MT	T-BAR	ALT-BAR
All	0.843	0.832	0.298	0.570
Cold Start Users	1.094	0.674	1.459	0.502
Heavy Raters	0.850	0.873	0.212	0.620
Opinionated Users	1.200	1.020	1.308	0.889

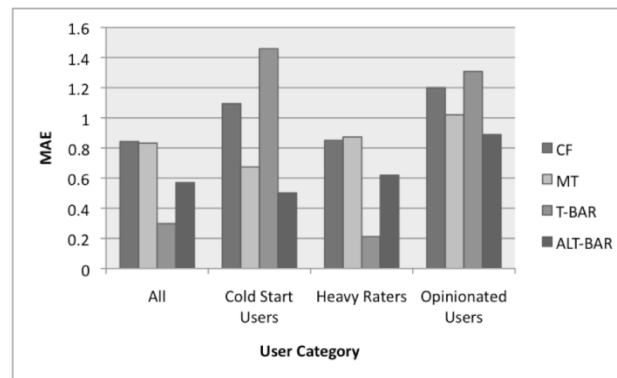


Fig. 1. The MAE of the algorithms across different user categories.

Table 2. The RC of the Algorithms across Different User Categories

Views	Algorithm			
	CF	MT	T-BAR	ALT-BAR
All	51%	28%	93%	90%
Cold Start Users	3%	11%	91%	53%
Heavy Raters	58%	31%	93%	92%
Opinionated Users	50%	23%	94%	54%

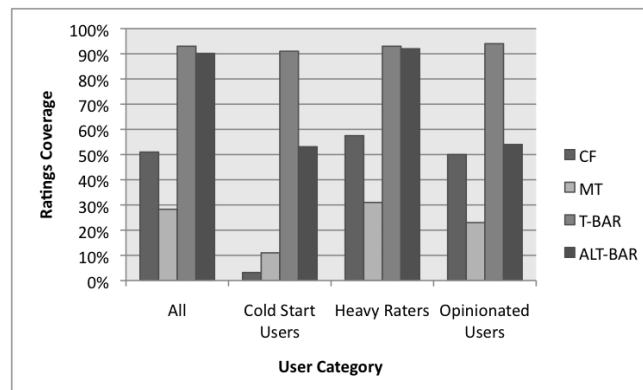


Fig. 2. The RC of the algorithms across different user categories.

When investigating the different user categories it is evident that T-BAR is always the better choice if the

dataset is composed mostly of heavy raters since it provides an amazing performance that surpasses the ones achieved by all other algorithms (MAE ~ 0.2 and 93% RC). On the other hand, ALT-BAR would be the perfect choice for any dataset that greatly suffers from cold start users. Although ALT-BAR's MAE for those users is ~ 0.5 which is somewhat close to MT's MAE of ~ 0.7 but by comparing their corresponding RC in Table 2 we can observe the big difference between ALT-BAR's RC of 53% and MT's RC of 11%.

Table 3. The MAUE of the Algorithms across Different User Categories

Views	Algorithm			
	CF	MT	T-BAR	ALT-BAR
All	0.938	0.790	1.203	0.592
Cold Start Users	1.173	0.674	1.581	0.430
Heavy Raters	0.903	0.834	0.282	0.683
Opinionated Users	1.316	0.938	1.262	0.755

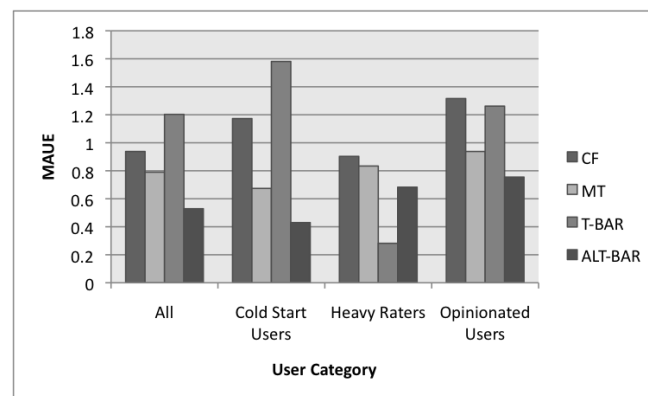


Fig. 3. The MAUE of the algorithms across different user categories.

Table 4. The UC of the Algorithms across Different User Categories

Views	Algorithm			
	CF	MT	T-BAR	ALT-BAR
All	40%	46%	96%	71%
Cold Start Users	3%	18%	97%	56%
Heavy Raters	86%	80%	93%	92%
Opinionated Users	61%	61%	94%	68%

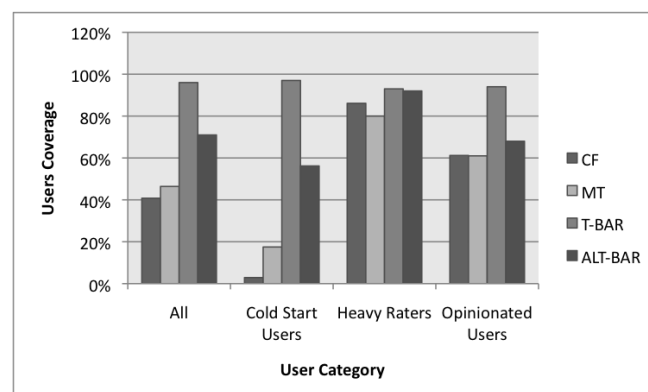


Fig. 4. The UC of the algorithms across different user categories.

Opinionated users are another category of users that can be a challenge to deal with in different algorithms, as is evident in the MAUE achieved by CF, MT, and T-BAR in Table 3. However, ALT-BAR

performs well for those users by decreasing the MAUE to ~ 0.8 with a 68% UC.

5. Conclusion

In this paper we proposed an algorithm called Averaged Localized Trust-Based Ant Recommender (ALT-BAR) to solve the problem of predicting item ratings for cold start users in TBRS. Since cold start users lack the availability of item ratings, ALT-BAR relies heavily on the trust issued between users to guide the ants in their exploration of the solution space. ALT-BAR achieves that by assigning each edge an initial pheromone level that reflects the edge's associated importance (trust).

Given the significance of the initial pheromone level in ACO algorithms in determining the system's convergence to the optimal solution, ALT-BAR's pheromone initialization approach proved to be feasible in terms of allowing the ants to expand their exploration scope for cold start users while managing to exploit the good discovered paths for heavy raters which results in finding good users with a rating for the target item in general.

Although ALT-BAR's expanded exploration of the other paths guided by the trust between users impacted its performance for heavy raters but despite this setback when compared to other algorithms in the literature, ALT-BAR managed to balance the overall trade-off between prediction accuracy and

References

- [1] Resnick, P., & Varian, H. R. (1997). Recommender systems. *Commun. ACM*, 40, 56–58.
- [2] Schafer, J. B., Frankowski, D., Herlocker, J., & Sen, S. (2007). Collaborative filtering recommender systems. In B. Peter, K. Alfred, & N. Wolfgang (Eds.), *The Adaptive Web* (pp. 291–324). Springer-Verlag.
- [3] Massa, P., & Avesani, P. (2007). Trust-aware recommender systems. *Proceedings of the 2007 ACM Conference on Recommender Systems* (pp. 17–24). Minneapolis, MN, USA.
- [4] Golbeck, J. (2005). *Computing and Applying Trust in Web-Based Social Networks*. Doctoral dissertation, University of Maryland at College Park.
- [5] Bellaachia, A., & Alathel, D. (2012). Trust-based ant recommender (T-BAR). *Proceedings of IEEE Conference of Intelligent Systems* (pp. 130–135). Sofia, Bulgaria.
- [6] Dorigo, M., Di Caro, G., & Gambardella, L. M. (1999). Ant algorithms for discrete optimization. *Artificial Life*, 5, 137–172.
- [7] Sinha, R. R., & Swearingen, K. (2001). Comparing recommendations made by online systems and friends. *Proceedings of the DELOS Workshop: Personalisation and Recommender Systems in Digital Libraries*.
- [8] Ricci, F., Rokach, L., Shapira, A. B., & Kantor, A. P. B. (2010). *Recommender Systems Handbook*, New York, NY, USA: Springer-Verlag.
- [9] Dorigo, M. (1992). *Learning and Natural Algorithms*. Doctoral dissertation, Politecnico di Milano.
- [10] Dorigo, M. (2001). Ant algorithms solve difficult optimization problems. *Proceedings of the 6th European Conference on Artificial Life, Vol. 2159. Lecture Notes in Artificial Intelligence* (pp. 11–22). Berlin: Springer-Verlag.
- [11] Bellaachia, A., & Alathel, D. (2014). A local pheromone initialization approach for ant colony optimization algorithms. *Proceedings of IEEE International Conference on Progress in Informatics and Computing*, Shanghai, China.
- [12] Dorigo, M., & Stützle, T. (2004). *Ant Colony Optimization*, MIT Press.
- [13] Dorigo, M., Bonabeau, E., & Theraulaz, G. (2000). Ant algorithms and stigmergy. *Future Generation Computer Systems*, 16, 851–871.
- [14] Avesani, P., & Massa, P. (2005). Moleskiing. it: A trust-aware recommender system for ski

mountaineering. *International Journal for Infonomics*, 1–10.

- [15] Golbeck, J. (2006). Generating predictive movie recommendations from trust in social networks. In K. Stølen, W. H. Winsborough, F. Martinelli, and F. Massacci (Eds.), *Lecture Notes in Computer Science: Vol. 3986* (pp. 93–104).
- [16] O'Donovan, J., & Smyth, B. (2005). Trust in recommender systems. *Proceedings of the 10th International Conference on Intelligent User Interfaces* (pp. 167–174). San Diego, California, USA.
- [17] Victor, P., Cornelis, C., Cock, M. D., & Teredesai, A. M. (2008). Key figure impact in trust-enhanced recommender systems. *AI Commun.*, 21, pp. 127–143.
- [18] Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, 22, pp. 5–53.
- [19] Massa P., & Avesani, P. (2004). Trust-aware collaborative filtering for recommender systems. *Proceedings of the Federated International Conference on the Move to Meaningful Internet* (pp. 492–508). Larnaca, Cyprus: Springer-Verlag.



Abdelghani Bellaachia received his B.S. and M.S. degrees in electrical engineering from Mohammadia School of Engineering, Rabat, Morocco in 1983. He later received his second M.S. degree in computer science from the George Washington University, Washington, DC, in 1991. He earned his Ph.D. degree from the same university in software systems a year later.

Since then he has been a faculty member at the George Washington University and is currently an associate professor there. His research interests include data mining, multi-lingual information retrieval systems, bio-informatics, design and analysis of algorithms, and parallel processing.

Prof. Bellaachia received the best paper award from IEEE's International Conference of Intelligent Systems for his work published in collaboration with Ms. Deema Alathel titled "Trust-based ant recommender".



Deema Alathel is a doctoral student at George Washington University, Washington, DC. She was born in Saudi Arabia in 1977. She received her B.S. and M.S. degrees in computer science from King Saud University, Riyadh, Saudi Arabia.

She has worked at the Institute of Public Administration in Riyadh as a technical support specialist for 2 years then as a faculty staff member after earning her master's degree. Her research interests include data mining, information retrieval systems, and bio-informatics.

Ms. Alathel received the best paper award from IEEE's International Conference of Intelligent Systems for her work published in collaboration with Prof. Bellaachia titled "Trust-based ant recommender".