Multi-agent recommendation system in Internet of Things

Agostino Forestiero
Institute for High Performance Computing and Networking
CNR-ICAR
Via P. Bucci, 7/11C - Rende (CS), Italy
email: agostino.forestiero@icar.cnr.it

***Abstract*—The Internet of Things (IoT) aims to bridge the gap between the physical and the cyber world to allow a deeper understanding of user preferences and behaviors. The interactions and relations between users and things need of an effective and efficient recommendation approaches to better meet users interests. Suggesting useful things in IoT environment is a very important task for many applications such as urban computing, smart cities, health care, etc., and it needs to be widely investigated. The goal of recommendation systems is to produce a set of significant suggestions for a user with given characteristics. In this paper, a multi-agent algorithm that, by exploiting of a decentralized and self organizing strategy, builds a distributed recommendation system in IoT environment, is proposed. Things are represented through bit vectors, the *thing descriptors*, obtained through a locality preserving hash function that maps similar things into similar bit vectors. Cyber agents manage the thing descriptors and exchange them on the basis of ad-hoc probability functions. The outcome is the emergence of an organized overlay-network of cyber agents that allows to obtain an efficient things recommender system. Preliminaries results confirm the validity of the approach.**

***Keywords*-Recommendation systems, Internet of Things, Multi-agent system**

I. Introduction

The basic concept of the Internet of Things (IoT) is that various things or objects can be interconnected with each other and achieve their common purpose [1]. The IoT infrastructures are growing hugely the amount available data on the Internet, which makes the traditional search mechanisms inadequate for managing the information. Moreover, due to the dynamic nature of smart objects, devices and services, involved in the IoT, intelligent and automated approaches are needed to support decision makers. Services able to perform "things recommendation" is a crucial step to promote and take full advantage of the IoT [2]. Searching related things (objects) is a key service in ubiquitous environments, such as the emerging IoT and smart environments. However, effectively searching for things is significantly more complicated than searching for documents because things are tightly bound to contextual information (e.g., location) and are often moving from one status to another [3]. Recommendation systems are an important research topic and several works have been proposed both in the industry and academia. These systems allow to to create a list of useful items for the users in a given contest. These systems can be built for documents, books, movies, news, articles, etc. The usefulness of an item or product or service is generally represented by a "rating", which indicates *how much a given user likes a particular item*. The items with an high value of rating are presented as recommendations for the user. Recommendation systems can be categorized as [4]: (i) *Collaborative Filtering* (CF), an item is recommended to the user according to the past ratings of all users. The approach evaluates the utility of the item *i* for the user *u* by estimating the usefulness assigned to item *i* by the users *v* who are "similar" to user *u*; (ii) *Content-based recommending* an item is recommended if it is similar to items that the user has chosen in the past. Information retrieval (IR) technics address this problem, where the content associated can be handled as a query, and the unrated documents marked with a similarity value to this query. Otherwise, the documents can be converted into word vectors, and then averaged to obtain a prototype vector of each category for a user, as showed in [5]; and (iii) *Hybrid approaches* in which collaborative and content-based approaches are combined. Computing the similarity between two users can be used various approaches, but the most popular are correlation and cosine similarity. In the first approach, the Pearson correlation coefficient used to compute the similarity is reported in formula (1), where **I** is the whole set of items rated by both users *u* and *v* [6].

$$S(u,v) = \frac{\sum_{i\epsilon I}(r_{u,i} - \bar{r_u})(r_{v,i} - \bar{r_v})}{\sqrt{\sum_{i\epsilon I}(r_{u,i} - \bar{r_u})^2 \sum_{i\epsilon I}(r_{v,i} - \bar{r_v})^2}} \quad (1)$$

The value of the rating *r* for user *u* and item *i* is computed as an aggregate of the ratings of some other users for the same item *i*.

The cosine-based method [7] uses two vectors in *n-dimensional* space to represent the users *u* and *v*, and *n* will be $|I|$. The cosine of the angle between two vectors, as reported in formula (2), can be computed to measure the similarity between them, where $\overrightarrow{u} \cdot \overrightarrow{v}$ indicates the *dot-product* between the vectors $\overrightarrow{u}$ and $\overrightarrow{v}$.

$$S(u,v) = cos(\overrightarrow{u}, \overrightarrow{v}) = \frac{\overrightarrow{u} \cdot \overrightarrow{v}}{|\overrightarrow{u}|_2 \times |\overrightarrow{v}|_2} \quad (2)$$

Collaborative and content-based approaches use the same cosine measure from information retrieval. But, in content-

978-1-5090-6611-7/17 $31.00 © 2017 IEEE
DOI 10.1109/CCGRID.2017.123

based recommendation systems measures the similarity between vectors of weights, whereas, in collaborative systems measures the similarity between vectors of the actual ratings specified of the users.

The heterogeneity of possible scenarios, arising from the massive deployment of an enormous amount of smart objects, imposes the use of sophisticated and innovative models and algorithms. In this paper, a multi-agent algorithm for building a things recommendation system, is proposed. The algorithm is able to organize the " things" of an IoT environment in order to improve localization operations. Each smart object is associated with a single "cyber agent", which represents it in a cyber layer. All cyber agents work together, in a peer to peer fashion, in order to organize themselves and improve the performances of the system [8]. As in peer to peer system, a *thing descriptor*, i.e. a bit vector, is exploited to describe a smart object. The presence or absence of a given characteristic can be represented thorough the value of a bit [9], [10], or an hash function locality preserving can be employed to map things in thing descriptors [11], thus similar things descriptors are assigned to things with similar characteristics. The cyber agents exchange among them the thing descriptors following a bio-inspired strategy. The outcome is that a featured regions are crated and similar things descriptors are managed by neighbor cyber agents. Thanks to this organized cyber layer the discovery operation faster, and it is highly probable to find similar and useful thing descriptors – recommendations – close to the target describer. In the rest of the paper a preliminary version of the algorithm is introduced in section II and an initial experimental analysis is showed in section III.

II. Things recommendation system

In this section the multi-agent algorithm for building a recommendation system in IoT environment, is introduced. Things are described by *thing descriptors*, i.e. bit vectors obtained through a locality preserving hash function. The thing descriptors are assigned to cyber agents that exchange them with each other to achieve a spatially organization. The exchanges are performed so that similar thing descriptors, thus representing things with similar features, will be kept and managed by neighbor cyber agents. The process is progressively and continuously performed by each cyber agent through simple and local operations. Probability functions – that steer the operations of keeping or delivering thing descriptors – are performed autonomously by each cyber agent and globally a logical organization emerges. The probability functions derive from biological systems in which unaware entities works locally in order to produce a global intelligent behavior [12].

A cyber agents, for each thing descriptor managed, evaluates the probability function P_d, reported in formula (3). Depending on the result, it decides if delivers the think descriptor towards a linked cyber agent. A given number of hops to perform, HtP, is associated to a delivered think descriptor. It represents the number of the hops that it has to perform before it can be evaluate by a cyber agent. The probability function P_k, reported in formula (4), is evaluated by a cyber agent when a think descriptor arrive from another linked cyber agent and it can be evaluate, i.e. has performed HtP hops.

$$P_d = \left(\frac{k_d}{k_d + F_{sim}}\right)^2 \quad (3)$$

$$P_k = \left(\frac{F_{sim}}{k_k + F_{sim}}\right)^2 \quad (4)$$

The parameters k_d and k_k, whose values are comprised between 0 and 1, can be tuned to modulate the degree of similarity. Here k_d is set to 0.1 and k_k is set to 0.5 [12]. The probability functions are based on a similarity function, F_{sim}, that measures the average similarity of a think descriptor t_d with all think descriptors located in the *local region*. All cyber agents reachable with a given number of hops from the cyber agent *a*, represent the local region *R* for the cyber agent *a*. The similarity function F_{sim} for a think descriptor $\bar{t_d}$ in local region *R* is reported in formula (5).

$$F_{sim} = \frac{1}{N}\sum_{t_d \epsilon R} N_{c_a} \cdot \left(1 - \frac{1 - cos(t_d, \bar{t_d})}{\alpha}\right) \quad (5)$$

N is the overall number of think descriptors in the local region *R*, N_{c_a} is the number of think descriptors maintained by each cyber agent, while $cos(t_d, \bar{t_d})$ is the cosine distance between t_d and $\bar{t_d}$. The similarity scale parameter α is set to 2. The value of F_{sim} assumes values ranging between -1 and 1, but negative values are fixed to 0.

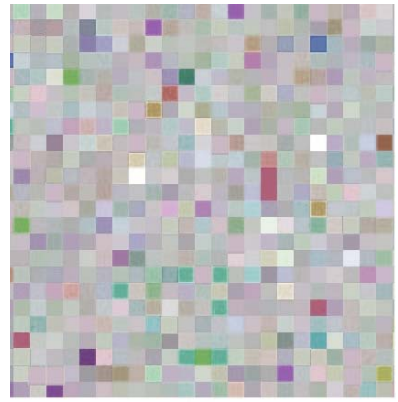
Designing a discovery algorithm that exploits the logical reorganization in order to obtain a set of recommendable things, is very simple and intuitive. Queries are issued by a device to search a given thing descriptor representing the wished thing. Usually, in non-informed environment, such as unstructured P2P networks, it is forwarded through the neighbors to collect as many target as possible. Thanks to the organized cyber layer, the queries can be forwarded towards the cyber agent with the maximum value of similarity with target thing descriptor. The similarity value between a thing descriptor and a cyber agent can be obtained by computing the similarity value between the thing descriptor and the average of the values of all thing descriptors managed by the cyber agent. The query is forwarded towards the neighbor cyber agent with the maximum value of similarity, based on formula (5). The same operation is done by each cyber agent when it receives a query. The query, going across the network of the cyber agents, collects a set of thing descriptors similar to the target thing descriptor. When none neighbor has a similarity value greater than the value of current cyber agent, or the maximum number of admissible query hops is finished, the search finishes and the query is forwarded to the "asking" cyber agent. The list of the thing descriptors collected by the query can be exploited to produce a set of suggestion/recommendation to the user.

III. Experimental Results

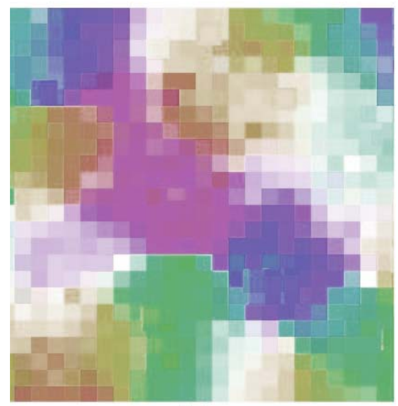
An event-based simulator was implemented to evaluate the performance of the algorithm. Each cyber agent manages about 15 thing descriptors and is linked to 8 cyber agent on average. The things descriptors have been obtained using a locality preserving hash function to guarantee that similar things have similar descriptors. A graphical description of the logical reorganization is reported in Figure 1. Each thing descriptor is associated to a RGB color and the cyber agent is represented with the color of the thing descriptor managed with the maximum number of element. A portion of the cyber agent network is photographed: (a) at *Time* = 0 time units, i.e. when the process is starting and the thing descriptors are randomly distributed and (b) at *Time* = 50,000 time units, i.e when the process is in a steady situation. Notice that similar thing descriptors are located in the same region and among near regions the color change gradually, which proves the spatial sorting on the cyber layer. The traffic load, T_{load}, i.e. the average number of thing descriptors processed per time unit by a cyber agent, that does not depend on the number of cyber agents, but only on the average number of sending performed by a new cyber agent F_g and on the frequency of their movements across the cyber agents network $1/T_m$, as shown in formula (6). In the simulation scenario, each cyber agent performs a single operation (delivery/keep) about every 20 time units, which can be considered an acceptable value.

$$T_{load} = \frac{N_{t_d}}{N_{c_a} \cdot T_m} = \frac{F_g}{T_m} \qquad (6)$$

Figure 2 shows the average number of operations per time units performed by each cyber agent when the HtP changes. Notice that the value of the load changes according to the value of the HtP of the thing descriptor and the reorganization process is accelerated if HtP is higher, because they can cover the network quickly. The maximum value of HtP is a compromise between the traffic load tolerable and the rapidity and efficiency of the reorganization. It was possible to note during the simulations that the processing load does not depend on system parameters such as the average number of thing descriptors handled by a cyber agent or the number of cyber agent. This confirms the scalability properties of the algorithm. A spatial index of sameness of the cyber agent network was defined to evaluate the goodness of the algorithm. For each cyber agent, the similarity among the thing descriptors managed and all thing descriptors in the local region, is calculated. The values of the sameness is averaged for all cyber agents. Our aim is to increase the similarity as more as possible. It would mean



(a)



(b)

Figure 1. Snapshots of a portion of the cyber agent network when the process is starting (a), and when the process is in a steady situation (b).

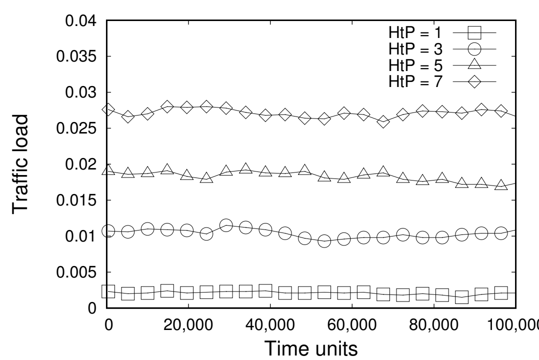


Figure 2. The traffic load generated by the algorithm when the number of hops to perform, HtP, of a thing descriptor ranges from 1 to 7

that similar thing descriptors are located into neighbor cyber agent and an effective sorted layer is becoming. Figure 3 shows the similarity of the whole cyber agent network when for different values of the dimension of the thing descriptors. It is possible to note how the logical reorganization is achieved independently of the dimension. The scalability of the algorithm is confirmed analyzing its behavior when

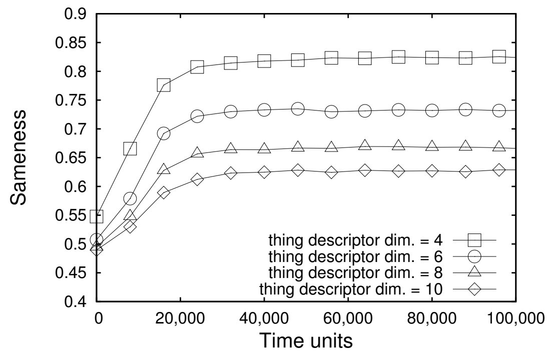


Figure 3. Similarity of the whole cyber agent network when the dimension of the thing descriptors ranges from 4 to 10;

the network size is varied. Figure 4 reports the values of

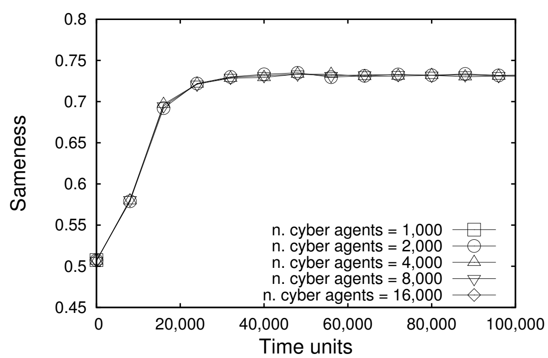


Figure 4. Sameness of the cyber agent network when the number of the cyber agent involved ranges from 1,000 to 16,000

sameness when the number of the cyber agent involved ranges from 1,000 to 16,000. Notice that the number of the cyber agent involved in the logical reorganization, has no detectable effect on the sameness of the cyber agent network.

IV. Conclusion

A distributed and self-organizing algorithm to build a distributed recommendation system in IoT environment, was introduced. The recommendable things are described through metadata obtained by exploiting of a locality preserving hash function able to map similar things into similar metadata. Cyber agents manage the thing descriptors and autonomously decide to delivery/keep them by exploiting of tailored probability functions. The outcome is an logically ordered cyber layer of cyber agents, in which similar things are managed by neighbor cyber agents. This allows discovery and recommendation operations faster. Experimental results show as the algorithm achieves an effective reorganization of thing descriptors and the performance obtained are very encouraging.

References

[1] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[2] L. Yao, Q. Z. Sheng, A. H. Ngu, and X. Li, "Things of interest recommendation by leveraging heterogeneous relations in the internet of things," *ACM Transactions on Internet Technology (TOIT)*, vol. 16, no. 2, p. 9, 2016.

[3] S.-M. Choi, H. Lee, Y.-S. Han, K. L. Man, and W. K. Chong, "A recommendation model using the bandwagon effect for e-marketing purposes in iot," *International Journal of Distributed Sensor Networks*, vol. 2015, p. 5, 2015.

[4] M. Balabanović and Y. Shoham, "Fab: content-based, collaborative recommendation," *Communications of the ACM*, vol. 40, no. 3, pp. 66–72, 1997.

[5] K. Lang, "Newsweeder: Learning to filter netnews," in *In Proceedings of the Twelfth International Conference on Machine Learning*. Citeseer, 1995, pp. 331–339.

[6] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "Grouplens: an open architecture for collaborative filtering of netnews," in *Proceedings of the 1994 ACM conference on Computer supported cooperative work*. ACM, 1994, pp. 175–186.

[7] J. S. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," in *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 1998, pp. 43–52.

[8] A. Forestiero, C. Mastroianni, and G. Spezzano, "A multi agent approach for the construction of a peer-to-peer information system in grids," *Self-Organization and Autonomic Informatics (I)*, vol. 135, pp. 220–236, 2005.

[9] A. Crespo and H. Garcia-Molina, "Routing indices for peer-to-peer systems," in *Distributed Computing Systems, 2002. Proceedings. 22nd International Conference on*. IEEE, 2002, pp. 23–32.

[10] A. Forestiero, C. Mastroianni, G. Papuzzo, and G. Spezzano, "A proximity-based self-organizing framework for service composition and discovery," in *Cluster, Cloud and Grid Computing (CCGrid), 2010 10th IEEE/ACM International Conference on*. IEEE, 2010, pp. 428–437.

[11] M. Cai, M. Frank, J. Chen, and P. Szekely, "Maan: A multi-attribute addressable network for grid information services," *Journal of Grid Computing*, vol. 2, no. 1, pp. 3–14, 2004.

[12] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm intelligence: from natural to artificial systems*. Oxford university press New York, 1999, vol. 4.