From *SIAM News*, Volume 41, Number 2, March 2008

Reputation Systems and Optimization

By Cristobald de Kerchove and Paul Van Dooren

The World Wide Web is making more and more use of interactive ratings collected from various users: Books are evaluated on Amazon, movies are rated on Movielens, and buyers and sellers rate one another on eBay. The list of such interactive sites is constantly growing. This clearly is a form of voting, but one in which not all raters can be expected to be fully reliable or even honest. There is nothing to stop Movielens raters from giving random ratings to movies they have not even seen, or dishonest voters from giving biased opinions that favor their "friends."

From a commercial point of view, it is clear that Web sites have a lot to earn by promoting confidence in such interactive rating systems. Ideally, they would achieve this by penalizing raters who give random or biased ratings. Two questions ought to be addressed in this context: (1) *How should the reputation of evaluated items be defined?* (2) *How can we measure the reliability of the raters?* We distinguish here between the *reputation* of an item, i.e., what is generally said or believed about its character or standing, and the *reliability* of a rater, i.e., the probability that a rater will give a fair or relevant evaluation. We illustrate these definitions in the context of eBay: User Smith has a reputation that is simply equal to the percentage of positive votes that he receives. That aggregated reputation, however, does not take into account the relevance of the votes given to Smith; the reliability of each rater needs to be taken into account.

A natural way of tackling the problem of unreliable or unfair raters in reputation systems is to weight the raters' evaluations. The resulting range of weights corresponds to a continuous validation scale for the votes. These weights change via an iterative procedure that is guaranteed to converge to a reputation score for every evaluated item and a reliability score for every rater. At each step, the reliability of every rater is calculated from the squared distance between the evaluations he gives and the reputations of the items he evaluates. The vector of these quantities is labelled the *belief divergence*. Typically, a rater diverging too much from the group should be distrusted to some extent. This definition of distance appears in [2–4], where it is used in the same context. The strength of the reputation system we describe here is that it can be applied to any static network of raters and items, and that it converges to a unique fixed point. It can also be extended to dynamical systems with time-dependent votes.

We describe our approach in some detail for a static system in which n voters are to rate m objects. For the sake of simplicity, we assume that every rater evaluates all items, with votes ranging from 0 to 1; see [1] for a complete description. In the $n \times m$ rating matrix X, X_{ij} represents the vote of rater $i \in \{1, \ldots, n\}$ for item $j \in \{1, \ldots, m\}$. The items' reputation vector r is the weighted sum of the votes

$$r = X^T \frac{f}{\sum_{i=1}^{n} f_i}. \qquad (1)$$

A rater's weight vector f depends on the discrepancy between his votes and those of others, that is, on the belief divergence

$$f_i(r) = d - \sum_{j=1}^{m} \left(X_{ij} - r_j \right)^2, \qquad (2)$$

for $i = 1, \ldots, n$ and with d a positive parameter. Clearly, when d tends to infinity, f_i tends to d for $i = 1, \ldots, n$ and r tends to the average of the votes. A decrease in d corresponds to greater discrimination of outliers. We have proved that when $d > m$—which ensures that f is always positive—there exists a unique pair of vectors r and $f(r)$ that satisfy the pair of nonlinear formulas (1) and (2). Moreover, the nonlinear iteration resulting from the recursive application of these two formulas converges to r and $f(r)$. Because this is so, the method can be applied to dynamic voting, in which the rating matrix changes over time. Clearly, with votes and Web raters constantly evolving, development of dynamic reputation systems is a necessity; see [1].

The uniqueness of the solution is established via the definition that the cost function $E(x) = -f(x)^T f(x)$ is minimized when x equals the reputation vector r. Moreover, each step given by the nonlinear iteration resulting from (1) and (2) corresponds to the steepest descent on the cost function with a particular step size. Eventually, it converges to the fixed point r of the iteration that is also the unique minimum of the cost function $E(x)$ for x in the hypercube $[0,1]^m$. This is illustrated in Figure 1 for the case of two objects.

The solution can thus be viewed not only as the fixed point of a nonlinear iteration, but also as the maximizer of the 2-norm of the raters' weight vector f. Furthermore, if the ratings are assumed to follow a normal law, i.e., i.i.d. ~ $N(r,\sigma^2)$, then the solution maximizes some global degree of confidence in the ratings, where the degree of confidence is the logarithm of the probability that the rating is correct.

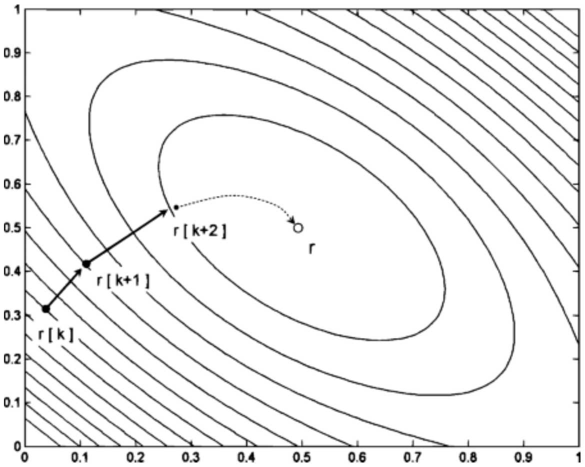

Figure 1. *Representation of steps r[k] of the nonlinear iteration in the unit box [0,1] × [0,1]. The sequence (E(r[k])) decreases with k and converges to E(r).*

To illustrate this method, we describe an experiment involving a data set (supplied by the GroupLens Research Project) of 100,000 ratings of 1682 movies by 943 users. The ratings range from 1 to 5. To test the robustness of our reputation system, we added 237 random raters to the original group; 20% of the raters thus made random evaluations. Figure 2 shows the distribution of the raters' weights at different stages of our iteration. After convergence, the random voters are seen in general to have been penalized with smaller weights. Hence, the reputations of the movies are barely perturbed by the addition of random raters. We could argue that the random raters still contribute to the final reputation vector and, with this justification, could have removed the group of raters considered outliers. But this would be to forget the goal of our reputation system: to supply a continuous range of weights to all raters.

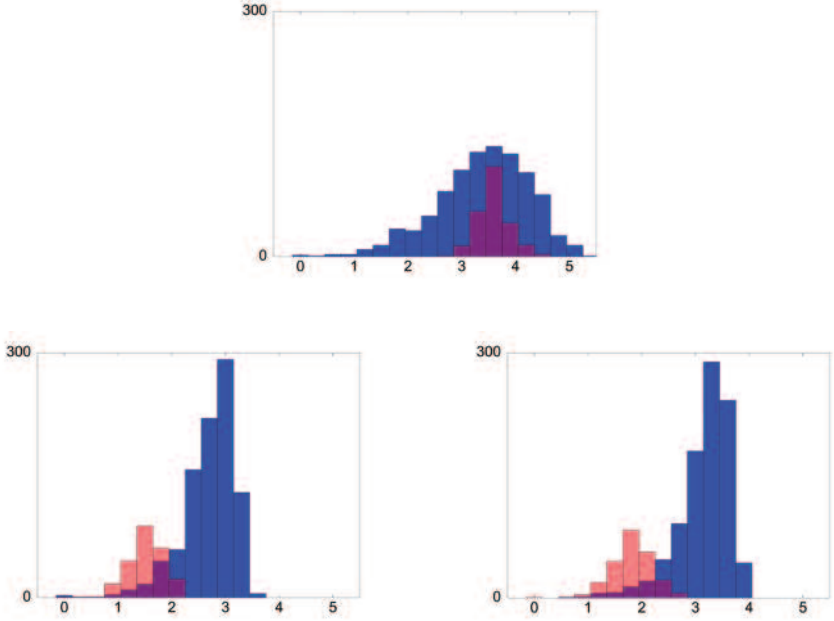

Figure 2. *X-axis: trust values for the raters, that is, the values in vector f. Y-axis: distribution of the values in vector f after one iteration (top), two iterations (left), and convergence (right). In blue: original raters; in red: random raters; in purple: overlap of the two groups of raters.*

Bringing order to voting on the Web is certainly a promising research topic that needs further investigation. Refining votes, and hence reputations, is one way to achieve that aim. In conclusion, we review what we consider the main issues in creating a reputation system: relevance of the measure, robustness against attackers of different types, applicability of the method to data of any type, and the ease with which the measure can be understood by users.

It is surely tricky to determine how relevant a measure is in the context of voting; in our case we accept the idea of *belief divergence* as a basis for calculating raters' weights, even if it implies the disqualification of marginal users. Nevertheless, the parameter *d* allows us to quantify the degree of discrimination. Moreover, the exponent in equation (2) can be made greater than 2, although the uniqueness of the fixed point is then no longer guaranteed. The presence of several fixed points, however, could be interpreted as representing different opinion groups. In that way a marginal group, if its membership is large enough, would maintain its reputation. Allowing several opinions makes sense for a movie, but providing one intrinsic value for each item should be more relevant in most contexts (such as the reputations of sellers and buyers on eBay).

A durable reputation system must be robust. Smart cheaters who work to understand the system in order to take advantage of it are certainly the greatest challenge. In our case, we proceed as simply as possible: Smart spammers would need to evaluate correctly a group of items in order to prove trustworthy, after which they could rate some target items. To significantly change the reputations of these target items, they must provide a number of coordinated evaluations larger than those of honest raters. Therefore, we can easily disqualify such cheaters by looking into coordinated ratings for one or several items. Unfortunately, this requires extra procedures, such as those used by Google to investigate the spam farms that create thousands of links to boost their PageRanks.

The increasing size of data sets on the Web creates the need for algorithms that are not too time consuming. Typically, a complexity linear in the number of votes is ideal. This is especially true for dynamic reputation systems, where the frequency of updates is high. Along with the need for efficiency, the method must be applicable to any "sparse" data set. In general, the network resulting from votes of raters for objects is not complete, i.e., each object is not evaluated by all raters. These two requirements—linear complexity and ability to handle sparse data—must be met if a reputation system is to find widespread use.

Ideally, reputation systems should also be able to cope with time-dependent data. More recent opinions may be considered more valuable than older ones, especially in such timely contexts as news, arts, fashion, etc. The system described here can easily be extended to incorporate dynamical systems with time-dependent votes (see [1]), but with the convergence to a fixed point replaced by the tracking of a time-varying trend.

Last but not least, the method must be understandable to those most directly affected: the users. Indeed, how could users have confidence in a system in which the measure of reputation looks like a black box? Our method, although more complicated than a simple eBay-like system in which all ratings have the same weight, remains relatively simple. An additional consideration is that users like transparent systems. A record of the history of votes and comments from the raters, for example, can help users to build their own opinions.

References

[1] C. de Kerchove and P. Van Dooren, *Iterative filtering for a dynamical reputation system*, ArXiv, 2007.

[2] E. Kotsovinos, P. Zerfos, N.M. Piratla, N. Cameron, and S. Agarwal, *A scalable incentive-based architecture for improving rating quality*, iTrust06, LNCS 3986, 2006, 221–236.

[3] P. Laureti, L. Moret, Y.-C. Zhang, and Y.-K. Yu, *Information filtering via iterative refinement*, EuroPhys. Lett., 75 (2006), 1006–1012.

[4] S. Zhang, Y. Ouyang, J. Ford, and F. Make, *Analysis of a low dimensional linear model under recommendation attacks*, Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 2006, 517–524.

Cristobald de Kerchove is a PhD student and Paul Van Dooren is a professor in the Department of Mathematical Engineering at Université Catholique de Louvain. They are both members of the Large Graphs and Networks Group of UCL and of the Belgian Network DYSCO.