A Semantic-Based Algorithm for Data Dissemination in Opportunistic Networks*

Marco Conti, Matteo Mordacchini,
Andrea Passarella, and Liudmila Rozanova

IIT – CNR, Pisa, Italy
{firstname.lastname}@iit.cnr.it

Abstract. The opportunistic data dissemination problem for mobile devices is an open topic that has attracted many investigations so far. At the best of our knowledge, none of these approaches takes into account the semantic side of the data shared in an opportunistic network. In this paper, we present an algorithm that, starting from the semantic data annotations given by the users themselves, builds a semantic network representation of the information. Exploiting this description, we detail how two different semantic networks can interact upon contact, in order to spread and receive useful information. In order to provide a performance evaluation of such a solution, we show a preliminary set of results obtained in a simulated scenario.

1 Introduction

The increasing, pervasive presence of devices interacting among themselves and their users is leading to a complex and vast information environment, where information flows from the physical world to the cyber one, and vice-versa. Users mobile devices, sensor networks, and all the devices spread in the environment with data generation capabilities (e.g., in Internet of Things applications) will constantly generate huge amounts of data thus generating a very rich information landscape. This scenario is known as the *Cyber–Physical World* (CPW) convergence [1]. Mobile devices will act in the CPW convergence scenario as *proxies* of their human users. They will be in charge of discovering and evaluating the relevance for their human users of the huge amount of information available in the cyber world. This has to be done quickly and using limited resources, as devices will be constantly face large amounts of data items. This situation resemble what the *human brain* does when it has to assess the relevance of the information coming from the surrounding environment. Since devices will act as the *avatars* of their owners, considering the way the human brain deals with huge amounts

* This work is funded by the EC under the RECOGNITION (FP7-IST 257756) and EINS (FP7-FIRE 288021) projects. This work was carried out during the tenure of L. Rozanova's ERCIM "Alain Bensoussan" Fellowship Programme. This Programme is supported by the Marie Curie Co-funding of Regional, National and International Programmes (COFUND) of the European Commission.

W. Elmenreich, F. Dressler, and V. Loreto (Eds.): IWSOS 2013, LNCS 8221, pp. 14–26, 2014.

of data in a short time is a sensible point for designing effective and efficient information dissemination schemes in the CPW convergence scenario .

Opportunistic networking [2] is one of the key paradigms to support direct communication between devices in scenarios like the CPW convergence. In this paradigm, nodes are mobile, and forwarding of messages occurs based on the store, carry and forward concept. In this paper we present a data dissemination algorithm for opportunistic networks inspired by real human communication schemes. It exploits the *semantic* representation of data (e.g., tags and other metadata associated to data items) in order to assess the relevance of information to be exchanged among mobile nodes upon contact. A dissemination scheme for spreading the actual content associated with semantic data can be built on top of such an algorithm, as we point out later in Sec. 3. The focus of this paper is to define and give an initial performance evaluation of a semantic data dissemination mechanism, in order to study the viability of this approach before exploiting it to design effective and efficient content dissemination schemes in opportunistic networks. In this proposal, the data stored in a device is semantically represented using an approach based on semantic networks. A semantic network is a graph, where vertices are the semantic concepts (e.g., the tags associated to data items) and edges represent the connections that exist among them (e.g., the logical link between two tags). We describe how semantic networks can be used to determine the relevance of information to be exchanged upon physical contact among nodes. Essentially, similar to a real human communication, the selection of information starts from a set of semantic concepts in common between the two nodes. The relevance for one node of the rest of the information available in the other node's semantic network can then be assessed based on the links between the semantic concepts in the semantic network. Similarly to a human dialogue, information logically closer to a set of common concepts is exchanged first, and the longer the discussion (i.e. the duration of the physical contact among nodes), the greater the amount of information exchanged. Once an encounter finishes, new semantic concepts are passed from one node to the another, and new connections among new and old concepts can be established, thus increasing the knowledge of each node.

The rest of this paper is organized as follows. In Sec. 2 we review the relevant literature for the opportunistic data dissemination problem. In Sec. 3 we give a general overview of the concepts behind the solution we propose, while in Sec. 3.1, Sec. 3.2 and Sec. 3.3 we describe how semantic networks can be constructed and interact. Sec. 4 presents some preliminary simulation results obtained with this solution. Finally, Sec. 5 concludes the paper.

2 Related Work

The data dissemination problem in opportunistic networks has been faced by many solutions in literature. PodNet [3] is one of the first works on this subject. The authors of PodNet propose four different strategies to weight the relevance of data to be exchanged on the basis of the estimated popularity of the general topic

(*channel*) the data belongs to. More refined approaches try to take advantage of the information about users' social relationships to drive the dissemination process. For instance, in [4], the authors propose to build a pub/sub overlay in an opportunistic network. The most "socially-connected" nodes, i.e., those nodes that are expected to be most available and easily reachable in the network, take the role of brokers, as in more traditional pub/sub systems. They are in charge to collect information from their social groups, spread this data among them, and eventually deliver it toward interested peers. Also the authors of ContentPlace [5] propose to drive the data dissemination process using the social structure of the network of users. Specifically, each node tries to fetch from other encountered peers those data items that are likely of interest to other users it has social relationships with (and which, therefore, are expected to be in touch with them in the near future). In [6,7], the authors define a data dissemination scheme that directly embodies and exploits the very same cognitive rules (*cognitive heuristics*) used by the human brain to assert the relevance of information. This solution proves to be as effective as another scheme like ContentPlace in disseminating the information, while requiring much less overhead.

With respect to all these approaches, in this paper we take a different direction. None of the approaches above take into consideration the semantic dimension of data. In the following we define how a semantic representation of information can be used to determine the relevance of information to be exchange in an opportunistic network scenario, and we validate this proposal with preliminary simulation results.

3 Data Dissemination Using Semantic Networks

In order to semantically represent the information carried by each user, we exploit an approach based on semantic networks. A semantic network is a graph where vertices are semantic concepts and edges connect semantically related concepts. In order to derive this representation, we consider that each user has a collection of data elements, like pictures, blog posts, status updates, geospatial data, etc. Some of these data items could be difficult to analyze semantically. Anyway, we could take advantage of the fact that many of these data items are usually associated with tags, as happens in many real-world social networks like Twitter, Flickr, Instagram, etc. Thus, a simple and, at the same time, effective way of creating a semantic network is to use data tagging, as depicted in Fig. 1. Tags can then become the nodes of a user semantic network, while edges can be derived from the associations the user has done when tagging their own data.

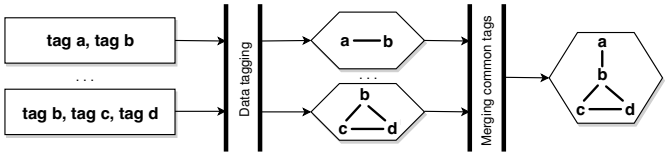


Fig. 1. Creation of a Semantic Network by data tagging

Note that not all the information belonging to a user may by represented in the user's semantic network. It seems reasonable that only information that is assumed to be relevant for the user at a given time is mapped in her semantic network. In order to select this information, we can act in a similar way like the human brain does, exploting an approach built on models of the human forgetting process. Thus, we can assume that information that has not been used for a long time is forgotten, while information that is frequently accessed is harder to forget. Rather than a limit, forgetting is a mechanism that could aid some human cognitive processes (e.g. [8]) by letting them keep and consider only the most relevant data. To model a forgetting process into our system, we propose to introduce a special "forgetting" function similar to an experimental curve that has been first obtained by H. Ebbinghaus in 1885 [9] and then confirmed and refined by numerous subsequent studies (Fig. 2). This curve reflects the exponential decline of individual human memory retention with time. The rate of forgetting depends on the repetition rate of incoming information.

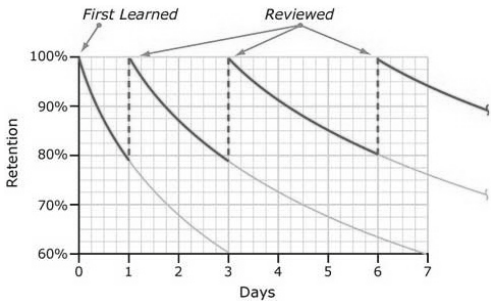


Fig. 2. Example of a forgetting curve

Given the description above, the semantic network of each user is dynamic; it is constructed from the individual information by tagging, taking into account the relationships between tags defined by the user herself, and frequency and/or duration of accessing them.

Using this description, in the following sections we first describe how each device can build its own semantic network from the data it owns. Next, we exploit the semantic network associated to data items in order to define a data dissemination scheme in an opportunistic network scenario, where information is exchanged only upon physical contact between nodes. The algorithm we describe deals with semantic data only. Anyway, it can be easily used as the basis for designing the dissemination of the actual content the semantic data is related to, as we stated in the Introduction. The semantic data exchanged with another node can be used to sort the content items in order to select which of them can be transferred to the encountered peer. For istance, a simple way to do this can be to give precedence to those items with the largest intersection between the semantic concepts associated to them and the set of semantic concepts that are going to be exchanged with the other peer. In this paper, we describe how

semantic data can be spread in an opportunistc network scenario, leaving the design of an effective and efficient content dissemination mechanism based on this scheme as a future research direction.

3.1 Semantic Network Creation

We define the semantic network of each user as a *dynamic weighted graph* $G = \{V, E, f(e,t)\} : t \in T$, where t is the time, V is the set of vertices in the graph, E is the set of edges and $f(e,t)$ is a weighting function for each $e \in E$ that reproduces the human forgetting process in our system. In addition, each edge e_{ij} between two vertices i and j has an associated "popularity" p_{ij}^t that measures the number of times e_{ij} was present in exchanges with other nodes' semantic networks in the encounters happened till time t.

In the context of an active user participation to the creation of content, we assume that each data item owned by a user is associated with a set of tags, defined by the user herself. In the graph G that represents the user's semantic network, tags are the vertices, while edges connecting tags are created using the following strategy. Firstly, for each data item, its tags are linked together in order to form a completely connected component. Then, each set of vertices carrying the same label (i.e. they where created from tags having the same name) is considered. These are vertices belonging to different components and they are merged together, forming a single vertex with their same label. This single vertex inherits all the edges pointing to the original vertices in their respective components.

As an example of this process, consider the example given in Fig. 3. The user has two different pictures and their associated tags. For the two pictures, two completely connected components are created. Then, the components are merged using the common vertex "lake" as the pivot of this process.

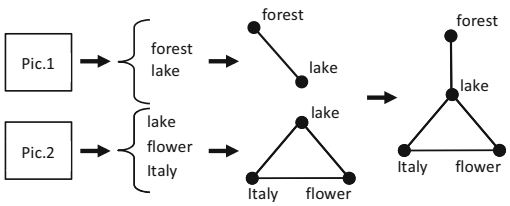


Fig. 3. Creation process of a user Semantic Network

3.2 Forgetting Mechanism

Let us now define the forgetting function $f(e,t)$ as a function that is able to assign a weight to any edge e at a time t. If $f(e,t) \leq f_{min}$, where f_{min} is a limiting threshold value, then the edge e does not exist in the semantic network at time t. Initially, at time $t_0 = 0$, for any $e \in E$, we have that: $f(e,t_0) = 1$. Subsequently, for any edge e_{ij} and time $t > t_0$, in each interval (t^*, t) , where

t^* is the last time this edge was used in exchanges with other peers (i.e. its last "activation"):

$$f(e_{ij}, t) = \alpha e^{-\beta_{ij}(t-t^*)}$$

where α is a normalizing coefficient and β_{ij} is the "speed of forgetting", i.e. the weakening of the connection (taken in accordance with the experimental curve obtained by H. Ebbinghaus [9]). Obviously, β_{ij} depends on the total number of previous connections. Then the "popular" connections are "being forgotten" more slowly in the situation when there are no subsequent connections. So, we can define this parameter as follows: $\beta_{ij} = \frac{\beta}{p^t_{ij}}$, where β is a speed coefficient and p^t_{ij} is the "popularity" of e_{ij} at time t, i.e. the number of times e_{ij} has been used in the encounters happened before t. Fig. 4 shows one of the side effects of the forgetting process. Not only edges, but even vertices can be forgotten. In fact, if the deletion of an edge e implies that a vertex v at one of e's endpoints is no longer connected to any other vertex in the network, then v is also deleted from the semantic network. After that, in order for it to reappear in the semantic network, it should be received during successive encounters with other nodes, using the data exchange scheme detailed in the following section.

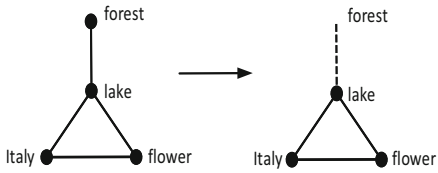


Fig. 4. Effects of the forgetting process

3.3 Interaction between Semantic Networks

In a real human communication, a dialogue begins with some concepts that are in common between both parties. Similarly to this behaviour, we let the interaction between two semantic networks start with one or few key concepts (vertices) that belong to both semantic networks. Starting from these key vertices, each device is able to compute which are the vertices and edges from its own semantic network that are to be communicated and transferred to the other party. Precisely, looking from the viewpoint of a device, let its semantic network $G = (V, E, f(e, t))$ be the *donor network*, while the other party semantic network $G' = (V', E', f'(e', t))$ is termed the *recipient network*. During the interaction between these two networks, concepts of the donor network are included in the recipient network and new connections between concepts are formed. As a result we obtain a *new (updated) recipient* semantic network $G^* = (V^*, E^*, f^*(e^*, t))$. Being $|V| = n$ and $|V'| = m$, we have that $|V^*| = l \leq n + m$.

In order to determine which vertices and edges from the donor network are exchanged during a communication, a node computes a *contributed network* $C = (\bar{V}, \bar{E}, \bar{f}(\bar{e}, t))$, which will contain the data that will be transmitted. Once the contributed network is received, it is merged with the recipient network in order to

create the updated recipient network. In the following, we describe how a node computes the contributed network and how this is finally merged with the recipient network. Hereafter, when we say that an edge is included in the contributed network, we imply that the vertices at both the endpoints of that edge are also included, in case the contributed network does not already contain them.

Supposing that the interaction starts at time t, the pseudo-code used for initializing the contributed network is presented in Alg.1. As already stated, we assume that a data exchange between two nodes starts from a set of shared semantic concepts, i.e. a set of vertices $K = \{v_k : v_k \in V \cap V'\}$ (lines 1–6 of the pseudo-code). Note that in case $K = \emptyset$, nothing is passed from one node to the other. Vertices and the corresponding edges that connect them directly (one-hop distance) with key nodes are included instantly in the contributed network (lines 7–12). This process is analogue to the idea of *"gestalt"*, when the understanding is not limited to a single concept, but brings a system of linked concepts i.e. a part of the semantic network. Note that the weights of edges included in the contributed network are set to the maximum in both the donor and contributed networks (lines 10–11), as the exchange of this data leads to a sort of "activation" of the corresponding edge in memory, thus inducing the forgetting process to "restart". In order to compute which of the remaining vertices and edges of the donor network should be included in the contributed network, we proceed as detailed in Alg. 2. Edges will be subject to a "warming" process, that will mainly depend on the duration of the contact and the proximity of an edge's endpoints to a key vertex. Thus, vertices and edges will be included in the contributed network by levels, as shown in Fig. 5. The proximity value is computed as the minimum number of hops needed to the reach any of the key vertices in the donor network. Edges that connect vertices that are closer to a key node will be "warmed up" faster than those linking vertices located far away. Moreover, the longer will last an interaction between two nodes, the easier an edge will be "warmed". This process mimics what happens in a human communication process, where, starting from a set of common concepts, other semantically connected notions could be included in the dialogue. The longer the discussion

Algorithm 1. Contributed Network Creation at time t

1: Let $G = (V, E, f(e, t))$ be the donor network;
2: Let $C = (\bar{V}, \bar{E}, \bar{f}(\bar{e}, t))$ be the contributed network;
3: Let K be the set of *key vertices*, $K \subseteq V$
4: **for** each $k \in K$ **do**
5: $\quad \bar{V} \cup = k$
6: **end for**
7: **for** each $v \in V$ and each $k \in K$ such that $\exists e_{kv} \in E$ **do**
8: $\quad \bar{V} \cup = v$
9: $\quad \bar{E} \cup = e_{kv}$
10: $\quad$ Set $f(e_{kv}, t) = 1$ in G
11: $\quad$ Set $\bar{f}(e_{kv}, t) = 1$ in C
12: **end for**

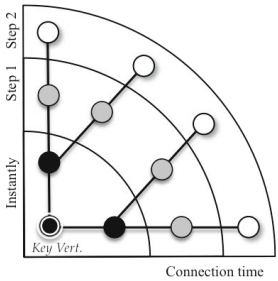


Fig. 5. Selection of vertices from the donor network using proximity levels

takes, the more concepts are exchanged. When the interaction is terminated, the contributed network has only those edges (and related vertices) that exceed a "warm" activation threshold. Thus, only that information is transferred to the recipient network.

In detail, supposing that a connection starts at time t and ends at time t^*, edge warming is computed using the following formula:

$$w(e_{ij}, \Delta_t) = \frac{\gamma_{step}}{1 + e^{-p_{ij}^{t^*}}}(1 - e^{-\tau \Delta_t})$$

where $e_{ij} \in E$ is an edge of the donor network, Δ_t is the duration of the connection, i.e. $\Delta_t = t^* - t$, $p_{ij}^{t^*}$ is the popularity of e_{ij} at time t^* and τ is a normalizing factor. The coefficient γ_{step} is used to weight the proximity of e_{ij} to any key vertex. We can define this value as $\gamma_{step} = \frac{\gamma}{n}$, where n is the the number of hops in the shortest path to the nearest key vertex and γ is a normalizing factor. Being w_{min} the minimum warm threshold, the contributed network will contain an edge e_{ij} **iff** $w(e_{ij}, \Delta_t) \geq w_{min}$ (lines 10–17). Moreover, in order to limit the amount of information exchanged during an encounter, we consider that, apart from its warm weight, an edge e_{ij} is included in the contributed network **iff** it is within h hops from any key vertex in the donor network (line 8). At the end of the interaction, the contributed network is transferred from the donor node to the recipient one. The contributed network is merged with the recipient network using Alg. 3. Edges and vertices that do not exist in the recipient network are added (lines 5–14). In case an edge of the contributed network already exists in the recipient one (line 16), its weight is set to the maximum between the weigth already assigned to it in the recipient network and the weight passed by the contributed network.

4 Simulation Results

In order to evaluate the proposed algorithm, we simulated its behaviour in the following scenario. We consider 99 mobile nodes that move in a 1000mx1000m area. In order to simulate real user movement patterns, nodes move according to the HCMM model [10]. This is a mobility model that integrates temporal,

Algorithm 2. Contributed Network computation at the end of an encounter

Let $G = (V, E, f(e, t))$ be the donor network;
Let $C = (\bar{V}, \bar{E}, \bar{f}(\bar{e}, t))$ be the already initialized contributed network;
Let K be the set of *key vertices*, $K \subseteq V$
Let $A = (V \cap \bar{V}) - K$
Let h be the depth limit
Let w_{min} be the weight threshold
Let $depth = 2$
while $depth \leq h$ **do**
  Let $B = \emptyset$
  for each $v \in (V - \bar{V})$ such that $\exists e_{av} \in E, a \in A$ **do**
    if $w(e_{av}, t^*) \geq w_{min}$ **then**
      $\bar{V} \cup = v$
      $\bar{E} \cup = e_{a,v}$
      Set $f(e_{av}, t) = 1$ in G
      Set $\bar{f}(e_{av}, t) = w(e_{av}, \Delta_t)$ in C
      $B \cup = v$
    end if
  end for
  $A = B$
  $depth = depth + 1$
end while

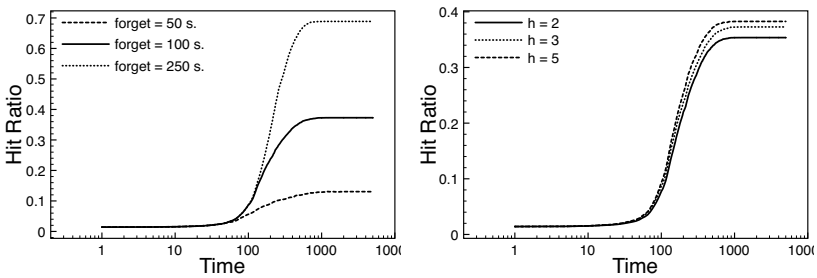
Algorithm 3. Merging of contributed and recipient networks

Let $G' = (V', E', f'(e', t)$ be the recipient network;
Let $C = (\bar{V}, \bar{E}, \bar{f}(\bar{e}, t))$ be the contributed network;
Let $G^* = (V^*, E^*, f^*(e^*, t))$ be the updated recipient network;
$G^* = G'$;
for each $\bar{e}_{ij} \in \bar{E}$ **do**
  if $\bar{e}_{ij} \notin E^*$ **then**
    if $\bar{v}_i \notin V^*$ **then**
      $V^* \cup = \bar{v}_i$
    end if
    if $\bar{v}_j \notin V^*$ **then**
      $V^* \cup = \bar{v}_j$
    end if
    $E^* \cup = \bar{e}_{ij}$
    Set $f^*(\bar{e}_{ij}, t) = \bar{f}(\bar{e}_{ij}, t)$
  else
    Set $f^*(\bar{e}_{ij}, t) = \max(f^*(\bar{e}_{ij}, t), \bar{f}(\bar{e}_{ij}, t))$;
  end if
end for

social and spatial notions in order to obtain an accurate representation of real user movements. Specifically, in HCMM the simulation space is divided in cells representing different social communities. Nodes move between social communities, and nodes movements are driven by social links between them. In this preliminary study, we consider that there exists only one social community, i.e.

the simulation space consists of one cell that covers all the simulation area. Data assigned to these nodes is selected from the CoPhIR dataset [11]. This is a dataset containing more than 100 million images taken from Flickr. Along with other data, for each image it is possible to know the user that generated it and the associated tags. In order to create a useful dataset to test our solution, we selected images with at least 5 tags each. This number was chosen considering that the overall mean number of tags per image in the dataset is 5.02. Then, we extracted those users that have at least 10 such images in their collections. Finally, from this set of users, we randomly chose the 99 users that we used in the simulation. For each of these users, a corresponding semantic network is created, according to the description given in Sec. 3.1. We then study the transient state of the interaction between these users, by repeating 10 different tests obtained by producing 10 different mobility traces using the HCMM model. Results reported in the following simulations are the average of all the performed tests. Each simulation experiment runs for 5000 sec.

Fig. 6(a) (log-scale on the x axis) shows the evolution over time of the tags Hit Ratio for three different settings of the forget function. The overall Hit Ratio is defined as the mean of the per tag hit ratio. This latter quantity is computed as the mean number of nodes having a given tag in their semantic networks at a given time. We set the parameters of the forget function in order to have the less popular edges be deleted after 50, 100 and 250 sec of inactivity, respectively. In this scenario, vertices at more than 3 hops from a key vertex are not exchanged during contacts. The first thing to note is that tags are not spread to any node in all the cases we present. This is consistent with the view that each user is probably not interested in every semantic concept available. Rather, she is more interested in fetching the concepts more pertinent with her own knowledge. Anyway, as one could expect, the forget function plays a key role in the data dissemination process. The longer we let the edges (and related vertices) stay in their semantic networks before being forgotten, the higher will be the final diffusion of the corresponding tags in the network. Since there are great variations in the Hit Ratio values, a proper tuning of the forget function parameters implies

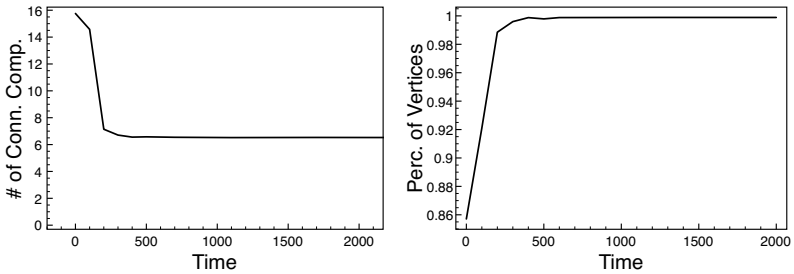


(a) (b)

Fig. 6. Hit Ratios for different settings of (a) the forget function and (b) parameter h

a trade-off between the willingness to let the concepts permeate the network and the need to limit the associated resource consumption (storage, bandwidth, etc.). In all the reported results, the Hit Ratio reaches a stabilization value. This is due to the fact that (i) useful information has been already shared among potentially interested nodes; (ii) the meeting rate of this data is faster than the forget process, i.e. data is seen again before the associated forget function value falls below the forgetting threshold. In Fig. 6(b) we report the impact on the Hit Ratio of the proximity level limit h for the exchange of vertices and related edges. In this case we fixed the forget function parameters in order to let the least popular edges disappear after 100 sec. of inactivity. Results are obtained for $h = 2, 3$ and 5. We can see that allowing to include in the contributed networks a larger (i.e. more distant from key vertices) portion of the donor networks result in a larger diffusion of semantic concepts (i.e. an higher Hit Ratio). Anyway, the Hit Ratio is less sensitive to changing in the h value rather than to changes in the forget function parameters. Thus, although different values of h lead to different Hit Ratios, tuning of this parameter is less critical than that of the forget function, since it leads to relatively small differences in the final Hit Ratio values.

In the next sets of results, we study some of the general properties of the semantic networks as they result at the end of the simulation. Main parameters are: $h = 3$; the forget function deletes the least popular edges after 100 sec. The left side of Fig. 7 shows the evolution over time of the mean number of different connected components that form each semantic network. Each semantic network is not a complete graph. Rather it is a set of different weakly connected components. On one hand, these different sets of semantic concepts (i.e. vertices) may be thought to represent different, semantically uncorrelated groups of topics the user is interested in. Anyway, these sets could also be disconnected one from the others since the user lacks the knowledge needed to put them together. We can see that, as time passes, the mean number of different connected components rapidly falls down, as an effect of the data dissemination process. Since new vertices and new connections between old and new vertices are created, some sets of previously disconnected nodes start to merge, until the process stabilizes. Moreover, from the right side of Fig. 7, we can deduce that, on average, the biggest connected component of each semantic network acts as an attractor of other previously disconnected components. In fact, in parallel with the reduction of the number of disconnected components, the average relative size of the biggest connected component rapidly increase. Once the number of disconnected components stabilizes, the size of the biggest connected component comprises almost all the vertices of a semantic network. Indeed, less then 1% of all the vertices are in the other components. Finally, Fig. 8 plots the degree distribution at the beginning and end of the simulation. The figures uses a log-scale on the y axis only, in order to let the differences between the two distributions be more visible. It is possible to note that, at the end of the simulation, the algorithm preserves the same slope of the nodes degree distribution that was present before the simulation starts. Anyway, there is an increased probability



(a) (b)

Fig. 7. Evolution of (a) the number of distinct connected components and (b) the size of the biggest connected component

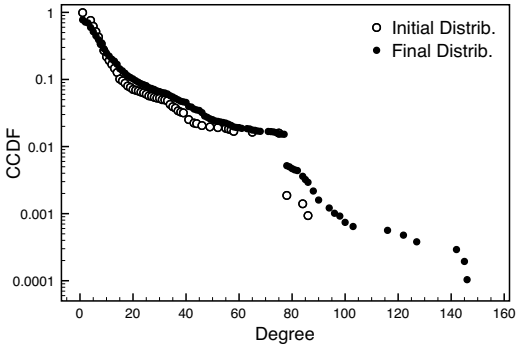


Fig. 8. Nodes' degree distribution before and at the end of the simulation

to find nodes with high degree, as shown by the CCDF of the final degree distribution. Node with an intial higher degree have more chances to be involved in an exchange than nodes with lower degrees. Moreover, the forget process cuts less popular edges, thus further reducing the degree of less spread vertices. Edges attached to high-degree nodes take advantage of the nodes' probability to be exchanged in order to avoid to be forgotten. Eventually, this mechanism favours the increase of the degree of already well-connected nodes.

5 Conclusions

The semantic information associated with data items could be a powerful tool in a data dissemination scheme in order to assert the relevance and relationship of already owned information and newly discovered knowledge. Exploiting the data semantic tagging done by the users themselves, we defined a semantic-based data dissemination algorithm for opportunistic networks. We show how each device is able to give a semantic network representation of its own data and how this

representation can be used to select the information to be exchanged by users upon physical contact. In a first set of preliminary simulation results based on this approach, we studied the impact of various parameters on both the data dissemination process and the evolution and final properties of the users' semantic networks. Future research directions encompass the definition of a content dissemination scheme based on this solution, an even more formal mathematical description of this proposal, a more comprehensive study via simulation of the performances and properties of the algorithm and its application to different scenarios, where other factors, like social relationships among users, should be taken into account.

References

1. Conti, M., Das, S.K., Bisdikian, C., Kumar, M., Ni, L.M., Passarella, A., Roussos, G., Trster, G., Tsudik, G., Zambonelli, F.: Looking ahead in pervasive computing: Challenges and opportunities in the era of cyberphysical convergence. Pervasive and Mobile Computing 8(1), 2–21 (2012)
2. Pelusi, L., Passarella, A., Conti, M.: Opportunistic networking: data forwarding in disconnected mobile ad hoc networks. IEEE Communications Magazine 44(11), 134–141 (2006)
3. Lenders, V., May, M., Karlsson, G., Wacha, C.: Wireless ad hoc podcasting. SIGMOBILE Mob. Comput. Commun. Rev. 12, 65–67 (2008)
4. Yoneki, E., Hui, P., Chan, S., Crowcroft, J.: A socio-aware overlay for publish/subscribe communication in delay tolerant networks. In: MSWiM, pp. 225–234 (2007)
5. Boldrini, C., Conti, M., Passarella, A.: Design and performance evaluation of contentplace, a social-aware data dissemination system for opportunistic networks. Comput. Netw. 54, 589–604 (2010)
6. Conti, M., Mordacchini, M., Passarella, A.: Data dissemination in opportunistic networks using cognitive heuristics. In: The Fifth IEEE WoWMoM Workshop on Autonomic and Opportunistic Communications (AOC 2011), pp. 1–6. IEEE (2011)
7. Bruno, R., Conti, M., Mordacchini, M., Passarella, A.: An analytical model for content dissemination in opportunistic networks using cognitive heuristics. In: Proceedings of the 15th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, MSWiM 2012, pp. 61–68. ACM, New York (2012)
8. Schooler, L., Hertwig, R.: How forgetting aids heuristic inference. Psychological Review 112(3), 610 (2005)
9. Ebbinghaus, H.: Memory: A contribution to experimental psychology, vol. 3. Teachers college, Columbia university (1913)
10. Boldrini, C., Passarella, A.: Hcmm: Modelling spatial and temporal properties of human mobility driven by users' social relationships. Comput. Commun. 33, 1056–1074 (2010)
11. Bolettieri, P., Esuli, A., Falchi, F., Lucchese, C., Perego, R., Piccioli, T., Rabitti, F.: CoPhIR: A test collection for content-based image retrieval. CoRR abs/0905.4627v2 (2009)