Computer Communications 73 (2016) 200–210

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

Endorsement deduction and ranking in social networks

Hebert Pérez-Rosés [a,b,*], Francesc Sebé [a], Josep Maria Ribó [c,†]

[a] Department of Mathematics, Universitat de Lleida, Spain
[b] University of Newcastle, Australia
[c] Department of Computer Science and Industrial Engineering, Universitat de Lleida, Spain

ARTICLE INFO

Article history:
Available online 8 September 2015

Keywords:
Expertise retrieval
Social networks
LinkedIn
ResearchGate
PageRank

ABSTRACT

Some social networks, such as LinkedIn and ResearchGate, allow user endorsements for specific skills. In this way, for each skill we get a directed graph where the nodes correspond to users' profiles and the arcs represent endorsement relations. From the number and quality of the endorsements received, an authority score can be assigned to each profile. In this paper we propose an authority score computation method that takes into account the relations existing among different skills. Our method is based on enriching the information contained in the digraph of endorsements corresponding to a specific skill, and then applying a ranking method admitting weighted digraphs, such as PageRank. We describe the method, and test it on a synthetic network of 1493 nodes, fitted with endorsements.

1. Introduction

Directed graphs (digraphs) are an appropriate tool for modelling social networks with asymmetric binary relations. For instance, the blogosphere is a social network composed of blogs/bloggers and the directed 'recommendation' or 'follower' relations among them. Other examples include 'trust' statements in recommendation systems (some user states that he/she trusts the recommendations given by some other user) and 'endorsements' in professional social networks. Additionally, weighted arcs appear in situations where such relations can accommodate some degree of confidence ('trust' or 'endorsement' statements could be partial).

LinkedIn and ResearchGate are two prominent examples of professional social networks implementing the *endorsement* feature. LinkedIn[1] is a wide-scope professional network launched in 2003. More than a decade later it boasts a membership of over 364 million, and it has become an essential tool in professional networking. The LinkedIn endorsement feature, introduced about three years ago,[2] allows a user to endorse other users for specific skills.

On the other hand, ResearchGate[3] is a smaller network catering to scientists and academics. It was launched in 2008, and it reached five million members in August, 2014. ResearchGate also introduced an endorsement feature recently.[4] From the endorsements shown in an applicant's profile, a potential employer can assess the applicant's skills with a higher level of confidence than say, by just looking at his/her CV.

The two endorsement systems described above are very similar: for each particular skill, the endorsements make up the arcs of a directed graph, whose vertices are the members' profiles. In principle, these endorsement digraphs could be used to compute an authority ranking of the members with respect to each particular skill. This authority ranking may provide a better assessment of a person's profile, and it could become the basis for several social network applications.

For instance, this authority ranking could be the core element of an eventual tool for finding people who are proficient in a certain skill, very much like a web search engine. It could also find important applications in profile personalization. For example, if a certain user is an expert in some field, say 'Operations Research', the system can display ads, job openings, and conference announcements related to that field in the user's profile. Finally, we can envisage a world where people could vote on certain decisions via social networks. For example, a community of web developers could decide on the adoption of some particular web standard. In that scenario, we might think about a *weighted voting scheme*, where the weight of each vote is proportional to the person's expertise in that area.

Now, people usually have more than one skill, with some of those skills being related. For example, the skill 'Java' is a particular case of the skill 'Programming', which in turn is strongly related with the

* Corresponding author at: Department of Mathematics, Universitat de Lleida, Spain. Tel.: +34 973 702 781.
E-mail address: hebert.perez@matematica.udl.cat, Hebert.Perez@gmail.com (H. Pérez-Rosés).
† Deceased.
[1] http://www.linkedin.com.
[2] More precisely, on September 24, 2012.
[3] http://www.researchgate.net.
[4] On February 7, 2013.

http://dx.doi.org/10.1016/j.comcom.2015.08.018
0140-3664/

skill 'Algorithms'. It may well happen that a person is not endorsed for the skill 'Programming', but he/she is endorsed for the skills 'Java' and 'Algorithms'. From those endorsements it can be deduced with a fair degree of confidence that the person also possesses the skill 'Programming'. In other words, a person's ranking with respect to the skills 'Java' and 'Algorithms' affects his/her ranking with respect to the skill 'Programming'.

If the members of a social network were consistent while endorsing their peers, this 'endorsement with deduction' would not add anything to simple (i.e. ordinary) endorsement. In this ideal world, if Anna endorses Ben for the skill 'Java', she would be careful to endorse him for the skill 'Programming' as well.[5] In practice, however,

1. People are not systematic. That is, people do not usually go over all their contacts methodically to endorse, for each contact and alleged skill, all those contacts which, according to their opinion, deserve such endorsement. This may be the source of important omissions in members' profiles.
2. People are not consistent, for consistency, like method, would require a great effort. In an analysis of a small LinkedIn community consisting of 3250 members we have detected several inconsistencies. For example, there are several users who have been endorsed for some specific programming language, or a combination of programming languages, but have not been endorsed for the skill 'Programming'. Deciding whether there is an inconsistency entails some degree of subjectivism, for inconsistencies ultimately depend on the semantics of the skill names. Nevertheless, we can safely assert that practically 100% of the profiles sampled by us contained some evident inconsistency or omission. The Appendix lists some of the more significant inconsistencies and omissions encountered, together with a more comprehensive discussion about LinkedIn's endorsement mechanism.
3. Skills lack standardization. In most of these social networks, a set of standard, allowed skills has not been defined. As a result, many related skills (in many cases, almost synonyms) may come up in different profiles of the social network. Consider, for example, skills such as 'recruiting', 'recrutments', 'IT recruiting', 'internet recruiting', 'college recruiting', 'student recruiting', 'graduate recruiting', etc. which are, all of them, common in LinkedIn profiles. It may well happen that an expert in 'recruiting' has not even assigned to him/herself that specific skill, but a related one such as 'recruitments', which would hide him/her as an expert in the 'recruiting' skill.

Endorsement with deduction may help address those problems, and thus provide a better assessment of a person's skills. More precisely, we propose an algorithm that enriches the digraph of endorsements associated to a particular skill with new weighted arcs, taking into account the correlations between that 'target' skill and the other ones. Once this has been done, it is possible to apply different ranking algorithms to this enhanced digraph with the purpose of obtaining a ranking of the social network members concerning that specific skill.

1.1. Related work

This research can be inscribed into the discipline of *expertise retrieval*, a sub-field of information retrieval [1]. There are two main problems in expertise retrieval:

1. Expert finding: attempts to answer the question "*Who are the experts on topic X?*". In our approach, this question is answered by taking all the network members who are within a certain percentile of the ranking for topic *X*.
2. Expert profiling: addresses the question "*Which skills does person Y possess?*". We could answer this question by computing the rankings with respect to all the skills claimed by person *Y*, and taking those skills for which *Y* has fallen within the pre-defined percentile mentioned above.

Traditionally, these problems above have been solved via document mining, i.e. by looking for the papers on topic *X* written by person *Y*, combined with centrality or bibliographic measures, such as the H-index and the G-index, in order to assess the expert's relative influence (e.g. [29]). This is also the approach followed by ArnetMiner,[6] a popular web-based platform for expertise retrieval [45].

Despite their unquestionable usefulness, systems based on document mining, like ArnetMiner, face formidable challenges that limit their effectiveness. In addition to the specific challenges mentioned by Hashemi et al. [20], we could add several problems common to all data mining applications (e.g. name disambiguation). As a small experiment, we have searched for some known names in ArnetMiner, and we get several profiles corresponding to the same person, one for each different spelling.

That is one of the reasons why other expertise retrieval models resort to the power of PageRank in certain social networks, such as in the perused scientific citation and scientific collaboration networks (e.g. [10,20]). Another interesting example related to PageRank and social networks is TwitterRank [48], an extension of PageRank that measures the relative influence of Twitter users in a certain topic. Like our own PageRank extension, TwitterRank is topic-specific: the random surfer jumps from one user to an acquaintance following topic-dependent probabilities. However, TwitterRank does not consider any relationships among the different topics.

To the best of our knowledge, there are no precedents for the use of endorsements in social networks, nor for the use of known relationships among different skills, in the context of expertise retrieval. The closest approach might be perhaps the one in [41], which uses the ACM classification system as an ontology that guides the mining process and expert profiling. Another (very recent) model that uses semantic relationships to increase the effectiveness and efficiency of the search is given in [27].

Another related field which has attained a growing interest in the last few years is that of reputation systems, that is, systems intended to rank the agents of a domain based on others' agents reports. Strategies for ranking agents in a reputation system range from a direct ranking by agents (as used in eBay) to more sophisticated approaches (see [30] for a survey). One particularly important family of reputation system strategies is that of PageRank-based algorithms. There are many of such approaches. For instance, [8] provides an algorithm based on the so-called Dirichlet PageRank, which addresses problems such as: (1) some links in the network may indicate distrust rather than trust, and (2) how to infer a ranking for a node based on the ranking stated for a well-known subnetwork.

Another example of reputation system (again, based on PageRank) is one explained in [40]. In this case, a modification of the PageRank algorithm is used to create a reputation ranking among the members of an academic community. One remarkable issue of this approach is that the network does not exist explicitly, but it is created ad-hoc from the information harvested from the personal web pages of the members (e.g. a couple of members are connected if they have authored a research article together).

A thorough study of reputation systems is clearly beyond the scope of this article, but in any case, all these scenarios above differ significantly from our application for expertise retrieval with deduction of new endorsements, based on existing endorsements of related skills, and information about the correlation between skills.

[5] Some people may argue that knowledge of a programming language does not automatically imply programming skills, but this semantic discussion is out of the scope of this paper.

[6] http://www.arnetminer.org.

1.2. Contribution and plan of this paper

This paper focuses on professional social networks allowing user endorsements for particular skills, such as LinkedIn and ResearchGate. Our main contributions can be summarized as follows:

1. We introduce *endorsement deduction*: an algorithm to enrich/enhance the information contained in the digraph of endorsements corresponding to a specific skill ('target' skill or 'main' skill) in a social network. This algorithm adds new weighted arcs (corresponding to other skills) to the digraph of endorsements, according to the correlation of the other skills with the 'main' skill. We assume the existence of an 'ontology' that specifies the relationships among different skills.
2. After this pre-processing we can apply a ranking algorithm to the enriched endorsement digraph, so as to compute an authority score for each network member with respect to the main skill. In particular, we have used the (weighted) PageRank algorithm for that purpose, but in principle, any ranking method could be used, provided that it admits weighted digraphs (e.g. Hits [52]). The reasons why we have chosen PageRank in the first place are explained in Section 2.5. The authority score obtained by our method could be useful for searching people having a certain skill, for profile personalization, etc.
3. We propose a methodology to validate our algorithm, which does not rely as heavily on the human factor as previous validation methods, or on the availability of private information of the members' profiles. More precisely, we discuss the benefits of endorsement deduction in terms of (1) consistency with the results of simple weighted PageRank, (2) reduction in the number of ties and (3) robustness against spamming. Following this methodology, we test our solution on a synthetic network of 1493 nodes and 2489 edges, similar to LinkedIn, and fitted with endorsements [38].

The rest of the paper is organized as follows: Section 2 provides the essential concepts, terminology and notation that will be used throughout the rest of the paper. It also describes the PageRank algorithm, including the variant for weighted digraphs. After that, our proposal is explained in Section 3 together with a simple example. In Section 4 we compare the results obtained by ranking with deduction with those obtained by simple ranking, according to three criteria proposed by ourselves. Finally, in Section 5 we summarize our results, discuss some potential applications, and enumerate some open problems that arise as an immediate consequence of the preceding discussion.

2. Preliminaries

2.1. Terminology and notation

A *directed graph*, or *digraph* $D = (V, A)$ is a finite nonempty set V of objects called *vertices* and a set A of ordered pairs of vertices called *arcs*. The *order* of D is the cardinality of its set of vertices V. If (u, v) is an arc, it is said that v is *adjacent from* u. The set of vertices that are adjacent from a given vertex u is denoted by $N^+(u)$ and its cardinality is the *out-degree* of u, $d^+(u)$.

Given a digraph $D = (V, A)$ of order n, the adjacency matrix of D is an $n \times n$ matrix $\mathbf{M} = (m_{ij})_{n\times n}$ with $m_{ij} = 1$ if $(v_i, v_j) \in A$, and $m_{ij} = 0$ otherwise. The sum of all elements in the i-th row of M will be denoted Σm_{i*}, and it corresponds to $d^+(v_i)$.

A *weighted digraph* is a digraph with (numeric) labels or *weights* attached to its arcs. Given $(u, v) \in A$, $\omega(u, v)$ denotes the weight attached to that arc. In this paper we only consider directed graphs with non-negative weights. The reader is referred to Chartrand and Lesniak [7] for additional concepts on digraphs.

2.2. PageRank vector of a digraph

PageRank [3,36] is a link analysis algorithm that assigns a numerical weighting to the vertices of a directed graph. The weighting assigned to each vertex can be interpreted as a relevance score of that vertex inside the digraph.

The idea behind PageRank is that the relevance of a vertex increases when it is linked from relevant vertices. Given a directed graph $D = (V, A)$ of order n, assuming each vertex has at least one outlink, we define the $n \times n$ matrix $\mathbf{P} = (p_{ij})_{n\times n}$ as,

$$p_{ij} = \begin{cases} \frac{1}{d^+(v_i)} & \text{if } (v_i, v_j) \in A, \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

Those vertices without oulinks are considered as if they had an outlink pointing to each vertex in D (including a loop link pointing to themselves). That is, if $d^+(v_i) = 0$ then $p_{ij} = 1/n$ for each j. Note that $\mathbf{P}$ is a stochastic matrix whose coefficient p_{ij} can be viewed as the probability that a surfer located at vertex v_i jumps to vertex v_j, under the assumption that the next movement is taken uniformly at random among the arcs emanating from v_i. When the surfer falls into a vertex v_i such that $d^+(v_i) = 0$, then he/she is able to restart the navigation from any vertex of D uniformly chosen at random. So as to permit this random restart behavior when the surfer is at any vertex (with a small probability $1 - \alpha$), a new matrix $\mathbf{P}_\alpha$ is created as,

$$\mathbf{P}_\alpha = \alpha\mathbf{P} + (1 - \alpha)\frac{1}{n}\mathbf{J}^{(n)}, \tag{2}$$

where $\mathbf{J}^{(n)}$ denotes the order-n all-ones square matrix.

By construction, $\mathbf{P}_\alpha$ is a positive matrix [35], hence, $\mathbf{P}_\alpha$ has a unique positive eigenvalue (whose value is 1) on the spectral circle. The PageRank *vector* is defined to be the (positive) left-hand eigenvector $\mathcal{P} = (p_1, \ldots, p_n)$ with $\sum_i p_i = 1$ (the left-hand Perron vector of $\mathbf{P}_\alpha$) associated to this eigenvalue. The probability α, known as the *damping factor*, is usually chosen to be $\alpha = 0.85$.

The relevance score assigned by PageRank to vertex v_i is p_i. This value represents the long-run fraction of time the surfer would spend at vertex v_i.

2.3. PageRank vector of a weighted digraph

When the input digraph is weighted, the PageRank algorithm is easily adapted so that the probability that the random surfer follows a certain link is proportional to its (positive) weight [49]. This is achieved by slightly modifying the definition, previously given in Eq. (1), of matrix $\mathbf{P}$ so that,

$$p_{ij} = \begin{cases} \frac{\omega(v_i, v_j)}{\sum_{v\in N^+(v_i)} \omega(v_i, v)} & \text{if } (v_i, v_j) \in A, \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

Nodes with no outlinks are treated in the same way as before.

2.4. Personalized PageRank

Personalized PageRank [21] is a variant of PageRank in which, when the surfer performs a random restart (with probability $1 - \alpha$), the vertex it moves to is chosen at random according to a *personalization vector* $\mathbf{v} = (\mathbf{v_i})$ so that $\mathbf{v_i}$ is the probability of restarting navigation from vertex v_i. Now, matrix $\mathbf{P}_\alpha$ is computed as,

$$\mathbf{P}_\alpha = \alpha\mathbf{P} + (1 - \alpha)\mathbf{e}\mathbf{v}^T, \tag{4}$$

with $\mathbf{e}$ denoting the order-n all ones vector. As a result, the computation is biased to increase the effect of those vertices v_i receiving a larger $\mathbf{v_i}$.

2.5. PageRank *in context*

PageRank is actually a variant of spectral ranking, a family of ranking techniques based on eigenvalues and eigenvectors. Vigna [46] traces the origins of spectral ranking to the 1950's, with [23] and [47]. Afterwards, the method was rediscovered several times until the 1970's. Other articles which are frequently cited as the original sources of spectral ranking include [4,16,39]. Eventually, the method became widely popular when it was adopted by Google for its search engine.

The reasons for the popularity of spectral ranking in general, and Google's PageRank in particular, are, in the first place, their nice mathematical properties. Under some reasonable mathematical assumptions, PageRank produces a unique ranking vector, which reflects very accurately the relative importance of the nodes. Other competing algorithms, such as Hits and Salsa do not guarantee such properties [13]. As we have seen in the previous sections, PageRank can be adapted to weighted digraphs and supports personalization. Additionally, it can be efficiently approximated [2,5], and can be computed in a parallel or distributed framework [31,43].

Besides information retrieval, spectral ranking in general, and PageRank in particular, have been applied in social network analysis [4,37,48], scientometrics [15,34,39,50], geographic networks [16], and many other areas with great success.

Last but not least, Google's PageRank has withstood the test of public scrutiny, as it has been validated by millions of users for more than 15 years now.

3. Endorsement deduction and ranking

Let us consider a professional network in which users can indicate a set of topics they are skilled in. So as to attract attention, some dishonest network members could be tempted to set an over-inflated skill list. The effect of such malicious behavior is reduced if network members are able to endorse other users for specific skills and the relevance they get depends on the received endorsements. Since cheating users will rarely be endorsed, their relevance in the network will be kept low.

In such a social network we get an endorsement digraph for each skill. Our objective is to compute an authority ranking for a particular skill, which is not only based on the endorsement digraph of that particular skill, but also takes into account the endorsement digraphs of other related skills. From now on, the skill for which we want to compute the ranking will be called the *main skill*.

Let $S = \{s_0, s_1, \dots s_\ell\}$ be the set of all possible skills, with s_0 being the main skill. Let $D_k = (V, A_k)$ denote the endorsement digraph corresponding to skill s_k, and let $\mathbf{M}_k$ be its adjacency matrix.

We now define the *skill deduction matrix* $\mathbf{\Pi} = (\pi_{kt})$ as follows: given a pair of skills s_k and s_t, π_{kt} represents the probability that a person skilled in s_k also possesses the skill s_t. In other words, from s_k we can infer s_t with a degree of confidence π_{kt}. By definition, $\pi_{kk} = 1$ for all k. In this way, if some user endorses another user for skill s_k but no endorsement is provided for skill s_t, we can deduce that an endorsement (for s_t) should really be there with probability π_{kt}. In general, $\mathbf{\Pi}$ will be non-symmetric and sparse, thus it is better represented as a directed graph with weighted arcs.

Note that $\mathbf{\Pi}$ can be seen as an ontology that also accounts for hierarchies among the topics. For example, 'Applied Mathematics' is a sub-category of 'Mathematics', and this would be reflected in $\mathbf{\Pi}$ as a link with weight 1, going from 'Applied Mathematics' to 'Mathematics'.

Our proposal takes as input the skill deduction matrix $\mathbf{\Pi}$, together with those endorsement digraphs D_k, with $0 < k \leq \ell$, such that $\pi_{k0} > 0$. Without loss of generality, we will assume that the set of skills related to s_0 is $S_0 = \{s_k \mid k \neq 0,\ \pi_{k0} > 0\} = \{s_1, \dots, s_\ell\}$.

The proposed endorsement deduction method constructs a weighted endorsement digraph $D_0^{we} = (V, A_0^{we})$ on skill s_0, with weights ranging from 0 to 1, considering the endorsements deduced from related skills $\{s_1, \dots, s_\ell\}$.

1. First of all, if user v_i directly endorsed v_j for skill s_0, that is $(v_i, v_j) \in A_0$, then D_0^{we} has arc $(v_i, v_j) \in A_0^{we}$ with $\omega(v_i, v_j) = 1$ (that endorsement receives a maximum confidence level).
2. If $(v_i, v_j) \notin A_0$ but $(v_i, v_j) \in A_k$, for just one k, $1 \leq k \leq \ell$, then arc (v_i, v_j) is added to D_0^{we} with weight $\omega(v_i, v_j) = \pi_{k0}$, that is, the arc is assigned a weight that corresponds to the probability that v_i also considers v_j proficient in skill s_0, given an existing endorsement for skill s_k.
3. Finally, if $(v_i, v_j) \notin A_0$ but $(v_i, v_j) \in A_{k_1}, \dots, A_{k_\ell}$, then the arc (v_i, v_j) is assigned a weight corresponding to the probability that v_i would endorse v_j for s_0 given his/her endorsements for $s_{k_1}, \dots, s_{k_l}$. That is, let "$(s_{k_i} \to s_0)$" denote the event "endorse for skill s_0 given an endorsement for s_{k_i} (its probability is $p(s_{k_i} \to s_0) = \pi_{k_i,0}$) then (v_i, v_j) is assigned a weight that corresponds to the probability of the union event "$\cup_{k_i \in \{k_1, \dots, k_\ell\}} (s_{k_i} \to s_0)$", assuming those events are independent.

Next we show how to construct the weighted adjacency matrix of D_0^{we} by iteratively adding deduced information from related skills. Computations are shown in Eq. (5). After the k-th iteration, matrix $\mathbf{Q}_k$ corresponds to the weighted digraph of skill s_0 after having added deduced information from skills $s_1, \dots, s_k$. The matrix computed after the last iteration $\mathbf{Q}_\ell$ corresponds to the weighted adjacency matrix of digraph D_0^{we}. Computations can be carried out as follows:

$$\mathbf{Q}_0 = \mathbf{M}_0 \tag{5a}$$

$$\mathbf{Q}_k = \mathbf{Q}_{k-1} + \pi_{k0}((\mathbf{J}^{(n)} - \mathbf{Q}_{k-1}) \circ \mathbf{M}_k), \quad \text{for } k = 1, \dots, \ell, \tag{5b}$$

where the symbol '$\circ$' represents the Hadamard or elementwise product of matrices.

Note that Eq. (5b) acts on the entries of $\mathbf{Q}_{k-1}$ that are smaller than 1, and the entries equal to 1 are left untouched. If some entry $\mathbf{Q}_{k-1}(i, j)$ is zero, and the corresponding entry $\mathbf{M}_k(i, j)$ is non-zero, then $\mathbf{Q}_{k-1}(i, j)$ takes the value of $\mathbf{M}_k(i, j)$, modified by the weight π_{k0}. This corresponds to the second case above.

If $\mathbf{Q}_{k-1}(i, j)$ and $\mathbf{M}_k(i, j)$ are both non-zero, then we are in the third case above. To see how it works, let us suppose that some entry $\mathbf{M}_0(i, j)$ is zero, but the corresponding entries $\mathbf{M}_1(i, j), \mathbf{M}_2(i, j), \mathbf{M}_3(i, j), \dots$, are all equal to 1. In other words, person i does not endorse person j for the main skill (skill 0), but does endorse person j for skills $1, 2, 3, \dots$. In order to simplify the notation we will drop the subscripts i, j, and we will refer to q_k as the (i, j)-entry of $\mathbf{Q}_k$. Applying Eq. (5), we get:

$$\begin{aligned}
q_0 &= m_0 = 0 \\
q_1 &= q_0 + \pi_{1,0}(1 - q_0) = \pi_{1,0} \\
q_2 &= q_1 + \pi_{2,0}(1 - q_1) = \pi_{1,0} + \pi_{2,0}(1 - \pi_{1,0}) \\
&= \pi_{1,0} + \pi_{2,0} - \pi_{1,0}\pi_{2,0} \\
q_3 &= q_2 + \pi_{3,0}(1 - q_2) \\
&= \pi_{1,0} + \pi_{2,0} - \pi_{1,0}\pi_{2,0} + \pi_{3,0}(1 - (\pi_{1,0} + \pi_{2,0} - \pi_{1,0}\pi_{2,0})) \\
&= \pi_{1,0} + \pi_{2,0} + \pi_{3,0} - \pi_{1,0}\pi_{2,0} - \pi_{1,0}\pi_{3,0} - \pi_{2,0}\pi_{3,0} \\
&\quad + \pi_{1,0}\pi_{2,0}\pi_{3,0} \\
&\vdots
\end{aligned}$$

which corresponds to the probabilities of the events $(s_1 \to s_0)$, $(s_1 \to s_0) \cup (s_2 \to s_0)$, $(s_1 \to s_0) \cup (s_2 \to s_0) \cup (s_3 \to s_0)$, and so on.

Once we have the matrix $\mathbf{Q}_\ell = (q_{ij})_{n \times n}$, we can apply any ranking method that admits weighted digraphs, such as the weighted

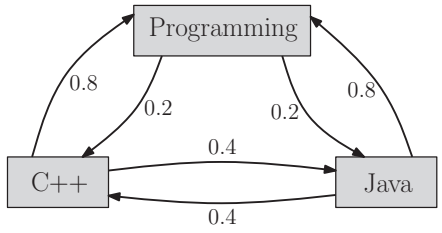

Fig. 1. Directed graph representing a skill deduction matrix $\mathbf{\Pi}$.

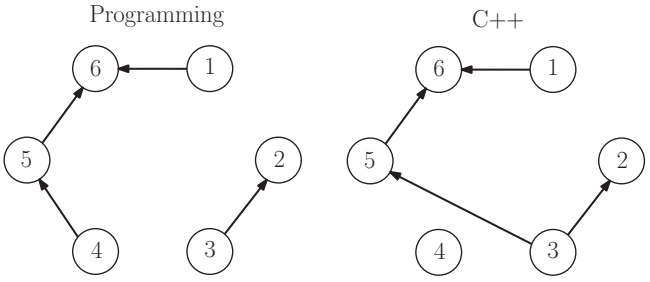

Fig. 2. Endorsements for 'Programming' (left) and 'C++' (right).

PageRank algorithm [49]. For that purpose we have to construct the normalized weighted link matrix $\mathbf{P}$, as in Eq. (3):

$$p_{ij} = \begin{cases} \frac{q_{ij}}{\Sigma q_{i*}} & \text{if } \Sigma q_{i*} > 0, \\ \frac{1}{n} & \text{if } \Sigma q_{i*} = 0. \end{cases} \tag{6}$$

Then we compute $\mathbf{P}_\alpha$ from $\mathbf{P}$, as in Eq. (4), and we finally apply the weighted PageRank algorithm on $\mathbf{P}_\alpha$.

3.1. An example

As a simple illustration, let us consider a set of three skills: 'Programming', 'C++' and 'Java'. The probabilities relating them, depicted in Fig. 1, have been chosen arbitrarily, but in practice, they could have been obtained as a result of some statistical analysis.

Let us further assume that we have a community of six individuals, labeled from '1' to '6'. Fig. 2 shows the endorsement digraphs among the community members for the skills 'Programming' and 'C++'.

Let us suppose that the skill 'Programming' is our main skill (skill 0). Thus, $\mathbf{Q}_0 = \mathbf{M}_0$ is the adjacency matrix of the digraph shown in Fig. 2 (left). If we compute the PageRank for the skill 'Programming', without considering its relationships with other skills, we get the following scores ($\mathcal{P}(v)$ denotes the PageRank score assigned to vertex v):

$$\mathcal{P}(1) = \mathcal{P}(3) = \mathcal{P}(4) = 0.0988,\ \mathcal{P}(2) = \mathcal{P}(5) = 0.1828,$$
$$\text{and } \mathcal{P}(6) = 0.3380.$$

In other words, on the basis of the endorsements for 'Programming' alone, the individuals '2' and '5' are tied up, and hence equally ranked.

Now we will include the endorsements for 'C++' in this analysis (skill 1). We apply Eq. (5) to compute $\mathbf{Q}_1$, as follows:

$$\mathbf{Q}_1 = \mathbf{Q}_0 + \pi_{1,0}((\mathbf{J}^{(6)} - \mathbf{Q}_0) \circ \mathbf{M}_1),$$

where $\pi_{1,0} = 0.8$, and $\mathbf{M}_1$ is the adjacency matrix of the digraph shown in Fig. 2 (right). This yields the endorsement digraph depicted in Fig. 3.

The PageRank scores assigned to nodes in that digraph are:

$$\mathcal{P}(1) = \mathcal{P}(3) = \mathcal{P}(4) = 0.0958,\ \mathcal{P}(2) = 0.1410,\ \mathcal{P}(5) = 0.2133,$$
$$\text{and } \mathcal{P}(6) = 0.3585.$$

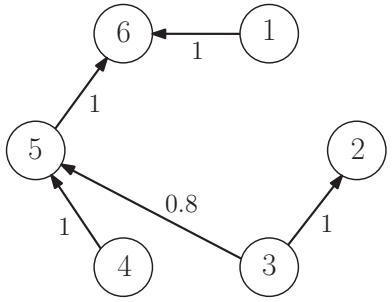

Fig. 3. Endorsements for 'Programming', with information deduced from 'C++'.

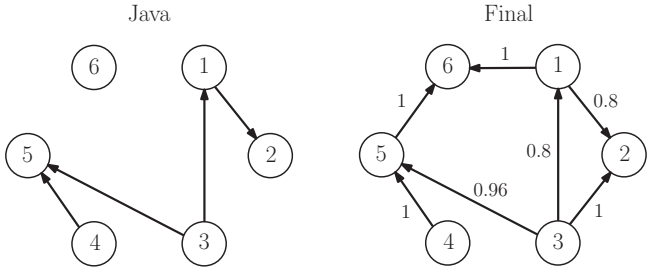

Fig. 4. Endorsements for 'Java' (left), and endorsements for 'Programming', with information deduced from 'C++' and 'Java' (right).

The individuals '2' and '5' are now untied, and we have better grounds to trust Programmer '5' over Programer '2'.

Let us now suppose that the endorsement digraph for 'Java' is the one given in Fig. 4 (left). We can include the endorsements for 'Java' in the same manner:

$$\mathbf{Q}_2 = \mathbf{Q}_1 + \pi_{2,0}((\mathbf{J}^{(6)} - \mathbf{Q}_1) \circ \mathbf{M}_2),$$

where again $\pi_{2,0} = 0.8$. The result is given in Fig. 4 (right).

If we apply PageRank to this final digraph we get:

$$\mathcal{P}(1) = 0.1178,\ \mathcal{P}(2) = 0.1681,\ \mathcal{P}(3) = \mathcal{P}(4) = 0.0945,$$
$$\mathcal{P}(5) = 0.2027,\ \text{and } \mathcal{P}(6) = 0.3224.$$

With the aid of the new endorsements, Programmer '1' differentiates itself from Programmers '3' and '4'.

3.2. Some properties

We now pay attention to some properties of endorsement deduction:

Proposition 1. *Let the matrices $\mathbf{Q}_k$ be defined as in Eq. (5). Then, for all $1 \le i, j \le n$, the entry $\mathbf{Q}_k(i,j)$ satisfies the following properties:*

(a) $0 \le \mathbf{Q}_k(i,j) \le 1$, *for all* $0 \le k \le \ell$.
(b) $\mathbf{Q}_k(i,j) \ge \mathbf{Q}_{k-1}(i,j)$, *for all* $1 \le k \le \ell$.
(c) $\mathbf{Q}_\ell(i,j) = 0$ *if, and only if,* $\mathbf{M}_k(i,j) = 0$ *for all* $0 \le k \le \ell$.
(d) $\mathbf{Q}_\ell(i,j) = 1$ *if, and only if, there exists some skill* k, *with* $0 \le k \le \ell$, *such that* $\mathbf{M}_k(i,j) = 1$, *and* $\pi_{k,0} = 1$. *In particular, if* $\mathbf{M}_0(i,j) = 1$, *then* $\mathbf{Q}_\ell(i,j) = 1$, *since* $\pi_{0,0} = 1$.

We omit the proofs, as they follow from the straightforward application of Eq. (5), and previous definitions.

Put into plain words, Proposition 1 states that a particular entry $\mathbf{Q}_k(i,j)$ can only grow with k, up to a limit of 1. This maximum can only be reached if i endorses j directly for skill 0, or for some other skill that implies skill 0 with probability 1. No other endorsement can have the same effect.

This implies that, if two members of the network, i and j, who were tied up in plain PageRank, get untied after deduction, it is because one of them has received additional endorsements for other

skills that are related to the main skill (or has received more relevant endorsements).

4. Simple ranking vs. ranking with deduction

4.1. Evaluation criteria

There is an extensive literature on the evaluation of information retrieval and ranking systems (see [[6], Section 1.2]; [42], and others). Several evaluation criteria and measures have been developed for that purpose, such as *precision*, *recall*, *F-measure*, *average precision*, *P@n*, etc. All these measures rely on a set of assumptions, which include, among others, the existence of:

1. a benchmark collection E of personal profiles (potential experts),
2. a benchmark collection S of skills,
3. a (total binary) judgement function r: $E \times S \to \{0, 1\}$, stating whether a person $e \in E$ is an expert with respect to a skill $s \in S$.

The above conditions have been taken from [[6], Section 1.2], and adapted to our situation. Unfortunately, none of these assumptions applies in our case.

To the best of our knowledge, there does not exist any reliable open-access ground-truth dataset of experts and skills, connected by endorsement relations. To begin with, the endorsement feature is relatively new, and still confined to a few social networks (e.g. LinkedIn and ResearchGate). In ResearchGate in particular, it was only introduced in February, 2013, and not enough data has accumulated so far. On the other hand, LinkedIn does not disclose sensitive information of its members (including their contacts or their endorsements), due to privacy concerns. Crawling the network, such as in [11] is not allowed: LinkedIn's terms of use specifically prohibit to *"scrape or copy profiles and information of others through any means (including crawlers, browser plugins and add-ons, and any other technology or manual work)"* (see [32]). Therefore, assumptions 1 and 2 do not hold in our case.

The third assumption is also problematic: even if we had a dataset with endorsements, we would still need a 'higher authority', or an 'oracle', to judge about the expertise of a person. Moreover, since our goal is to rank experts, a binary oracle would not suffice.

Traditionally, ranking methods have been validated by carrying out surveys among a group of users (e.g. [14]), which in our opinion, is very subjective and error-prone. We propose a more objective validation methodology, which is based on the following criteria:

1. *Sanity check:* our ranking with deduction is close to the ranking provided by PageRank. If we use endorsement deduction in connection with PageRank, results should not differ too much from PageRank; i.e. our method should only modulate the ranking provided by PageRank by introducing local changes. In order to evaluate the closeness between two rankings we can use some measure of rank correlation. Measures of rank correlation have been studied for more than a century now, and the best known of them are Spearman's correlation coefficient ρ [44], and Kendall's correlation coefficient τ [24].
2. Ranking with deduction results in less ties than PageRank. Ties are an expression of ambiguity, hence a smaller number of ties is desirable. In the example of Section 3, we have seen that ranking with deduction resolves a tie produced by PageRank. However, this has to be confirmed by meaningful experiments.
3. Ranking with deduction is more robust than PageRank to *collusion spamming*. Collusion spamming is a form of *link spamming*, i.e. an attack to the reputation system, whereby a group of users collude to create artificial links among themselves, and thus manipulate the results of the ranking algorithm, with the purpose of getting higher reputation scores than they deserve [17,18]. If the users create false identities (or duplicate identities) to carry out the spamming, the strategy is known as *Sybil attack* [12].

4.2. Experimental setup and results

We now proceed to evaluate our ranking with deduction, according to each of the above criteria. Our experimental benchmark consists of a randomly generated social network that replicates some features of LinkedIn at a small scale [38]. LinkedIn consists of an undirected *base network* (L), or *network of contacts*, and for each skill, the corresponding endorsements form a directed subgraph of (L). In [28], Leskovec formulates a model that describes the evolution of several social networks quite accurately, including LinkedIn, although this model is limited to the network of contacts (L), and does not account for the endorsements, since that feature was introduced in LinkedIn later.

Thus, we have implemented Leskovec's model and used it to generate an undirected network of contacts with 1493 nodes and 2489 edges, represented in Fig. 5.

Additionally, we have considered five skills: 1. Programming, 2. C++, 3. Java, 4. Mathematical Modeling, 5. Statistics. We have chosen these skills for two main reasons:

1. These five skills abound in a small LinkedIn community consisting of 278 members, taken from our LinkedIn contacts, which we have used as a sample to collect some statistics.
2. These five skills can be clearly separated into two groups or clusters, namely programming-related skills, and mathematical skills, with a large intra-cluster correlation, and a smaller inter-cluster correlation. This is a small-scale representation of the real network, where skills can be grouped into clusters of related skills, which may give rise to different patterns of interaction among skills.

We have computed the co-occurrences of the five skills in our small community, resulting in the co-occurrence matrix $\mathbf{\Pi}_1$ of Eq. (7). The entry $\mathbf{\Pi}_1(i, j)$ is the ratio between the number of nodes that have been endorsed for both skills, i and j, and the number of nodes that have been endorsed for skill i alone.

$$\mathbf{\Pi}_1 = \begin{pmatrix} 1 & 0.42 & 0.42 & 0.5 & 0.33 \\ 0.62 & 1 & 0.62 & 0.25 & 0.12 \\ 0.62 & 0.62 & 1 & 0.12 & 0.12 \\ 0.75 & 0.25 & 0.12 & 1 & 0.5 \\ 0.5 & 0.12 & 0.12 & 0.5 & 1 \end{pmatrix} \tag{7}$$

Now, for each skill we have constructed a random endorsement digraph (a random sub-digraph of the base network), in such a way that the above co-occurrences are respected. We have also taken care to respect the relative endorsement frequency for each individual skill. The problem of constructing random endorsement digraphs according to a given co-occurrence matrix is not trivial, and may bear some interest in itself [38]. We have chosen to skip the details here because it is not our main concern in the present paper. The base network and the endorsement digraphs can be downloaded from http://www.cig.udl.cat/sitemedia/files/MiniLinkedIn.zip.

Next, for each skill we have computed two rankings, one using the simple PageRank algorithm, and another one using PageRank with deduction. For PageRank with deduction we have used the skill deduction matrix $\mathbf{\Pi}_2$ given in Eq. (8). This matrix has been constructed by surveying a group of seven experts in the different areas involved.

$$\mathbf{\Pi}_2 = \begin{pmatrix} 1 & 0.7 & 0.7 & 0.4 & 0.3 \\ 1 & 1 & 0.6 & 0.4 & 0.3 \\ 1 & 0.7 & 1 & 0.4 & 0.3 \\ 0.3 & 0.2 & 0.2 & 1 & 0.8 \\ 0.3 & 0.2 & 0.2 & 1 & 1 \end{pmatrix} \tag{8}$$

For each skill we have computed the correlation between both rankings, and the number of ties in each case, according to the first two criteria described above. Additionally, in order to test the robustness of the method to collusion spamming, we have added to

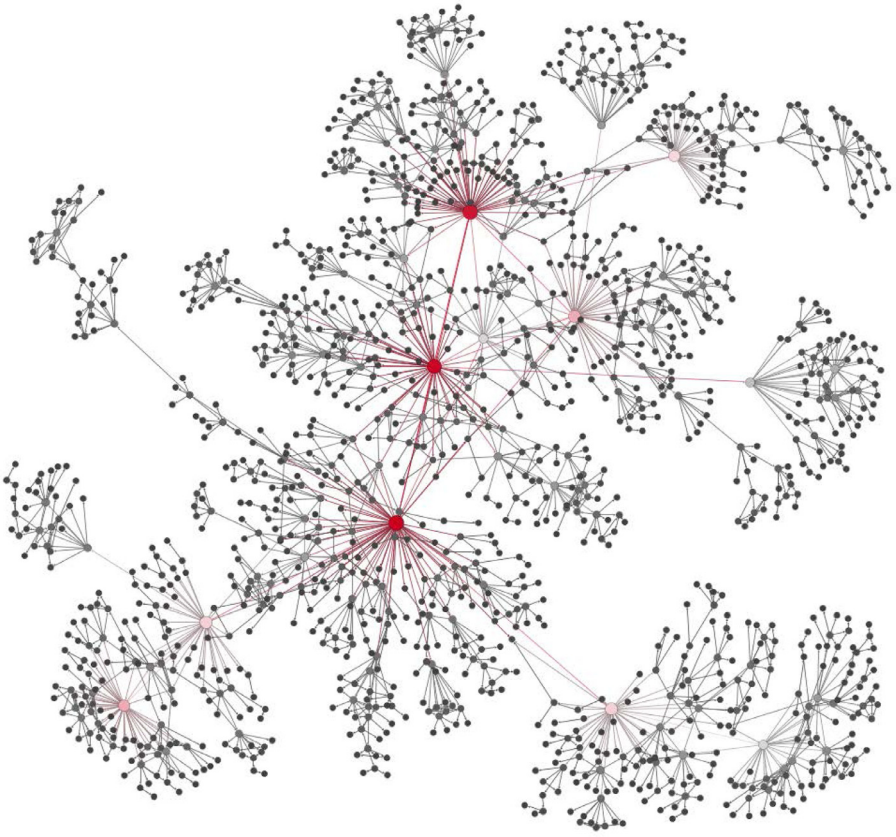

Fig. 5. Base network of 1493 nodes, used for experiments.

Table 1
Results from the first experiment.

Skill	Number of endorsements (arcs)	Correlat.		Number of ties			Position of leader		
		ρ	τ	Without deduct.	With deduc.	Reduc. (%)	Without deduct.	With deduc.	Fall (%)
Program.	220	0.89	0.76	1460	1316	10	1	48	3
C++	140	0.85	0.63	1478	1304	12	4	48	3
Java	137	0.85	0.63	1486	1292	13	1	48	3
Math mod	134	0.85	0.63	1483	1318	11	1	45	3
Statistics	128	0.85	0.63	1486	1304	12	1	45	3
AVG						**11.6**			**3**

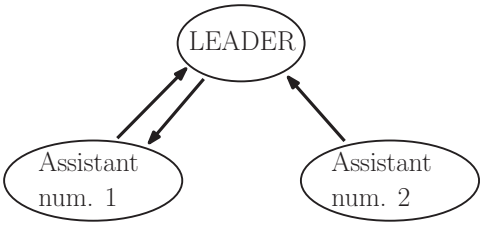

Fig. 6. Link spam alliance: three people collude to promote one of them.

each endorsement digraph, a small community of new members (the *cheaters*), who try to subvert the system by promoting one of them (their *leader*) as an expert in the corresponding skill. We have chosen the most effective configuration for such a spamming community, as described in [17], and depicted in Fig. 6. Thereupon we have compared the position of the leader of cheaters in simple PageRank with its position in PageRank with deduction.

Table 1 summarizes the results of the aforementioned experiments. Regarding the first criterion, we can see that there is a very high correlation between PageRank with deduction and PageRank without deduction for all skills. According to Kendall's τ, there is a significant agreement between both rankings, with a significance level of 0.001, or even better. Spearman's ρ shows an even higher agreement.

With respect to the second criterion, the experiments also yield unquestionable results: for all skills, the number of ties is significantly reduced. This is also reflected in the distribution of PageRank scores, shown in Fig. 7. After deduction, the scores are more evenly distributed.

As for the third criterion, in all cases there is a detectable drop in the position of the leader of cheaters, which may lead us to conclude that PageRank with deduction is more robust to collusion spam than simple PageRank. However, this may not lead us to the conclusion that PageRank with deduction is an effective mechanism against collusion spam. Actually, the spam alliance that we have introduced in our experiments is rather weak. If we strengthen the spam alliance, then PageRank with deduction may also be eventually deceived. Table 2 illustrates the effect of strengthening the spam alliance, by increasing the number of assistants from 2 to 8. For each spam alliance there are three columns, labeled as '−' (position of the leader in the ranking without deduction), '+' (position in the ranking

Table 2
Effect of endorsement deduction in the presence of different spam alliances.

Skill	Number of assistants																				
	2			3			4			5			6			7			8		
	–	+	%	–	+	%	–	+	%	–	+	%	–	+	%	–	+	%	–	+	%
Program.	1	48	3	1	31	2	1	10	1	1	6	0	1	4	0	1	2	0	1	1	0
C++	4	48	3	3	29	2	3	11	1	1	6	0	1	5	0	1	2	0	1	1	0
Java	1	48	3	1	30	2	1	10	1	1	6	0	1	4	0	1	2	0	1	1	0
Math mod	1	45	3	1	27	2	1	12	1	1	4	0	1	2	0	1	2	0	1	1	0
Statistics	1	45	3	1	28	2	1	11	1	1	4	0	1	2	0	1	2	0	1	1	0

Table 3
Results from the second experiment.

Skill	Number of endorsements (arcs)	Correlat.		Number of ties			Position of leader		
		ρ	τ	Without deduct.	With deduc.	Reduc. (%)	Without deduct.	With deduc.	Fall (%)
Program.	427	0.76	0.63	1428	625	56	1	175	12
C++	1793	0.97	0.93	1005	575	43	66	178	7
Java	1856	0.97	0.93	1005	566	44	63	180	8
Math Mod	1406	0.95	0.89	1130	652	42	56	168	7
Statistics	1447	0.96	0.90	1113	580	48	58	169	7
AVG						**47**			**8**

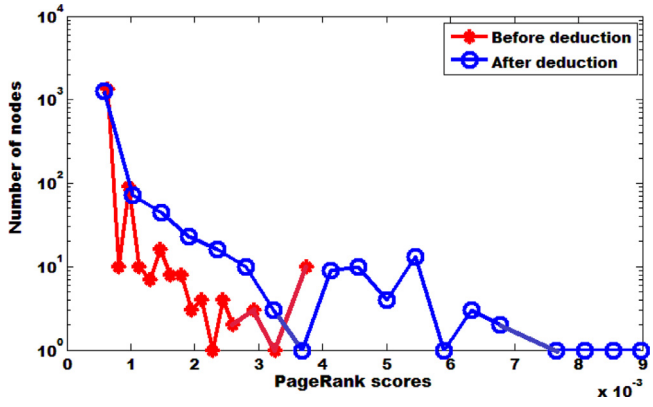

Fig. 7. Histograms of PageRank scores, before and after deduction.

with deduction), and '%' (effectiveness of deduction, measured as the leader's fall in position, in percentage points).

The simplest way to implement a collusion spam attack is the so-called Sybil attack, in which a single attacker creates several fake identities, without the need to collaborate with other people. Proactive measures against the Sybil attack focus on limiting the capability of a malicious user to create a large amount of accounts. It has been proven that a trusted central authority issuing credentials unique to an actual human being is the only method that may eliminate Sybil attacks completely [12]. Requiring fees per identity could mitigate them when the cost of the accounts is larger than the benefit for the attacker. Reactive measures try to mitigate the effect of such an attack. The SybilGuard [51] proposal bounds the number of identities a malicious user can create under the assumption that a malicious user could create many identities but few trust relations, so that there exists a small cut in the graph between Sybil nodes and honest ones.

Our proposal belongs to the second category. It is not designed to be a safeguard against the Sybil attack, but the experiments have shown that it provides some reactive measure behavior. In any case, the social network platform implementing our proposal could include all the proactive and reactive measures against sybil attacks without interferring with our method. A complete survey of attack and defense techniques for reputation systems is given in [19].

On the other hand, our endorsement digraphs are rather sparse, since our contacts are rather lazy when it comes to endorsing each other. It is reasonable to predict that if we should consider more skills, and if the total number of endorsements should increase, then the effects of PageRank with deduction will be stronger.

In order to verify this prediction, we have carried out a second experiment on the same base network. For practical reasons we have decided to keep the set of skills invariant for the moment, and increase the number of endorsements. Thus we have generated a second set of endorsement digraphs, with a larger number of arcs. This time we cannot enforce the co-occurrences observed in our small LinkedIn community. The co-occurrence matrix obtained is given in Eq. (9) for the sake of comprehensiveness.

$$\mathbf{\Pi}_3 = \begin{pmatrix} 1 & 0.88 & 0.87 & 1 & 0.61 \\ 0.32 & 1 & 0.9 & 0.73 & 0.61 \\ 0.31 & 0.89 & 1 & 0.63 & 0.59 \\ 0.42 & 0.85 & 0.75 & 1 & 0.7 \\ 0.24 & 0.67 & 0.66 & 0.66 & 1 \end{pmatrix} \tag{9}$$

Subsequently we have performed the same computations on this second set of endorsement digraphs, obtaining the results recorded in Table 3. These results fully confirm our prediction: There is an increase in the correlation coefficients (except in one case), as well as a larger reduction in the number of ties, and a more significant fall in the position of the leader of cheaters.

5. Conclusions and future research

In this paper we describe endorsement deduction, a preprocessing algorithm to enrich the endorsement digraphs of a social network with endorsements, such as LinkedIn or ResearchGate, which can then be used in connection with a ranking method, such as PageRank, to compute an authority score of network members with respect to some desired skill. Endorsement deduction makes use of the relationships among the different skills, given by an ontology in the form of a *skill deduction matrix*. A preliminary set of experiments shows that the rankings obtained by this method do not differ much from the rankings obtained by simple PageRank, and that this method represents an improvement over simple PageRank with respect to two additional criteria: number of ties, and robustness to collusion spam.

Our experiments also show that the benefits provided by PageRank with deduction are likely to increase in the future, with the densification of the endorsement networks, and the introduction of new skills. However, this has to be confirmed by larger-scale experiments.

It could also be interesting to test our method with other ranking algorithms, such as HITS.

Although LINKEDIN and RESEARCHGATE are perhaps the best examples at hand, this system can also be extended to other social networks and platforms. Take, for instance, the open access publishing platform arXiv.[7] In order to submit a paper on a particular topic, say 'Statistics', an author has to be endorsed by another trusted author for 'Statistics'. However, it might as well be possible to allow an author submit a paper on 'Statistics' if he/she has been endorsed for 'Probability Theory'.

There are many Internet forums, such as the 'StackExchange' suite, that assign an authority score to their members. As an illustration, let us pick one of the most popular forums of this family: The MATHSTACKEXCHANGE,[8] where users can pose questions and obtain answers about mathematical problems. As of today (July, 2015), the site has more than 152,000 members, and more than 467,000 questions have been posed. Members get credit points for the questions, answers, or comments that they post, via the votes of other members. A high authority score entitles a member to certain privileges. By design, all the votes are worth the same number of points, but a more realistic model would be to make the value of the votes dependent from the authority score of the voting person. Additionally, authority scores could be disassembled into areas of knowledge, since questions are tagged with the areas to which they belong (e.g. 'Linear Algebra', 'Calculus', 'Probability', etc.).

MATHOVERFLOW[9] is very similar to MATHSTACKEXCHANGE, but it focuses on more technical questions in state-of-the-art mathematics. Due to its more 'elitist' nature, MATHOVERFLOW is smaller than the MATHSTACKEXCHANGE. Nevertheless, it is also a success story, with its more than 37,000 users, and *circa* 62,000 questions posted, and it has become an undisputable actor in mathematical research, having attracted some of the world's top mathematicians [25].

A competitor to RESEARCHGATE is ACADEMIA, another academic social network designed to disseminate research results, ask and answer questions, and find like-minded collaborators. In both platforms, users can upload their papers, and tag them with the different research topics. Questions can be tagged too. It has been argued that, for the moment, these platforms have failed to attract some of the most experienced scientists. This may be partly due to the fact that scientists are conservative when it comes to adopting new technologies, but judging from EGOMNIA's experience, it may also be partly due to the unreliability (or outright inexistence) of scoring and ranking mechanisms [33]. It would not be difficult for RESEARCHGATE to make the RG score more reliable by adopting the techniques discussed above. As a starting point, it would be interesting to extend the experimental analysis of Section 4 to a RESEARCHGATE-like network, and compare the results with the ones obtained here.

It is worth observing that all these ideas are also applicable outside the academic realm. In principle, even GOOGLE's search engine could make use of these techniques to find content by synonyms.[10] In order to do that, they would need a very large semantic network, comprising all the potential keywords and their correlation.

Similarly, every social network or video repository displays some featured content on the start page, whose popularity has been computed on the basis of the number of votes (i.e. clicks on the 'Like' button). Yet, this content is usually tagged by topic, and hence, it might be possible to compute a more specific popularity score, and thus display content is specifically tuned to the user's interests. The authors in [26] propose a method for ranking weblogs. Their proposal consists in aggregating links by considering similarity in authors and topics between pairs of blogs.

[7] http://arxiv.org.
[8] http://math.stackexchange.com.
[9] http://mathoverflow.net.
[10] GOOGLE already has some functionality for synonyms via the '∼' operator.

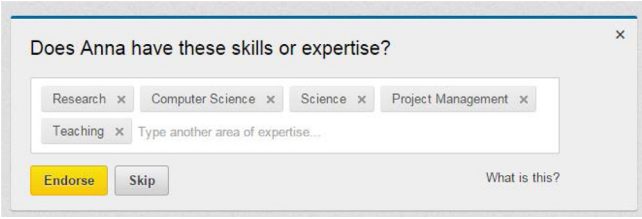

Fig. 8. First endorsement mechanism: batch endorsement.

In any case, for the practical implementation of these techniques, two obvious problems arise:

1. The first problem has to do with the estimation of the skill deduction matrix, which in this paper we estimate by polling a group of experts. There may be several alternatives to estimate the matrix from the social network itself, and it may be necessary to ponder the pros and cons of each alternative.
2. The second problem has to do with the feasibility of the computation. Assuming that the skill deduction matrix is available, computation of the weighted PAGERANK is a costly process for a large social network. Methods for accelerating PAGERANK computations have been considered in [9,22]; we might need to adapt them to our situation. Additionally, we might need methods to accelerate the computation of the accumulated endorsement matrix **Q**.

A subsidiary problem has to do with modeling the dynamics of endorsements in both LINKEDIN and RESEARCHGATE.

Acknowledgments

Authors have been partially funded by the Spanish Ministry of Economy and Competitiveness under projects TIN2010-18978, IPT-2012-0603-430000, and MTM2013-46949-P. The first two authors wish to dedicate this paper to the memory of our friend and colleague Josep Maria Ribó Balust, who could not live to see the conclusion of his work.

Appendix. The endorsement mechanism and its inconsistencies

We make a brief discussion about LINKEDIN's endorsement mechanism, which may be useful for the reader who is unfamiliar with the social network. The process starts when some user, say Anna, declares her skills (in this respect, LINKEDIN differs from other social networks, such as RESEARCHGATE, where users can suggest skills to their contacts). Then, Anna's contacts can endorse her for those alleged skills in three different ways:

1. When one of Anna's contacts (say Ben) opens Anna's profile, the system presents Ben with a list of Anna's presumed skills, and asks Ben whether it is true that Anna possesses those skills. By pressing a single button Ben can endorse Anna for all the skills in the list. We may call that mechanism *batch endorsement*. Fig. 8 shows the dialog box that is presented to Ben. The main advantage of batch endorsement is that it requires very little effort by Ben, since he only has to press a single button. Nevertheless, batch endorsement may be a source of inconsistencies, since the batch list presented to Ben is usually made up of several unrelated skills, and not necessarily those skills where Ben is an expert. It is true that Ben may remove some of the skills from the list, but that requires some additional effort.
2. After Ben has batch-endorsed Anna, he is then asked to endorse other users, one skill at a time. The people appear in groups of four, and their order of appearance, as well as their skills, seem to be random. Fig. 9 shows a group of four users waiting to be endorsed. This endorsement mechanism is more specific, but also

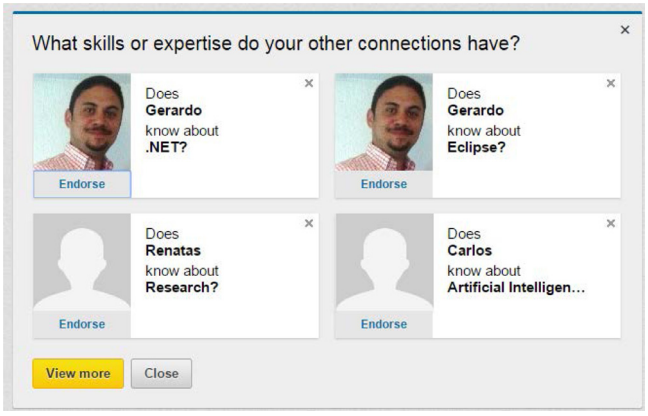

Fig. 9. Second endorsement mechanism: a group of four candidates to be endorsed.

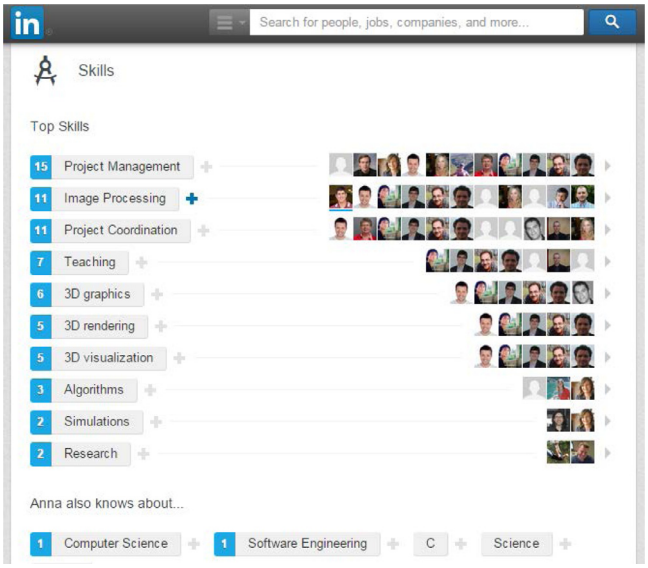

Fig. 10. Third endorsement mechanism: Anna's complete list of skills, which can be endorsed individually.

more time-consuming than batch endorsement, and people usually skip it after a few clicks.

3. Finally, if Ben wants to endorse Anna for some specific skill which did not appear, either in the initial batch list or in the subsequent endorsement suggestions, then he has to go to Anna's list of skills, and click on the specific skill he wants to endorse Anna for. This is by far the most reliable mechanism for endorsement, but it requires Ben's determination to make the endorsement. In Fig. 10 we can see Anna's list of skills, which can be clicked on individually.

Now it becomes evident why there are inconsistencies in people's endorsements. In fact, we can say that practically all profiles surveyed by us contain some inconsistency, although some inconsistencies are more obvious than others, and the actual percentage may vary according to the definition of inconsistencies that is agreed upon. In any case, whatever the convention adopted, the percentage of profiles containing some form of inconsistency will be very close to 100%.

We have manually surveyed 100 profiles from a pool of 3250, mainly from the areas of Mathematics and Computer Science. Our sampling method consisted in a random DFS of depth two with backtracking. We started at the root profile (belonging to one of the authors), then checked for inconsistencies, and selected one of the contacts at random, by generating a random number modulo the total number of contacts in the profile. Then we repeated the process from the new profile. Since the contacts are not visible for the profiles located at distance two (or greater) from the root profile, when we arrive at some profile of the second level, we check for inconsistencies and bactrack. Thus, our pool consists of all the profiles within distance two from the root profile.

In the literature it is possible to find several sampling methods that are probably more effective than ours, but at this point, effectiveness and accuracy are not a concern, since we are not collecting any formal statistics. Our main purpose at this point is to find examples of inconsistencies, and classify the most significant inconsistencies encountered. Roughly speaking, the inconsistencies can be classified in the following categories:

- Inconsistencies associated with the existence of hierarchies among skills. These can be subdivided into two groups:
  - Bottom-up inconsistencies: a user is endorsed for some subcategory of a larger category, but not for the larger category. This is by far the most common inconsistency we have encountered. For example, some users have many endorsements for several programming languages, but do not have any endorsement, or have very few endorsements for the skill 'Programming' itself, even though they have declared the skill 'Programming' in their profiles. Also in relation to programming, some users have several endorsements for one or more object-oriented programming languages, such as Java, but are not endorsed for the skill 'Object-Oriented Programming'. Still other users have been endorsed for 'Object-Oriented Programming', but not for 'Programming'. In a different domain, some users have been endorsed for several mathematical skills, e.g. 'Graph Theory', 'Discrete Mathematics', 'Applied Mathematics', 'Mathematical Modeling', 'Optimization' etc., but not for 'Mathematics'. Finally, some users have been widely endorsed for 'Digital Signal Processing', 'Digital Image Processing', 'Image Segmentation', but not for the more generic 'Image Processing'.
  - Top-down inconsistencies: a user is endorsed for some category, but is not endorsed for any sub-category of the larger category. For example, some users have been endorsed for the skill 'Programming', but not for any specific programming language.
- Inconsistencies associated with the existence of synonyms for a skill, or translations in different languages: a user is endorsed for some skill, but lacks endorsements in other skills that are synonyms of the first one. For example, several users have been endorsed either for the skill 'Simulation' or for 'Simulations', but not both. Some users have been endorsed for some skill (say 'Programming') in a language other than English, but not in English.
- Inconsistencies between the information contained in the endorsements, and the information contained in the rest of the profile, or the public information available about the user. E.g. some user, who is a renowned expert in some area, is not endorsed for the corresponding skill. This may happen if the user himself has not declared the skill. For two concrete examples, Prof. Edy Tri Baskoro, who is a renowned graph theorist, is not endorsed for 'Graph Theory', and Prof. Cheryl Praeger, who is a renowned group theorist, is not endorsed for 'Group Theory'.

This taxonomy does not attempt to cover all the situations encountered, which might be considered inconsistencies. It is very difficult to compile comprehensive statistics here, due to the huge diversity of cases, and to the subjectivism associated with defining inconsistent behavior, but in any case it becomes quite clear that the endorsement mechanism offers some scope for improvement.

References

[1] K. Balog, Y. Fang, M. de Rijke, P. Serdyukov, L. Si, Expertise retrieval, Found. Trends Inf. Retr. 6 (1–2) (2012) 127–256.
[2] Z. Bar-Yossef, L.-T. Mashiach, Local approximation of pagerank and reverse pagerank, in: Proceedings of Conference on Information and Knowledge Management, CIKM08, 2008, pp. 279–288.
[3] P. Berkhin, A survey on PageRank computing, Internet Math. 2 (1) (2005) 73–120.
[4] P. Bonacich, Factoring and weighting approaches to status scores and clique identification, J. Math. Sociol. 2 (1972) 113–120.
[5] A.Z. Broder, R. Lempel, F. Magul, J. Pedersen, Efficient pagerank approximation via graph aggregation, Inf. Retr. 9 (2006) 123û–138.
[6] S. Ceri, A. Bozzon, M. Brambilla, E. Della Valle, P. Fraternali, S. Quarteroni, Web Information Retrieval, Springer, 2013.
[7] G. Chartrand, L. Lesniak, Graphs and Digraphs, fourth ed., CRC Press, Boca Raton, 2004.
[8] F. Chung, A. Tsiatas, W. Xu, Dirichlet pagerank and trust-based ranking algorithms, in: Proceedings of the 8th International conference on Algorithms and Models for the Web Graph, WAW'11, 2011, pp. 103–114.
[9] G. Del Corso, A. Gulli, F. Romani, Fast PageRank computation via a sparse linear system, Internet Math. 2 (2005) 251–273.
[10] H. Deng, I. King, M.R. Lyu, Enhanced models for expertise retrieval using community-aware strategies, IEEE Trans. Syst. Man Cybern. – Part B: Cybern. 42 (2012) 93–106.
[11] C. Ding, Y. Chen, X. Fu, Crowd crawling: towards collaborative data collection for large-scale online social networks, Proceedings of Conference on Online Social Networks, COSN (2013) 183–188.
[12] J.R. Douceur, The Sybil Attack (LNCS), vol. 2429, Springer Verlag, 2002, pp. 251–260.
[13] A. Farahat, T. Lofaro, J.C. Miller, G. Rae, L.A. Ward, Authority rankings from Hits, PageRank, and Salsa: existence, uniqueness, and effect of initialization, SIAM J. Sci. Comput. 27 (2006) 1181–1201.
[14] K. Fujimura, H. Toda, T. Inoue, N. Hiroshima, R. Kataoka, M. Sugizaki, Blogranger – a multi-faceted blog search engine, Proceedings of Workshop on the Weblogging Ecosystem, WWW 2006 (2006) 22–26.
[15] B. González-Pereira, V.P. Guerrero-Bote, F. Moya-Anegón, A new approach to the metric of journals scientific prestige: the SJR indicator, J. Informetr. 4 (2010) 379–391.
[16] P.R. Gould, On the geographical interpretation of eigenvalues, Trans. Inst. Br. Geogr. 42 (1967) 53–86.
[17] Z. Gyöngyi, H. Garcia-Molina, Link spam alliances, in: Proceedings of the 31st International Conference on Very Large Data Bases, VLDB, 2005.
[18] Z. Gyöngyi, H. Garcia-Molina, Web spam taxonomy, in: Proceedings of Workshop on Information Retrieval on the Web, AIRWeb, 2005.
[19] K. Hoffman, D. Zage, C. Nita-Rotaru, A survey of attack and defense techniques for reputation systems, ACM Comput. Surv. 42 (2009) 1–31.
[20] S.H. Hashemi, M. Neshati, H. Beigy, Expertise retrieval in bibliographic network: a topic dominance learning approach, in: Proceedings of Conference on Information and Knowledge Management, CIKM, 2013, pp. 1117–1126.
[21] T.H. Haveliwala, Topic-sensitive pagerank: a context-sensitive ranking algorithm for web search, IEEE Trans. Knowl. Data Eng. 15 (4) (2003) 784–796.
[22] S.D. Kamvar, T.H. Haveliwala, C.D. Manning, G.H. Golub, Extrapolation methods for accelerating PageRank computations, in: Proceedings of Workshop on the Weblogging Ecosystem, WWW'03, 2003, pp. 261–270.
[23] L. Katz, A new status index derived from sociometric analysis, Psychometrika 18 (1) (1953) 39–43.
[24] M.G. Kendall, A new measure of rank correlation, Biometrika 30 (1/2) (1938) 81–93.
[25] E. Klarreich, The Global Math Commons, 2011, https://www.simonsfoundation.org/features/science-news/mathematics-and-physical-science/the-global-math-commons.
[26] A. Kritikopoulos, M. Sideri, I. Varlamis, Blogrank: ranking weblogs based on connectivity and similarity features, in: Proceedings of Workshop on Advanced Architectures and Algorithms for Internet Delivery and Applications, AAA-IDEA, 2006.
[27] J. Lee, J.-K. Min, A. Oh, C.-W. Chung, Effective ranking and search techniques for web resources considering semantic relationships, Inf. Process. Manag. 50 (2014) 132–155.
[28] J. Leskovec, Dynamics of Large Networks, (Ph.d. thesis) School of Computer Science, Carnegie-Mellon University, 2008.
[29] N. Li, D. Gillet, Identifying influential scholars in academic social media platforms, in: Proceedings of the IEEE/ACM Interntional Conference on Advances in Social Network Analysis and Mining, ASONAM, 2013, pp. 608–614.
[30] B. Khosravifar, Trust and Reputation in Multi-Agent Systems, (Ph.d. thesis), Department of Electrical and Computer Engineering. Concordia University. Montréal, Québec, Canada, 2012.
[31] C. Kohlschütter, P.A. Chirita, W. Nejdl, Efficient parallel computation of pagerank, Adv. Inf. Retr. (LNCS) 3936 (2006) 241–252.
[32] t. U. Agreement, 2014, https://www.linkedin.com/legal/user-agreement, (accessed 29.07.15).
[33] B. Lugger, Ein vergleich für forscher unter sich: Der ResearchGate score, 2012, http://www.scilogs.de/blogs/blog/quantensprung/2012-10-09/ein-vergleich-f-r-forscher-unter-sich-der-researchgate-score.
[34] N. Ma, J. Guan, Y. Zhao, Bringing pagerank to the citation analysis, Inf. Process. Manag. 44 (2008) 800–810.
[35] C.D. Meyer, Matrix Analysis and Applied Linear Algebra, SIAM, 2001.
[36] L. Page, S. Brin, R. Motwani, T. Winograd, The PageRank Citation Ranking: Bringing Order to the Web, Technical report, Stanford InfoLab, 1998.
[37] F. Pedroche, F. Moreno, A. González, A. Valencia, Leadership groups on social network sites based on personalized pagerank, Math. Comput. Model. 57 (2013) 1891û–1896.
[38] H. Pérez-Rosés, F. Sebé, Synthetic generation of social network data with endorsements, J. Simul. (2014), doi:10.1057/jos.2014.29.
[39] G. Pinski, F. Narin, Citation influence for journal aggregates of scientific publications: theory, with applications to the literature of physics, Inf. Process. Manag. 12 (1976) 297–312.
[40] J.M. Pujol, R. Sangüesa, Reputation measures based on social networks metrics for multi agent systems, in: Proceedings of the 4th Catalan Conference on Artificial Intelligence, CCIA-01, 2001, pp. 205–213.
[41] R. Punnarut, G. Sriharee, A researcher expertise search system using ontology-based data mining, in: Proceedings of the 7th Asia-Pacific Conference on Conceptual Modelling, APCCM, 2010, pp. 71–78.
[42] S. Robertson, Evaluation in information retrieval, in: M. Agosti, F. Crestani, M. Pasi (Eds.), Lectures on Information Retrieval, LNCS 1980, Springer Verlag, 2000, pp. 81–92.
[43] A.D. Sarma, A.R. Molla, G. Pandurangan, E. Upfal, Fast distributed pagerank computation, Theor. Comput. Sci. 561 (2015) 113–121.
[44] C. Spearman, The proof and measurement of association between two things, Am. J. Psychol. 15 (1904) 72–101.
[45] J. Tang, J. Zhang, L. Yao, J. Li, L. Zhang, Z. Su, ArnetMiner: extraction and mining of academic social networks, in: Proceedings of the 14th ACM International Conference on Knowledge Discovery and Data Mining, KDD, 2008, pp. 990û–998.
[46] S. Vigna, Spectral ranking, 2013, ArXiv:0912.0238v13.
[47] T.-H. Wei, The Algebraic Foundations of Ranking Theory, University of Cambridge, 1952.
[48] J. Weng, E.P. Lim, J. Jiang, Q. He, Twitterrank: finding topic-sensitive influential twitterers, in: Proceedings of the ACM International Conference on Web Search and Data Mining, WSDM 10, 2010, pp. 261–270.
[49] W. Xing, A. Ghorbani, Weighted PageRank algorithm, in: Proceedings of the 2nd Annual IEEE Conference on Communication Networks and Services Research, 2004, pp. 305–314.
[50] E. Yan, Y. Ding, Discovering author impact: a PageRankperspective, Inf. Process. Manag. 47 (2011) 125–134.
[51] H. Yu, M. Kaminsky, P.B. Gibbons, A. Flaxman, Sybilguard: defending against sybil attacks via social networks, in: Proceedings of SIGCOMM'06, 2006.
[52] X. Zhang, H. Yu, C. Zhang, X. Liu, An improved weighted Hits algorithm based on similarity and popularity, in: Proceedings of the 2nd IEEE International Multisymposium on Computer and Computational Sciences, 2007, pp. 477–480.