ECAI 2010
H. Coelho et al. (Eds.)
IOS Press, 2010

doi:10.3233/978-1-60750-606-5-595

Propagation of Opinions in Structural Graphs

Nardine Osman and **Carles Sierra** and **Jordi Sabater-Mir** [1]

Abstract. Trust and reputation measures are crucial in distributed open systems where agents need to decide whom or what to choose. Existing work has overlooked the impact of an entity's position in its structural graph and its effect on the propagation of trust in such graphs. This paper presents an algorithm for the propagation of reputation in structural graphs. It allows agents to infer their opinion about unfamiliar entities based on their view of related entities. The proposed mechanism focuses on the "part of" relation to illustrate how reputation may flow (or propagate) from one entity to another. The paper bases its reputation measures on opinions, which it defines as probability distributions over an evaluation space, providing a rich representation of opinions.

1 INTRODUCTION

Trust and reputation are key to the success of open systems. They aid agents in deciding what or whom to select. Extensive research has focused on various aspects of trust and reputation. The most relevant to our work are the mechanisms that compute reputation based on the sharing of past experiences, especially those that focus on formed *opinions*. This paper proposes a novel approach by highlighting the importance of the structural relations linking related entities and their use in indicating the flow of opinions from one entity to another. The mechanism allows a single agent, after it has formed opinions about a few entities (nodes) in a structural graph, to be able to infer its opinion concerning unfamiliar related entities. For example, say a new coffee machine is now out in the market and it has not been rated yet. What can an interested customer infer about this new item's reputation? Clearly, the reputation of other coffee machines of the same brand, or even other products of this brand in general, would be of help here. Hence, there is a need for representing the structural relations linking those entities together. A structural graph may be used, and the brand may be represented as one node in this graph, the brand's coffee machines as a child node to the former, the new coffee machine model as a child node to the latter, and so on. Such a representation will not only facilitate the flow of opinions amongst related entities, but also permit raters to choose the granularity level at which they would prefer to leave their opinions at. For instance, while one agent might be interested in rating this specific model in the future, it might also be interested in providing a rating for the brand's coffee machines in general.

Given this problem definition, the questions that arise and are addressed by this paper are: (1) how do we specify such structural graphs, (2) how do opinion-based reputation propagate in such graphs, and (3) how are all opinions, explicitly specified or propagated, aggregated to provide a final opinion-based reputation measure for a given node of this graph?

Question (1) above is addressed by Section 2. Questions (2) and (3) are addressed by Sections 3 and 4, which discuss the propagation and aggregation of opinions, respectively. The paper is then wrapped up by presenting our preliminary results in Section 5, an overview of related work in Section 6, and the conclusions in Section 7.

2 THE STRUCTURAL GRAPH

The running example of this paper is based on the publications world, where papers may be viewed as being composed of sections, proceedings composed of papers, books composed of chapters, and so on. We choose the publications field because the LiquidPub project (`http://project.liquidpub.org`) provides the framework needed for automatically building the structural graph as researchers write bits and pieces of their work.[2]

The basic idea is that researchers may write sections, papers, chapters, books, etc. They can then link these bits and pieces together, or reuse existing work, by making use of the "part of" relation. Researchers (or agents) may thus be authors of the nodes but also leave opinions on them. We understand opinions as probability distributions over an evaluation space, for a particular attribute, and at a moment in time; for example, one can define a set of elements for the evaluation space for *quality* of a node as {*poor, good, v.good, excellent*}. The set of attributes that opinions may address can be, for instance, {*novelty, clarity, significance, correctness*}.

The structural graph (SG) is formally defined below. We note, however, that the technique proposed in this paper is not restricted to the publications domain, and the SG graph defined below may be used for applications different than the LiquidPub project.

Definition 1

$$SG = \langle N, G, O, E, A, T, \mathcal{E}, \mathcal{F} \rangle$$

where, N is the set of nodes, G is the set of agents that may own or leave opinions on nodes, O is the set of opinions that agents may hold, E is the evaluation space for O, A is the set of attributes that opinions may address, T represents calendar time, $\mathcal{E} \subseteq N \times N$ specifies which nodes are part of the structure of which others (i.e. $(n, n') \in \mathcal{E}$ represents n as being part of n'), $\mathcal{F} : G \times N \times A \times T \to O$ is a relation that links a given agent, node, attribute, and time to their corresponding opinion.

A single opinion is then represented as the probability distribution $\mathbb{P}(E|G, N, A, T) \in O$. We note that probability distributions subsume classical approaches and are more informative.

3 PROPAGATION OF OPINIONS

Reputation is widely understood as group opinion. Opinions may be assigned by agents (or reviewers) to nodes of the structural graph. However, we believe these assigned opinions affect the opinion that

[1] IIIA-CSIC, Spain, email: {`nardine,sierra,jsabater`}`@iiia.csic.es`

[2] We assume agents do not need to save these graphs, since they may be provided by the relevant organisations (such as LiquidPub or Amazon).

the same agent has on neighbouring nodes, and therefore a propagation of opinions may be put in place. Our propagation concept is based on analogical reasoning: if a is similar to b, then the opinion about a is similar to that about b. For example, if 9 out of a book's 10 chapters are good, then probably the 10^{th} chapter is also good. When inferring opinions, we take into consideration all of the direct opinions that have been provided by the reviewer, and based on that, we try to maximise the similarity between parent and children nodes. Inferred values are essentially computed (and constrained) by the nodes that have received direct opinions. This section presents our proposal for this propagation mechanism.

The basic idea is that if a node does not receive a direct opinion, then its opinion may be *deduced* from its children nodes' opinions. This is because the parent node is structurally composed of its children nodes. Hence, the opinions on children nodes must necessarily influence the deduced opinion on a parent node. We refer to the direct opinion on a node or its deduced opinion from the parts that compose it as the *intrinsic* opinion of that node.

However, in the absence of information about the node itself, or the parts that compose it, then information may be inherited from the community to which one belongs. In other words, in the absence of information about intrinsic opinions, a node may inherit the opinions of its parents' nodes. For example, people assume that if a paper has been accepted by a highly reputable journal then the paper should be of good quality. Of course, when people actually read the paper, or parts of the paper, they may start constructing their own opinions, as opposed to relying on the default inherited one. We refer to opinions propagated from parent nodes to children nodes as *extrinsic opinions*. And we note that people usually rely on extrinsic opinions in the absence of intrinsic ones. Hence, extrinsic opinions may be viewed as providing some sort of a default measure that people may refer to in the absence of other more reliable sources of information.

In what follows, we first illustrate how intrinsic and extrinsic opinions are calculated (Sections 3.1 and 3.2, respectively). This is then followed by an algorithm summarising how opinions propagate within a structural graph through the calculation of intrinsic and extrinsic opinions (Section 3.3).

We note that propagation of opinions would only make sense for a single reviewer agent and a single attribute. In other words, one agent's opinion cannot affect another's,[3] or an opinion addressing novelty should not affect the correctness aspect of a paper.[4] The aggregation stage, described by Section 4 is responsible for dealing with several reviewer agents and several attributes being reviewed. Hence, in this section, since the reviewer agent $r \in G$ and the attribute $a \in A$ being assessed are fixed, we simplify notation and replace $\mathbb{P}(e_i|r,n,a,t)$ with $\mathbb{P}^t_n(e_i)$, or even $\mathbb{P}^t_n$.

3.1 Intrinsic opinions

The intrinsic opinion of a node is either based on the direct opinion that the node has received or on an aggregation of its children nodes' intrinsic opinions. In the latter case, we argue that the reliability of the aggregation should take into consideration the percentage of nodes with direct opinions that are contributing to the aggregated value, which we discuss below.

Reliability of opinions We say, when calculating node n's intrinsic opinion by aggregating the intrinsic opinions of its children nodes, it is crucial to know the proportion of nodes that have received a direct opinion in the structural sub-tree whose root node is n. This measure, in a way, provides information on the reliability of the aggregated intrinsic opinion of n. The higher the proportion of nodes with a direct opinion that are contributing to n's deduced intrinsic opinion, then the more reliable this deduced opinion is. In other words, the larger the number of direct opinions contributing to an inferred one then the more probable the inferred opinion is. We define the *reliability parameter* π as follows:

$$\pi^t_n = \begin{cases} 1 & \exists\, t' \leq t \cdot direct(\mathbb{P}^{t'}_n) \\ 0 & \forall\, t' \leq t \cdot \neg direct(\mathbb{P}^{t'}_n) \\ & \wedge\ \nexists\, c \cdot (c,n) \in \mathcal{E} \\ \sum\limits_{(c,n)\in\mathcal{E}} \dfrac{\pi^t_c}{|\{c' \mid (c',n) \in \mathcal{E}\}|} & \text{otherwise} \end{cases} \tag{1}$$

where the logical notation is that of first-order logic, and $direct(\mathbb{P}^{t'}_n)$ states that node n has received a direct opinion at time t'.

Note that if a node has received a direct opinion, then its π takes the value 1. However, if a node has never received a direct opinion from the agent and the node does not have any children nodes, then its π takes the value 0. Otherwise, the π of a node would be the average of its children's πs. We note that $\pi^t_n \in [0,1]$ and the value of π is non-decreasing along time t, since we currently assume the structural graph to be static.

Aggregation of children's opinions In the absence of direct opinions, we say a parent node n's intrinsic opinion is calculated as an aggregation of its children nodes' opinions, where each child node's contribution to the final aggregated value is based on the child node's π value. This is illustrated by the following equation:

$$\mathbb{P}^t_n = \frac{1}{\sum\limits_{(c,n)\in\mathcal{E}} \pi^t_c} \cdot \sum_{(c,n)\in\mathcal{E}} \pi^t_c \cdot \mathbb{P}^t_c \tag{2}$$

Decay of intrinsic opinions We consider the integrity of opinions to decrease with time. This is expressed by the following equation:

$$\mathbb{P}^t_n = \Lambda_i(\mathbb{D}^{t_n}_n, \mathbb{P}^{t_n}_n) \tag{3}$$

where $t_n \in T$ represents the latest point in time when a value (in this case, for $\mathbb{P}$ and $\mathbb{D}$) was recorded for node n, $\mathbb{D}$ is the probability distribution describing the node's extrinsic opinion and is the default opinion that the intrinsic opinion decays towards (we note that $\mathbb{D}$, which is defined in the following section, is updated along time), and Λ_i is a *decay function* satisfying the property: $\lim_{t\to\infty} \mathbb{P}^t_n = \mathbb{D}^{t_n}_n$. In other words, Λ_i is a function that makes $\mathbb{P}^t_n$ converge to $\mathbb{D}^{t_n}_n$ with time. One possible definition for Λ_i could be: $\mathbb{P}^t_n = (\mathbb{P}^{t_n}_n - \mathbb{D}^{t_n}_n)\nu^{\Delta_t} + \mathbb{D}^{t_n}_n$, where $\nu \in [0,1]$ is the decay rate, and:

$$\Delta_t = \begin{cases} 0 & t - t_n < \kappa \\ 1 + \frac{t-t_n}{t\_max} & \text{otherwise} \end{cases}$$

Δ_t serves the purpose of establishing a minimum 'grace' period during which the information does not decay and that once reached the information starts decaying. The period of 'grace' is determined by the parameter κ. The parameter t_max, which may also be defined in terms of multiples of κ, is used to control the pace of decay.

[3] Of course, persuasive dialogues can and actually do allow one agent to influence another's opinion. Additionally, [15, 1, 3] illustrate how the social network and the position of agents within their social network contribute to opinion formation and the influence of one agent's opinion on another's. However, these influences are beyond the scope of this paper.

[4] Again, there might be a strong correlation between attributes which might allow the opinion on one to affect the opinion on another. Additionally, reviewers may be susceptible to having their opinion on one attribute influence their opinion on another. However, again, these influences are beyond the scope of this paper.

The main idea behind this is that after this grace period, the decay happens very slowly; in other words, Δ_t decreases very slowly.

Initially, and in the absence of any information, we say $\mathbb{P}_n^{t_0} = \mathbb{D}_n^{t_0}$, where $t_0 \in T$ represents the initial time (or the time when node n joined the structural graph, in the case of a dynamic graph).

3.2 Extrinsic opinions

Extrinsic opinions are an aggregation of parents' intrinsic opinions. They represent the inheritance of parents' intrinsic opinions. Similar to intrinsic opinions, extrinsic opinions are calculated using a similar aggregation mechanism that uses the same reliability parameter π.

Aggregation of parents' opinions A child's extrinsic opinion is calculated by aggregating its parents' intrinsic opinions as follows:

$$\mathbb{D}_n^t = \frac{1}{\sum_{(n,p)\in\mathcal{E}} \pi_p^t} \cdot \sum_{(n,p)\in\mathcal{E}} \pi_p^t \cdot \mathbb{P}_p^t \tag{4}$$

Decay of extrinsic opinions The extrinsic opinion plays the role of the default opinion that a given opinion at a given node decays towards. However, similar to intrinsic opinions or any other type of information, the integrity of extrinsic opinions also decreases with time, although presumably at a much slower pace than the decay of intrinsic opinions towards extrinsic ones. Therefore, we say:

$$\mathbb{D}_n^t = \Lambda_e(\mathbb{F}, \mathbb{D}_n^{t_n}) \tag{5}$$

where $t_n \in T$ represents the latest point in time when a value for $\mathbb{D}$ has been recorded for node n, $\mathbb{F} = \frac{1}{|E|}$ represents the flat (or uniform) probability distribution that $\mathbb{D}_n^{t_n}$ decays towards, and Λ_e represents a decay function similar to Λ_i of equation 3 (although the decay rate should be much slower; i.e. the ν, κ, or t_max of Λ_e should be larger than those of Λ_i). In other words, while intrinsic opinions decay towards extrinsic ones, extrinsic opinions decay towards the flat distribution $\mathbb{F}$ at a much slower pace.

Initially, and with the absence of any information, we have $\mathbb{D}_n^{t_0} = \mathbb{F}$, where $t_0 \in T$ represents the initial time (or the time node n joined the structural graph, in the case of a dynamic graph).

3.3 An incremental propagation algorithm

As illustrated by the previous sections, a node's $\mathbb{P}$ and $\mathbb{D}$ values are based on the latest decayed values of all its children and parent nodes, respectively. And each of those is, in turn, based on the latest decayed values of their own children and parent nodes. In other words, to obtain the precise values of a given node at a given point in time, one should be recalculating every node's π, $\mathbb{P}$, and $\mathbb{D}$ values at every single step in time. Naturally, this is a very demanding computational algorithm which consumes loads of memory and time. For this reason, the algorithm we present (Algorithm 1) is an incremental algorithm for propagation that replaces equations 1, 2, and 4 with the incremental equations 6, 7, and 8, respectively, as illustrated below:

- **Incremental update of π:** Incremental equation 6 replaces equation 1 that updates node n's π value by simply considering the new π value of the child node c that triggered this update, as opposed to considering all the children nodes' π values.

$$\pi_n^t = \pi_n^{t_n} + \frac{\pi_c^t - \pi_c^{t_c}}{|\{c' \mid (c', n) \in \mathcal{E}\}|} \tag{6}$$

- **Incremental update of $\mathbb{P}$:** Equation 7 is used to update the inferred intrinsic opinion $\mathbb{P}$ of node n by simply considering the new π and $\mathbb{P}$ values of the child node c that triggered this update. The basic idea is that the old $\mathbb{P}$ value of node n ($\mathbb{P}_n^{t_n}$) is now modified by taking into consideration the new change in the π and $\mathbb{P}$ values of the child (π_c^t and $\mathbb{P}_c^t$). We remind the reader that, ideally, one should recalculate $\mathbb{P}_n^t$ by taking into consideration all the decayed values of the children, following equation 2. In practice, this is computationally expensive. Hence, we believe an approximate update that considers the new values of the child node may be good enough. Of course, further experimentation may be needed to help choose the exact weight that needs to be given for the new values.

$$\mathbb{P}_n^t = \frac{\pi_n^{t_n}\mathbb{P}_n^{t_n} + \pi_c^t\mathbb{P}_c^t}{\pi_n^{t_n} + \pi_c^t} \tag{7}$$

- **Incremental update of $\mathbb{D}$:** Similar to equation 7, equation 8 is used to update the extrinsic opinion $\mathbb{D}$ of node n by simply considering the new π and $\mathbb{P}$ of the parent node p that triggered this update.

$$\mathbb{D}_n^t = \frac{\pi_n^{t_n}\mathbb{D}_n^{t_n} + \pi_p^t\mathbb{P}_p^t}{\pi_n^{t_n} + \pi_p^t} \tag{8}$$

The basic idea behind Algorithm 1 is that the addition of a newly assigned opinion $\mathbb{P}$ to a node n (which is accompanied by the modification of n's π value) would trigger a wave of modifications in the structural graph by sending n's new π and $\mathbb{P}$ values to its neighbouring nodes (both parents and children), which would in turn send their modified values to their neighbouring nodes, and so on.

We note that in Algorithm 1, every node reacts to any new information it receives and updates its values accordingly, without having to keep track of the entire propagation mechanism. All it has to do is to transmit its updates to neighbouring nodes. This a distributed approach which suits applications where nodes may be physically located in different locations, such as the case of Liquid Publications.

Algorithm 1 Pseudocode of the propagation of opinions algorithm

while $\top$ **do**
  if n receives a new direct opinion $\mathbb{P}$ **then**
    Set n's π value to 1
    Send n's new π and $\mathbb{P}$ values to all parent nodes, keeping track of these nodes
  end if
  if n receives updated π and ı values from a child node **then**
    Update n's π and $\mathbb{P}$ values following Equations 6 and 7, respectively
    if The difference between n's old and new $\mathbb{P}$ is negligible, and n has at least one parent node **then**
      Send n's new π and $\mathbb{P}$ values to all parent nodes, keeping track of these nodes
    else
      Decay n's $\mathbb{D}$ following Equation 5
      Send n's new π and $\mathbb{P}$ values to all children nodes
    end if
  end if
  if n receives updated π and ı values from a parent node **then**
    Decay n's $\mathbb{P}$ and $\mathbb{D}$ following Equations 3 and 5, respectively
    Update n's $\mathbb{D}$ value following Equation 8
    if $[\_, \mathbb{P}_m^{t^{s'}}, \_] \notin W_n$ **then**
      Send n's new π and $\mathbb{P}$ values to all children nodes
    end if
  end if
  if n has not been updated for some time and n has no children nodes **then**
    Decay n's $\mathbb{P}$ following Equation 3
    Send n's new π and $\mathbb{P}$ values to all parent nodes, keeping track of these nodes
  end if
end while

There are three different cases of a node receiving new information. These are: (1) when a node receives a directly assigned opinion, (2) when a node receives updated π and $\mathbb{P}$ values from one of its children nodes, and (3) when a node receives updated π and $\mathbb{P}$ values from one of its parents' nodes. In summary, these three cases state that when a new opinion is assigned to some node, it propagates upwards (only if the difference between old and new values is not negligible) until it hits a root node. Then, the wave of propagation would start moving back down the tree to hit leaf nodes. And it is during these waves that nodes update their values. For this reason, there is a concern that some sections of the structural graph might not be hit by any wave for a long period of time. As a result, we introduce a fourth case that allows leaf nodes to decay their $\mathbb{P}$ values if the latest update is considered old enough, and send their updated values to their parents, triggering a new wave of opinion propagation in the structural graph. We call this the *house keeping* action that tries to keep the values in the structural graph as up to date as possible. We refer to the time interval that specifies how often a $\mathbb{P}$ value of a leaf node should be updated as the *HK* interval.[5]

4 AGGREGATION OF OPINIONS

The previous section has illustrated how a single agent's opinion on a node, and with respect to a given attribute, propagates within a structural graph. However, the ultimate goal is to be able to compute the reputation of a single node based on all the agents' opinions, and possibly for all attributes. In what follows we illustrate how this may be achieved. We note that the previous simplification of replacing $\mathbb{P}(e_i|r,n,a,t)$ with $\mathbb{P}^t_n$ no longer holds here, since the aggregation needs to be performed for all reviewer agents and/or for all attributes.

1. **Aggregating opinions for all attributes:**
   In this case, we aggregate the opinions that a single reviewer agent r holds w.r.t. several attributes of a given node n:

$$\mathbb{P}(e_i|r,n,\_,t) = \frac{\sum_{a\in A \,\wedge\, \mathbb{P}(e_i|r,n,a,t)\neq\mathbb{F}} \rho(a)\cdot\mathbb{P}(e_i|r,n,a,t)}{\sum_{a\in A \,\wedge\, \mathbb{P}(e_i|r,n,a,t)\neq\mathbb{F}} \rho(a)} \quad (9)$$

   Equation 9 states that the final opinion a reviewer agent r forms about a node n at time t ($\mathbb{P}(e_i|r,n,\_,t)$) is an aggregation of r's opinions about n w.r.t. all attributes, based on a preference value $\rho(a)$ that specifies the preference of each attribute $a \in A$. We assume that the agent computing the final opinion may specify the preferences of the attributes. Alternatively, we say that a default distribution may be provided with the set A by the responsible organisation, such as a conference specification in the LiquidPub case. Measures calculating the correlation between attributes, such as those described by [6], may also contribute to the specification of the $\rho(a)$ values. Finally, the default case may assume an equal preference to each. In any case, studying the correlation between attributes and their effect on the aggregation mechanism is outside the scope of this paper.
2. **Aggregating opinions for all reviewers:**
   In this case, we aggregate the opinions of several reviewer agents with respect to a given attribute a of a node n:

$$\mathbb{P}(e_i|\_,n,a,t) = \frac{\sum_{r\in G \,\wedge\, \mathbb{P}(e_i|r,n,a,t)\neq\mathbb{F}} \Omega(r,n)\cdot\mathbb{P}(e_i|r,n,a,t)}{\sum_{r\in G \,\wedge\, \mathbb{P}(e_i|r,n,a,t)\neq\mathbb{F}} \Omega(r,n)} \quad (10)$$

   Equation 10 aggregates all reviewer agents' opinions based on the reliability $\Omega(r,n)$ of the reviewer agent r in rating node n. The definition of Ω is complex, as it is affected by the reviewer agent's expertise in the field of n, its history of being correct (in other words, how close were its past reviews to the final group opinion), its history of bias, the degree of collaborative or competitive relationship between the reviewer and the node's owner(s) agent(s), etc. These issues are outside the scope of this paper; however, for our publications example, we base $\Omega(r,n)$ on the reviewer's h-index, which could be viewed as an indication of expertise.
3. **Calculating a final reputation measure $R^t_n \in [0,1]$:**
   When computing reputation, some users may be interested in calculating a single numeric value R for node n at time t, as opposed to a probability distribution. If this is the case, then the final probability distribution of node n at time t ($\mathbb{P}(e_i|\_,n,\_,t)$) may be translated into a numeric value in the range $[0,1]$ by calculating the center of gravity of the probability distribution, following the transformation equation of [8], which we present below:

$$R^t_n = \frac{1}{2\cdot|E|}\sum_{e_i\in E}(2\cdot(i-1))\cdot\mathbb{P}(e_i|\_,n,\_,t) \quad (11)$$

   where $\mathbb{P}(e_i|\_,n,\_,t)$ represents the value of the distribution at the element $e_i \in E$ and i represents the position of e_i.

When a user is interested in computing a final opinion measure of a node n at time t, then the latest $\mathbb{P}$ values should be obtained for all reviewer agents on all attributes. It is then up to the user to choose which of the steps above to perform, and in which order.

We note that alternative aggregation mechanisms may also be considered. [13] describes three different mechanisms: (1) the **dependent method** is used when dependencies are assumed to exist amongst the distributions being aggregated (this is useful if either correlations between attributes or strong social links between agents are observed), (2) the **independent method** amplifies similar probability distributions when it is believed that there are no dependencies between the distributions being aggregated, and (3) the Υ **method** computes the probability distribution that tries to maximise certainty.

5 RESULTS

To evaluate the correctness of the proposed propagation algorithm for the publications example, we present here initial results from simulated data. The correctness of the algorithm is based on comparing the opinions before and after propagation to check for consistency. We note that instead of generating reviewers' opinions, we choose to base initial direct opinions of a paper on the reputation of its authors by relying on their h-indexes. This is mainly because it may be argued that simulating reviewers' opinions requires a deeper analysis of reviewers' behaviour. The evaluation process is outlined below.

Step 1: Simulating data A graph is generated to represent two conference proceeding of the same series: in other words, a simple 3-level tree is generated. The root node represents the conference series, its two middle level nodes represent the conference's two proceedings and its 60 children nodes represent the 60 papers of the proceedings (30 papers per proceeding). We then created authors, assigned authors to the conference papers, and generated an h-index for each of these authors following the following constraints:[6]

[5] We believe there is a correlation between the *HK* interval and the attention a given node receives from a given reviewer agent. One plausible scenario is that when the attention is low, the *HK* value should be relatively high to keep things up to date. The *HK* values would then drop as attention increases. However, if the attention reaches extreme high values, then the *HK* value would increase again to prevent huge changes in inferred opinions every time a new opinion is assigned.

[6] Although it was straightforward to obtain real data on a conference proceedings, its papers, and their authors, obtaining the h-indexes for all authors whose papers has been accepted by a given conference was not as easy. Hence, we relied on simulated data.

- The number of authors per paper follows a Gaussian function whose expected value is $\mu = 3$ and its standard deviation $\sigma = 1$.
- The number of papers per author follows a Gaussian function whose expected value is $\mu = 1$ and its standard deviation $\sigma = 0.2$.
- The h-indexes of authors follow a Gaussian function whose expected value is $\mu = 8$ and its standard deviation $\sigma = 4$.

Step 2: Populating opinions We say the initial direct opinion that a node receives may be deduced from the reputation of its authors, which we base on the authors' h-index. As such, there is a need to transform h-index measures into probability distributions, which is carried out following equation 12:

$$\mathbb{A}_h^\alpha(e_i) = \frac{f(h, e_i)}{\sum_{e_i \in E} f(h, e_i)} \tag{12}$$

where h is the h-index of author α, $\mathbb{A}_h^\alpha(e_i)$ is the value of the probability distribution at point $e_i \in E$, and $f(h, e_i)$ is defined as:

$$f(h, e_i) = \frac{1}{e^{\left|\frac{h}{max\_h} - \frac{i}{|E|}\right|}} \tag{13}$$

where e represents Euler's number, h represents the h-index of the author, max_h represents the maximum h-index value in the sample, and i represents the position of the element e_i of the state space E.

Essentially, equation 13 states that $f(h, e_i)$ should have a high value when h and i both have either high or low values, whereas it should have a low value when either h or i has a high value and the other a low one. This is because prestigious authors (authors with high h-indexes) very probably write good papers (papers whose opinions have high values for $e_i \in E$, where i has a high value, and lower values otherwise), and unknown authors (authors with low h-indexes) very probably write not so good papers (papers whose opinions have high values for $e_i \in E$, where i has a low value, and lower values otherwise).

Equation 12 is then used to normalize the result of equation 13 to ensure that $\sum_{e_i \in E} \mathbb{A}_h^\alpha(e_i) = 1$.

While equation 12 computes the opinions about authors as probability distributions, our ultimate goal is to generate a direct opinion for each paper (or node) based on the aggregation of its authors' opinions. This is calculated as follows:

$$\mathbb{P}_n^0(e_i) = \frac{\sum_{\alpha \in \{x \mid author(x,n)\}} \mathbb{A}_h^\alpha(e_i)}{|\{x \mid author(x,n)\}|} \tag{14}$$

where $author(x, n)$ specifies that x is an author of n, and $\mathbb{P}_n^0(e_i)$ represents the direct opinion at node n.

Step 3: Propagating opinions Each direct opinion is then propagated within the graph following the propagation algorithm presented in this paper and summarised by Algorithm 1.[7]

Step 4: Analysing results Recall that the goal of the experiment is to verify that the final opinions resulting from the propagation mechanism do not contradict with the initial direct opinions. In our example, initial opinions have been based on authors' h-indexes. Hence, one way to analyse the results of the propagation is to compare, for each author α, the author's h-index based opinion $\mathbb{A}_h^\alpha(e_i)$ generated by equation 12 to the final author's opinion $\mathbb{A}_f^\alpha(e_i)$ resulting from the propagation algorithm, where $\mathbb{A}_f^\alpha(e_i)$ is an aggregation of the opinions on the author's papers which have been obtained after the propagation. The aggregation is calculated accordingly:

$$\mathbb{A}_f^\alpha(e_i) = \frac{\sum_{n \in \{x \mid author(\alpha,x)\}} \mathbb{P}_n^t(e_i)}{|\{x \mid author(\alpha, x)\}|} \tag{15}$$

The difference between both opinions $\Delta_{\mathbb{A}^\alpha} = |\mathbb{A}_f^\alpha(e_i) - \mathbb{A}_h^\alpha(e_i)|$ is calculated based on the Earth's movers distance [7]. Figure 1 presents the results by plotting $\Delta_{\mathbb{A}^\alpha}$ for all authors, where each author α is represented by its h-index, h_α. Note that since the Earth's movers distance is used, we have $\Delta_{\mathbb{A}^\alpha} \in [0, 1]$.

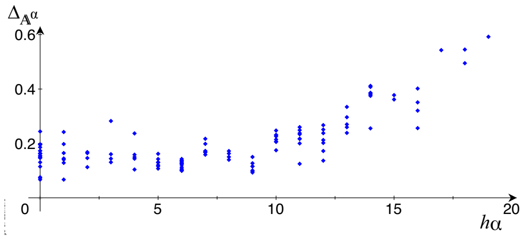

Figure 1. Difference between initial & propagated opinions

The results illustrate that an author's deduced opinion, which is a result of the propagation of opinions, remains consistent with its initial reputation. There is a minor difference between initial and final opinions, except for the case of authors with very high h-indexes, where the difference sometimes crosses the 0.5 value. In general, the difference is expected, since author's deduced opinions after propagation are influenced by their co-authors. Moreover, we notice a correlation between the difference $\Delta_{\mathbb{A}^\alpha}$ for a given h-index and the percentage of authors with that h-index. Figure 2 provides the percentage of authors associated with a given h-index. For example, we note that the percentage is minimum for the h-indexes 14, 17, 18, and 19 (1.6%, 0.8%, 1.6%, and 0.8%, respectively). Hence, the probability for authors with these h-indexes to have co-authors with considerably similar h-indexes is very low. It is for this reason that these authors have a larger $\Delta_{\mathbb{A}^\alpha}$ than others. We note that the purpose of this experiment was to validate the 'correctness' of the algorithm, and not the influence of information sources on opinions. Hence, we needed to illustrate that propagated results are coherent with initial ones. The only new information considered is the reputation of coauthors, which does influence the original reputation of authors a bit.

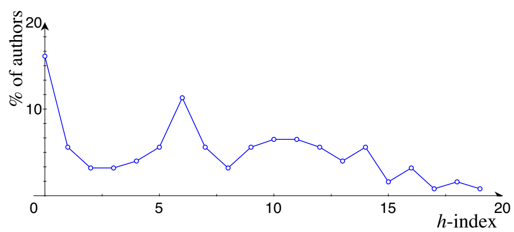

Figure 2. Percentage of authors with a given h-index

[7] We note that the opinions are added and propagated one after the other. However, the order in which the opinions are added is not very relevant since they are all added within a small interval of time. In other words, the addition of one opinion before another cannot have a huge (or even any, due to the 'grace' period determined by κ) difference in its effect on the opinions of others.

6 RELATED WORK

The research carried out by [15, 1, 3] studies the dynamics of opinion formation by focusing on the effect of social relations on how peoples' opinions may influence each other in a social network. Repage [12], ReGreT [11], and SUNNY [4] provide mechanisms for computing the confidence in a reviewer based on the social relations. These mechanisms mainly influence the reliability of the reviewer ($\Omega(r, n)$ in Equation 10), which is crucial when aggregating opinions, yet outside the scope of this paper. Similar to [12, 11, 4], additional aggregation mechanisms, such as [13, 10], may be viewed as complementary to the work presented in this paper, which mainly focuses on the propagation of opinions for one reviewer in the structural graph.

In the area of publications, SARA [9] and CiteRank [14] present algorithms on how reputation may propagate based on who is citing whom. Their reputation propagates along citation links. This paper, however, focuses on the propagation of reputation along structural links by focusing on the composition of entities and using the *part of* relation as an indication to the flow of opinions from one entity to another.

Finally, research work on ontology-based recommender systems, such as [2, 16], makes use of the clustering or classification of information and uses machine learning and data mining techniques for ranking and recommending entities. One may draw similarities between the taxonomies used by such systems and that of the structural graph of this document; although the propagation mechanism of this paper is unique in both its algorithm and semantics.

7 CONCLUSION

Opinions and ratings, which numerous trust and reputation mechanisms are based on, are not always abundant. Their abundancy differs from one field to another. For example, while tons of data may be available on Amazon or eBay, very little information is available in the publications field. This research illustrates how opinions may be deduced in areas where such information is scarce.

The main goal of this work is to provide the means for allowing an agent to deduce opinions about new entities by propagating the same agent's opinions from one entity to another. In other words, given one agent's opinions on a set of nodes of a structural graph, what can the agent deduce about its opinion concerning the remaining nodes? The true novelty of this work is in introducing the concept of the propagation of opinions along structural relations (such as *part of*).

There are several existing reputation mechanisms that focus on different types of relations, such as social relations between reviewers, correlations between attributes being assessed, etc. These mechanisms may be viewed as complementary to our proposal, since they influence the aggregation of opinions (see Equations 9 and 10) as opposed to the propagation of opinions for one reviewer along structural relations, which is the true novel aspect of this research work.

As such, existing data (e.g. the large data sets of the `www.Epinions.com` social networks) could not be useful enough for validating our propagation algorithm, which requires information about structural links. Hence, there was a need to simulate our own modest data set for validating the correctness of our algorithm.

Additionally, several sources of information may be interpreted as opinions, and hence may use the proposed propagation method. For example, in the publication example, we illustrate how h-indexes may be viewed as opinions about authors. Similarly, citations may be viewed as opinions about papers (or nodes). Propagation of citation-based opinions would then differ from the algorithms of SARA and CiteRank by allowing citation based reputation to propagate along structural relations, as opposed to citation links.

Finally, for simplicity, our algorithm has focused on static graphs only. Nevertheless, the switch to dynamic graphs is straight forward. To do so, one needs to consider the effects of adding or deleting a node or a link on the π, $\mathbb{P}$, and $\mathbb{D}$ values of the node's neighbours.

ACKNOWLEDGEMENTS

This work has been supported by both the LiquidPublications project (http://project.LiquidPub.org/), funded by the FET programme under the European Commission's FET-Open grant number 213360, and the Agreement Technologies project (http://www.agreement-technologies.org/), funded by CONSOLIDER CSD 2007-0022, INGENIO 2010.

REFERENCES

[1] V D. Blondel, J L. Guillaume, J M. Hendrickx, C. Kerchove, and R. Lambiotte, 'Local leaders in random networks', *Physical Review E*, **77**(3), 036114, (2008).

[2] Amancio Bouza, Gerald Reif, Abraham Bernstein, and Harald Gall, 'Semtree: Ontology-based decision tree algorithm for recommender systems', in *Proceedings of the Seventh International Semantic Web Conference (ISWC2008)*, (October 2008).

[3] P. Klimek, R. Lambiotte, and S. Thurner, 'Opinion formation in laggard societies', *EPL (Europhysics Letters)*, **82**(2), 28008+, (2008).

[4] Ugur Kuter and Jennifer Golbeck, 'Sunny: A new algorithm for trust inference in social networks using probabilistic confidence models', in *Proceedings of the 22nd national conference on Artificial intelligence (AAAI'07)*, pp. 1377–1382. AAAI Press, (2007).

[5] Matus Medo and Joseph Rushton Wakeling. The effect of discrete vs. continuous-valued ratings on reputation and ranking systems. `http://arxiv.org/abs/cond-mat/0407252`, 2010.

[6] University of Trento, University of Fribourg, and Consejo Superior de Investigaciones Cientificas, 'Analysis of reviewing systems', Project Deliverable D3.2, The LiquidPub Project, (May 2009).

[7] Shmuel Peleg, Michael Werman, and Hillel Rom, 'A unified approach to the change of resolution: Space and gray-level', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **11**(7), 739–742, (1989).

[8] Isaac Pinyol, Jordi Sabater-Mir, and Guifré Cuní, 'How to talk about reputation using a common ontology: From definition to implementation', in *Proceedings of the Ninth Workshop on Trust in Agent Societies. Hawaii, USA*, pp. 90–101, (2007).

[9] Filippo Radicchi, Santo Fortunato, Benjamin Markines, and Alessandro Vespignani. Diffusion of scientific credits and the ranking of scientists. `http://arxiv.org/abs/0907.1050`, Sep 2009.

[10] Matthew Richardson and Pedro Domingos, 'Building large knowledge bases by mass collaboration', in *Proceedings of the 2nd international conference on Knowledge capture (K-CAP'03)*, pp. 129–137, New York, NY, USA, (2003). ACM.

[11] Jordi Sabater and Carles Sierra, 'Social regret, a reputation model based on social relations', *ACM, SIGecom Exchanges*, **3.1**, 44–56, (2002).

[12] Jordi Sabater-Mir, Mario Paolucci, and Rosaria Contexs, 'Repage: REPutation and imAGE among limited autonomous partners', *Journal of Artificial Societies and Social Simulation (JASSS'06)*, **9**(2), (2006).

[13] Carles Sierra and John Debenham, 'Information-based reputation', in *First International Conference on Reputation: Theory and Technology*, ed., Mario Paolucci, pp. 5–19, Gargonza, Italy, (2009).

[14] Dylan Walker, Huafeng Xie, Koon-Kiu Yan, and Sergei Maslov. Ranking scientific publications using a simple model of network traffic. `http://arxiv.org/abs/physics/0612122`, 2006.

[15] Fang Wu and Bernardo A. Huberman. Social structure and opinion formation. `http://arxiv.org/abs/cond-mat/0407252`, 2004.

[16] Cai-Nicolas Ziegler, Lars Schmidt-Thieme, and Georg Lausen, 'Exploiting semantic product descriptions for recommender systems', in *Proceedings of the 2nd ACM SIGIR Semantic Web and Information Retrieval Workshop*, Sheffield, UK, (July 2004).