Dirichlet Reputation Systems

Audun Jøsang
Faculty of Information Technology
Queensland University of Technology
Australia
Email: a.josang@qut.edu.au

Jochen Haller
SAP Research
Germany
Email: jochen.haller@sap.com

Abstract— **Reputation systems can be used in online markets and communities in order to stimulate quality and good behaviour as well as to sanction poor quality and bad behaviour. The basic idea is to have a mechanism for rating services on various aspects, and a way of computing reputation scores based on the ratings from many different parties. By making the reputation scores public, such systems can assist parties in deciding whether or not to use a particular service. Reputation systems represent soft security mechanisms for social control. This article presents a type of reputation system based on the Dirichlet probability distribution which is a multinomial Bayesian probability distribution. Dirichlet reputation systems represent a generalisation of the binomial Beta reputation system. The multinomial aspect of Dirichlet reputation systems means that any set of discrete rating levels can be defined. This provides great flexibility and usability, as well as a sound basis for designing reputation systems.**

I. Introduction

Reputation systems [1] represent an important type of online trust management mechanisms. Such systems, which are attracting strong interest from industry and the academic research community, are increasingly being integrated with online services and applications.

The problem of determining whether something or somebody can be trusted was not thought of as a problem when the Internet and the Web were conceived, because the community consisted of a group of users motivated by the same goals, and with strong trust in each other. Each time new and groundbreaking Internet technologies are being developed and deployed, the early adopters typically have *good intentions* because they are motivated by the desire to make the new technology successful. However, people and organisations currently engaging in Internet activities are not uniformly well intentioned in the same sense, because they are increasingly motivated by financial profit and personal gain. The result is that we are poorly prepared for controlling markets and communities where the participants' behaviour is governed by self interest, or even worse, by a combination of selfish, malicious and criminal intentions.

Reputation systems are well suited for stimulating social control within online communities or markets. The basic idea is to let parties rate each other, for example after the completion of a transaction, and use the aggregated ratings about a given party to derive a reputation score, which can assist other parties in deciding whether or not to transact with that party in the future. A natural side effect of integrating reputation systems with services and applications is that it also provides an incentive for good behaviour, and therefore tends to have a positive effect on market quality.

Reputation systems represents *soft security* mechanisms that complement traditional information security mechanisms. This was first described by Rasmussen & Jansson (1996) [2] who used the term *hard security* for traditional mechanisms like authentication and access control, and *soft security* for what they called social control mechanisms.

Reputation systems are already being used in many successful commercial online applications[3]. There is also a rapidly growing literature around trust and reputation systems, but unfortunately this activity is not very coherent. The systems being proposed are often designed from scratch, and only in very few cases are researchers building on proposals by others. The current period can therefore be seen as pioneering for online trust management.

We have previously proposed and studied binomial Bayesian reputation systems [4], [5], [6] which allow ratings to be expressed with two values, as either positive (e.g. *good*) or negative (e.g. *bad*). The disadvantage of a binomial model is that it excludes the possibility of providing ratings with graded levels such as e.g. *mediocre - bad - average - good - excellent*. Binomial models are in principle unable to distinguish between polarised ratings (i.e. many very bad and many very good ratings) and average ratings. Although it would be possible to express binary ratings with graded values by splitting a binary rating into a partially positive and partially negative rating, the mathematical treatment of this approach remains awkward with the Beta distribution.

The mathematical representation of reputation systems based on the Dirichlet distribution allows graded ratings to be directly expressed and reflected in the derived reputation scores.

This article describes reputation systems based on traditional statistical principles in the form of the Dirichlet multinomial probability distribution function. This provides a sound as well as a flexible platform for designing practical reputation systems.

[0] Appears in the proceedings of the 2nd International Conference on Availability, Reliability and Security (ARES 2007), Vienna, April 2007.

II. The Dirichlet Multinomial Model

Multinomial Bayesian reputation systems are centered around the Dirichlet multinomial probability distribution. For self-containment, we briefly outline the Dirichlet multinomial model below, and refer to [7] for more details.

A. The Dirichlet Distribution

We are interested in knowing the probability distribution over the disjoint elements of a state space. In case of a binary state space, it is determined by the Beta distribution. In the general case it is determined by the Dirichlet distribution, which describes the probability distribution over a k-component random variable $p(\theta_i),\ i = 1 \ldots k$ with sample space $[0,1]^k$, subject to the simple additivity requirement $\sum_{i=1}^{k} p(\theta_i) = 1$.

The Dirichlet distribution captures a sequence of observations of the k possible outcomes with k positive real parameters $\alpha(\theta_i),\ i = 1 \ldots k$, each corresponding to one of the possible outcomes. In order to have a compact notation we define a vector $\vec{p} = \{p(\theta_i) \mid 1 \leq i \leq k\}$ to denote the k-component random probability variable, and a vector $\vec{\alpha} = \{\alpha_i \mid 1 \leq i \leq k\}$ to denote the k-component random observation variable $[\alpha(\theta_i)]_{i=1}^{k}$.

The Dirichlet probability density function is then given by

$$f(\vec{p} \mid \vec{\alpha}) = \frac{\Gamma\left(\sum_{i=1}^{k} \alpha(\theta_i)\right)}{\prod_{i=1}^{k} \Gamma(\alpha(\theta_i))} \prod_{i=1}^{k} p(\theta_i)^{\alpha(\theta_i)-1}\ , \tag{1}$$

$$\text{where} \begin{cases} p(\theta_1), \ldots, p(\theta_k) \geq 0 \\ \sum_{i=1}^{k} p(\theta_i) = 1 \\ \alpha(\theta_1), \ldots, \alpha(\theta_k) > 0. \end{cases}$$

The probability expectation value of any of the k random variables is defined as:

$$\mathrm{E}(p(\theta_i) \mid \vec{\alpha}) = \frac{\alpha(\theta_i)}{\sum_{i=1}^{k} \alpha(\theta_i)}\ . \tag{2}$$

Because of the additivity requirement $\sum_{i=1}^{k} p(\theta_i) = 1$, the Dirichlet distribution has only $k-1$ degrees of freedom. This means that knowing $k-1$ probability variables and their density uniquely determines the last probability variable and its density.

B. A Priori Distribution for k Alternatives

Now, we come to the question of an *a priori* density function for the probabilities of k exhaustive and mutually exclusive alternatives (*e.g.* k different colours of balls in an urn). Let $p(\theta_i)$ denote the random variable describing the probability of a random sample (*e.g.* drawing a ball of a particular colour) yielding alternative i. Since $p(\theta_i)$ describes a probability, then the sample space for $(p(\theta_i))_{i=1}^{k}$ is $[0,1]^k$. Since the alternatives are exhaustive and mutually exclusive, then

$$\sum_{i=1}^{k} p(\theta_i) = 1. \tag{3}$$

In order to have a uniform distribution, the common *a proiori* parameters must be $\alpha(x_i) = 1$. Generalising the case of 2 alternatives where the probability density function is called the Beta distribution, we will take an *a priori* Dirichlet distribution over k alternatives. Since there is no reason to assume a preference for any alternative over any other alternative, then the parameters will be taken to be equal, with the result that the *a priori* probability expectation value $\mathrm{E}(p(\theta_i)) = \frac{1}{k}$ for all i. This means that the common *a priori* parameter must be $\alpha_k(x_i) = \frac{C}{k}$ for some constant C. Since it is normally required that the *a priori* distribution in case of 2 alternatives is uniform, then necessarily the *a priori* constant is defined as $C = 2$, and the common value in the case of k alternatives is:

$$\alpha_k = C/k = 2/k\ . \tag{4}$$

Should one assume an *a priori* uniform distribution over state spaces other than binary, the constant, and also the common value would be different. The *a priori* constant C will always be equal to the cardinality of the state space over which a uniform distribution is assumed. The constant $C = 2$ the emerges when a uniform distribution over binary state spaces is assumed. This means that the *a priori* distribution over state spaces larger than binary will not be uniform.

The state space cardinality provides *a priori* information about the *base rate* of an arbitrary event out of the k possible events. We define the default base rate a_k for any of the k singleton events of a state space if size k as:

$$a_k = 1/k\ . \tag{5}$$

In case no other evidence is available, the base rate alone determines the probability distribution of the events. For example in the binary case, the *a priori* probability of any of the two possible outcomes is $\frac{1}{2}$, and the probability density function is the uniform $\mathrm{Beta}(1,1)$. As more evidence becomes available, the influence of the base rate is reduced, until the evidence alone determines the probability distribution of the events. It is thus possible to separate between the *a priori* base rate expressed by a_k in the default case, and the *a posteriori* evidence over the possible events denoted as a vector $\vec{r}$. Base rates different from the default value will be described below. The total evidence $\alpha(\theta_i)$ for each singleton event θ_i can then be expressed as:

$$\alpha(\theta_i) = r(\theta_i) + Ca_k, \qquad (C: \text{ \textit{a priori} constant}). \tag{6}$$

In order to distinguish between the *a priori* default base rate, and the *a posteriori* evidence, the Dirichlet distribution can be expressed with prior information represented as a base rate vector $\vec{a}$ over the state space. This will be called the Dirichlet Distribution with Prior.

Definition 1 (Dirichlet Distribution with Prior): Let Θ be a state space consisting of k mutually disjoint elements. Let $\vec{r}$ represent the evidence vector over the elements of Θ and let $\vec{a}$ represent the base rate vector over the same elements. Then the multinomial probability density function over Θ is expressed as:

$$f(\vec{p} \mid \vec{r}, \vec{a}) = \frac{\Gamma\left(\sum_{i=1}^{k}(r(x_i)+Ca(x_i))\right)}{\prod_{i=1}^{k}\Gamma(r(x_i)+Ca(x_i))} \prod_{i=1}^{k} p(x_i)^{(r(x_i)+Ca(x_i)-1)} \; . \tag{7}$$

$$\text{where} \begin{cases} p(x_1), \ldots, p(x_k) \geq 0, \\ \sum_{i=1}^{k} p(x_i) = 1, \\ \alpha(x_1), \ldots, \alpha(x_k) > 0, \\ \sum_{i=1}^{k} a(x_i) = 1. \end{cases}$$

The expression of Eq.(7) is useful, because it allows the determination of the probability distribution over state spaces where each element can have an arbitrary base rate as long as the simple additivity principle is satisfied. The probability expectation of any of the k random probability variables can be written as:

$$\mathrm{E}(p(x_i) \mid \vec{r}, \vec{a}) = \frac{r(x_i) + Ca(x_i)}{C + \sum_{i=1}^{k} r(x_i)} \; . \tag{8}$$

The *a priori* constant C can be set to $C = 2$ when a uniform distribution over binary state spaces is assumed. Selecting a larger value for C will result in new observations having less influence over the Dirichlet distribution, and can in fact represent specific *a priori* information provided by a domain expert or by another reputation system. It can be noted that it would be unnatural to require a uniform distribution over arbitrary large state spaces because it would make the sensitivity to new evidence arbitrarily small.

For example, requiring a uniform *a priori* distribution over a state space of cardinality 100, would force the constant C to be $C = 100$. In case an event of interest has been observed 100 times, and no other event has been observed, the derived probability expectation of the event of interest will still only be about $\frac{1}{2}$, which would seem totally counterintuitive. In contrast, when a uniform distribution is assumed in the binary case, and the same observations are analysed, the derived probability expectation of the event of interest would be close to 1, as intuition would dictate.

C. Visualising Dirichlet Distributions

Visualising Dirichlet distributions is challenging because it is a density function over $k-1$ dimensions, where k is the state space cardinality. For this reason, Dirichlet distributions over ternary state spaces are the largest that can be easily visualised.

With $k = 3$, the probability distribution has 2 degrees of freedom, and the equation $p(\theta_1) + p(\theta_2) + p(\theta_3) = 1$ defines a triangular plane as illustrated in Fig.1.

In order to visualise probability density over the triangular plane, it is convenient to lay the triangular plane horizontally in the x-y plane, and visualise the density dimension along the z-axis.

Let is consider the example of an urn containing balls of the three different colours: red black and yellow (i.e. $k = 3$). Let us first assume that no other information than the cardinality is available, meaning that the default base rate is $a(x_i) = 1/3$ for all states, and $r(\text{red}) = r(\text{black}) = r(\text{yellow}) = 0$. Then Eq.(8) dictates that the expected *a priori* probability of picking a ball of any specific colour is the default base rate probability, which is $\frac{1}{3}$. The *a priori* Dirichlet density function is illustrated in Fig.2.

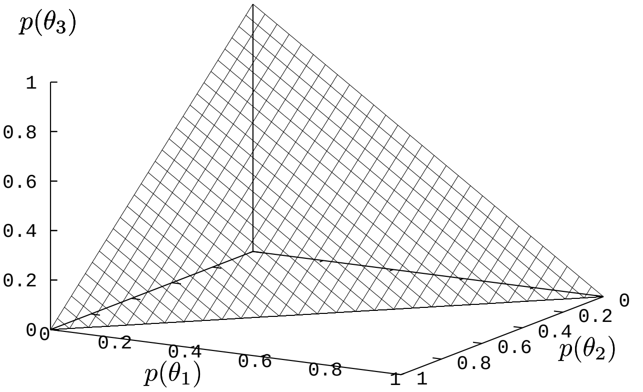

Fig. 1. Triangular plane

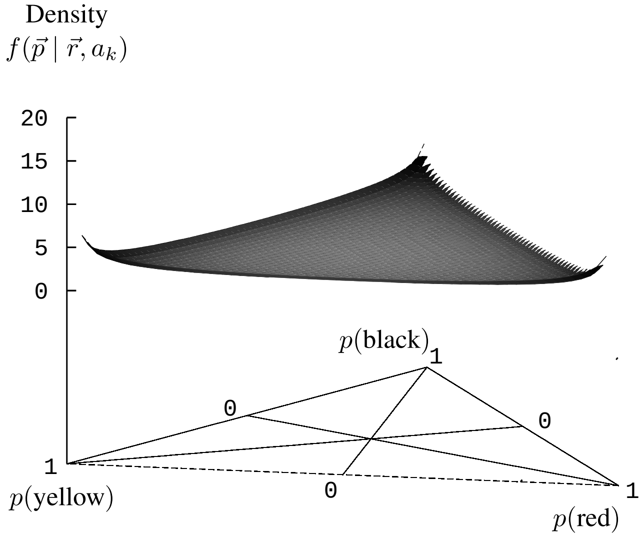

Fig. 2. Prior Dirichlet distribution in case of urn with balls of 3 different colours

Let us now assume that an observer has picked (with return) 6 red, 1 black and 1 yellow ball, i.e. $r(\text{red}) = 6$, $r(\text{black}) = 1$, $r(\text{yellow}) = 1$, then the *a posteriori* expected probability of picking a red ball can be computed as $\mathrm{E}(p(\text{red})) = \frac{2}{3}$. The *a posteriori* Dirichlet density function is illustrated in Fig.3.

III. The Dirichlet Reputation System

Multinomial Bayesian systems are based on computing reputation scores by statistical updating of Dirichlet PDF. The *a posteriori* (i.e. the updated) reputation score is computed by combining the *a priori* (i.e. previous) reputation score with the new rating. The same principle is also used for binomial Bayesian reputation systems based on the Beta distribution [8], [4], [9], [10].

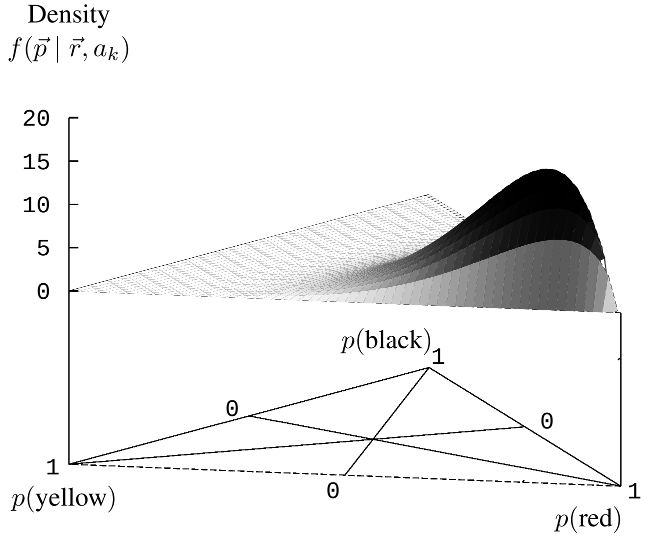

Fig. 3. *A posteriori* Dirichlet distribution after picking 6 red, 1 black and 1 yellow ball

A. Collecting Ratings

A general reputation system allows for an agent to rate another agent or service, with any level from a set of predefined rating levels. Some form of control over what and when ratings can be given is normally required, such as e.g. after a transaction has taken place, but this issue will not be discussed here. Let there be k different discrete rating levels. This translates into having a state space of cardinality k for the Dirichlet distribution. Let the rating level be indexed by i. The aggregate ratings for a particular agent y are stored as a cumulative vector, expressed as:

$$\vec{R}_y = (R_y(i) \mid i = 1 \dots k) \ . \tag{9}$$

The simplest way of updating a rating vector as a result of a new rating is by adding the newly received rating vector $\vec{r}$ to the previously stored vector $\vec{R}$. The case when old ratings are aged is described in Sec.III-B.

Each new rating of agent y by an agent x takes the form of a trivial vector $\vec{r}_y^{\,x}$ where only one element has value 1, and all other vector elements have value 0. The index i of the vector element with value 1 refers to the specific rating level.

B. Aggregating Ratings with Aging

Ratings may be aggregated by simple addition of the components (vector addition).

Agents (and in particular human agents) may change their behaviour over time, so it is desirable to give relatively greater weight to more recent ratings. This can be achieved by introducing a longevity factor $\lambda \in [0, 1]$, which controls the rate at which old ratings are aged and discounted as a function of time. With $\lambda = 0$ ratings are completely forgotten after a single time period. With $\lambda = 1$, ratings are never forgotten.

Let new ratings be collected in discrete time periods. Let sum of the ratings of a particular agent y in period t be denoted by the vector $\vec{r}_{y,t}$. More specifically, it is the sum of all ratings $\vec{r}_y^{\,x}$ of agent y by other agents x during that period, expressed by:

$$\vec{r}_{y,t} = \sum_{x \in M_{y,t}} \vec{r}_y^{\,x} \tag{10}$$

where $M_{y,t}$ is the set of all agents who rated agent y during period t.

Let the total accumulated ratings (with aging) of agent y after the time period t be denoted by $\vec{R}_{y,t}$. Then the new accumulated rating after time period $t + 1$ can be expressed as:

$$\vec{R}_{y,(t+1)} = \lambda \cdot \vec{R}_{y,t} + \vec{r}_{y,(t+1)}, \text{ where } 0 \leq \lambda \leq 1 \ . \tag{11}$$

Eq.(11) represents a recursive updating algorithm that can be executed once every period for all agents. Assuming that new ratings after n periods is received at time $t + n$, then the new rating can be computed as:

$$\vec{R}_{y,(t+n)} = \lambda^n \cdot \vec{R}_{y,t} + \vec{r}_{y,(t+n)} \ , \ \ 0 \leq \lambda \leq 1. \tag{12}$$

C. Convergence Values for Reputation Scores

The recursive algorithm of Eq.(11) makes it possible to compute convergence values for the rating vectors, as well as for reputation scores. Assuming that a particular agent receives the same ratings every period, the Eq.(11) defines a geometric series. We use the well known result of geometric series:

$$\sum_{j=0}^{\infty} \lambda^j = \frac{1}{1-\lambda} \quad \text{for } -1 < \lambda < 1 \ . \tag{13}$$

Let $\vec{r}_y$ represent the rating vector of agent y for each period. The Total accumulated rating vector after an infinite number of periods is then expressed as:

$$\vec{R}_{y,\infty} = \frac{\vec{r}_y}{1-\lambda}, \text{ where } 0 \leq \lambda < 1 \ . \tag{14}$$

Eq.(14) shows that the longevity factor determines the convergence values for the accumulated rating vectors.

D. Reputation Representation

A reputation score applies to member agents in a community M. Before any evidence is known about a particular agent y, its reputation is defined by the base rate reputation which is the same for all agents. As evidence about a particular agent is gathered, its reputation will change accordingly.

The reputation score of a multinomial system can be represented on different forms, which can be *evidence representation*, *density representation*, *multinomial probability representation*, or *point estimate representation*. Each form will be described in turn below.

1) Evidence Representation: The most direct form of representation is to simply express the aggregate evidence vector $\vec{R}_y$. The amount of ratings of level i for agent y is denoted by $R_y(i)$.

It is not necessary to express individual base rate vectors, as it will be the same for all agents.

2) Density Representation: The reputation score of an agent can be expressed as a multinomial probability density function (PDF) in the form of Eq.(7). For ternary state spaces, the PDF can be visualised as in Fig.3. Visualisation of PDFs for state spaces larger than ternary is not practical.

3) Multinomial Probability Representation: The most natural is to define the reputation score as a function of the probability expectation values of each element in the state space. The expectation value for each rating level can be computed with Eq.(8).

Let $\vec{R}$ represent a target agent's aggregate ratings. Then the vector $\vec{S}$ defined by:

$$\vec{S}_y : \left(S_y(i) = \frac{R_y(i) + Ca(i)}{C + \sum_{j=1}^{k} R_y(j)}; \mid i = 1 \ldots k \right) . \quad (15)$$

is the corresponding multinomial probability reputation score. As already stated, $C = 2$ is the value of choice, but larger value for the constant C can be chosen if a reduce influence of new evidence over the base rate is required.

The reputation score $\vec{S}$ can be interpreted like a multinomial probability measure as an indication of how a particular agent is expected to behave in future transactions. It can easily be verified that

$$\sum_{i=1}^{k} S(i) = 1 . \quad (16)$$

The multinomial reputation score can for example be visualised as columns, which would clearly indicate if ratings are polarised. Assume for example a rating scale with the 5 levels:

1) Mediocre
2) Bad
3) Average
4) Good
5) Excellent

We assume a default base rate distribution. Before any ratings have been received, the multinomial probability reputation score will be represented as in Fig.4.

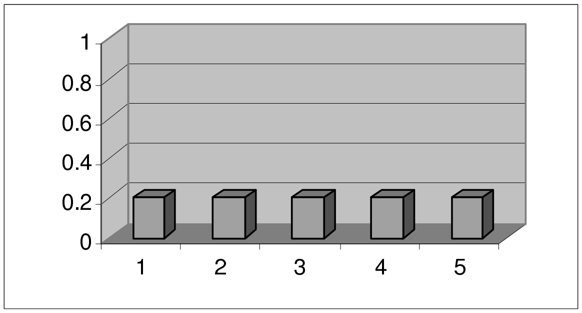

Fig. 4. Base rate probability expectation values

Let us assume that 10 ratings are received, where 5 are mediocre, and 5 are excellent. This translates into the multinomial probability reputation score of Fig.5.

Let us now instead assume that 10 *average* ratings have been received. This translates into the multinomial probability reputation score of Fig.6.

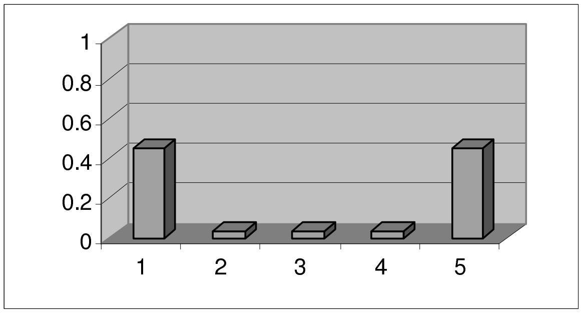

Fig. 5. Probability expectation values after 5 mediocre and 5 excellent ratings

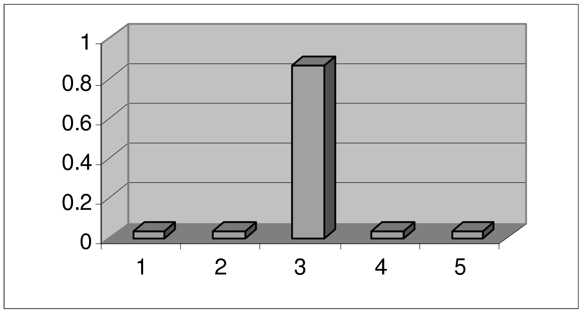

Fig. 6. Probability expectation values after 10 average ratings

In case an agent receives the same ratings every period, the reputation scores will converge to specific values. These values emerge by inserting the convergence values of Eq.(14) into Eq.(15). Let $\vec{r}_y$ be the constant ratings that agent y receives every period. The convergence score value for each rating level i can then be expressed as:

$$S_{y,\infty}(i) = \frac{\lambda \cdot r_y(i) + (1 - \lambda)Ca(i)}{(1 - \lambda)C + \lambda \sum_{j=1}^{k} r_y(j)} \quad (17)$$

In particular it can be seen that when no ratings are received (i.e. $\vec{r}_y$ is the null vector), then the convergence score value for each level is simply the base rate for that level.

4) Point Estimate Representation: While informative, the multinomial probability representation can require considerable space to be displayed on a computer screen. A more compact form can be to express the reputation score as a single value in some predefined interval. This can be done by assigning a point value ν to each rating level i, and computing the normalised weighted point estimate score σ.

Assume e.g. k different rating levels with point values evenly distributed in the range [0,1], so that $\nu(i) = \frac{i-1}{k-1}$. The point estimate reputation score is then computed as:

$$\sigma = \sum_{i=1}^{k} \nu(i) S(i) . \quad (18)$$

However, this point estimate removes information, so that for example the polarised ratings of Fig.5 are no longer visible. Let for example service y_1 receive 10 average ratings , and let service y_2 receive 5 mediocre ratings and 5 excellent. In this case, y_1 and y_2 would both have the same point estimate reputation score of 0.5, although the ratings in fact are quite different.

A point estimate in the range [0,1] can be mapped to any range, such as 1-5 stars, a percentage or a probability etc.

E. Dynamic Community Base Rates

Bootstrapping a reputation system to a stable and conservative state is important. In the framework described above, the base rate distribution $\vec{a}$ will define initial default reputation for all agents. The base rate can for example be evenly distributed, or biased towards either a negative or a positive reputation. This must be defined by those who set up the reputation system in a specific market or community.

Agents will come and go during the lifetime of a market, and it is important to be able to assign new members a reasonable base rate reputation. In the simplest case, this can be the same as the initial default reputation that was given to all agents during bootstrap.

However, it is possible to track the average reputation score of the whole community, and this can be used to set the base rate for new agents, either directly or with a certain additional bias.

Not only new agents, but also existing agents with a standing track record can get the dynamic base rate. After all, a dynamic community base rate reflects the whole community, and should therefore be applied to all the members of that community.

The aggregate reputation vector for the whole community at time t can be computed as:

$$\vec{R}_{M,t} = \sum_{y_j \in M} \vec{R}_{y,t} \tag{19}$$

This vector then needs to be normalised to a base rate vector as follows:

Definition 2 (Community Base Rate): Let $\vec{R}_{M,t}$ be an aggregate reputation vector for a whole community, and let $S_{M,t}$ be the corresponding multinomial probability reputation vector which can be computed with Eq.(15). The community base rate as a function of existing reputations at time $t+1$ is then simply expressed as the community score at time t:

$$\vec{a}_{M,(t+1)} = \vec{S}_{M,t}. \tag{20}$$

The base rate vector of Eq.(20) can be given to every new agent that joins the community. In addition, the community base rate vector can be used for every agent every time their reputation score is computed. In this way, the base rate will dynamically reflect the quality of the market at any one time.

If desirable, the base rate for new agents can be biased in either negative or positive direction in order to make it harder or easier to enter the market.

When base rates are a function of the community reputation, the expressions for convergence values with constant ratings can no longer be defined with Eq.(14).

IV. Example Visualisations of Reputation

A. Example 1: Periods of Mediocre and Excellent Ratings

In this example, agents can be rated at 5 discrete levels, with base rates evenly distributed as shown in Table IV-A.

Level	Verbal tag	Base rate
L1	Mediocre	0.2
L2	Bad	0.2
L3	Average	0.2
L4	Good	0.2
L5	Excellent	0.2

TABLE I

Example rating levels with base rates

Let an agent first be rated L1 (mediocre) every period in 5 periods, and subsequently L5 (excellent) every period in the next 5 periods. The longevity factor is set to $\lambda = 0.9$.

The multinomial probability reputation score of the agent can then be visualised as a function of level/time, as in Fig.7 below.

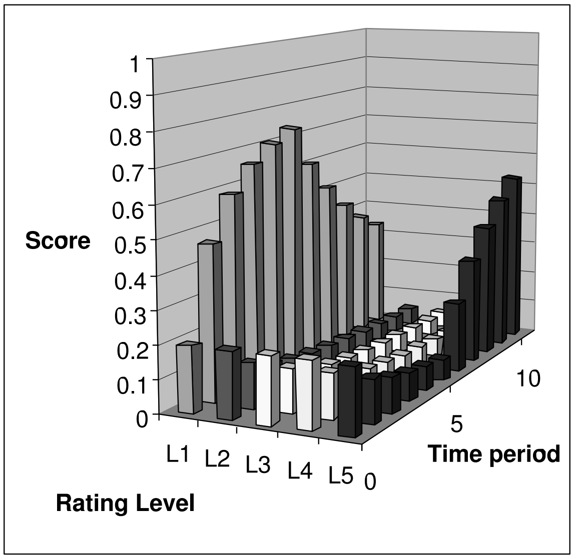

Fig. 7. Evolution of an agent's reputation after a sequence of 5 mediocre (L1) and 5 excellent (L5) ratings

It can be seen that the initial multinomial probability is evenly distributed according to the base rate, and that from then on the probability values of the different levels change as a function of the ratings. The trend during periods 1-5 is clearly different from the trend during periods 6-10.

B. Example 2: Score Convergence with Fixed and Dynamic Base Rates

In this example we will compare the convergence of reputation scores in case of fixed and in case of dynamic base rates.

We assume again a reputation system with 5 rating levels and a longevity factor $\lambda = 0.9$. In the first 10 periods the agent is rated as mediocre, and in periods 11-50 the agent is rated as excellent. It can be seen that the evolution of the score values change abruptly between period 10 and 11. This is illustrated in Fig.8

With constant base rates, the score values converge to the values that Eq.(17) predicts. With an infinite series of excellent

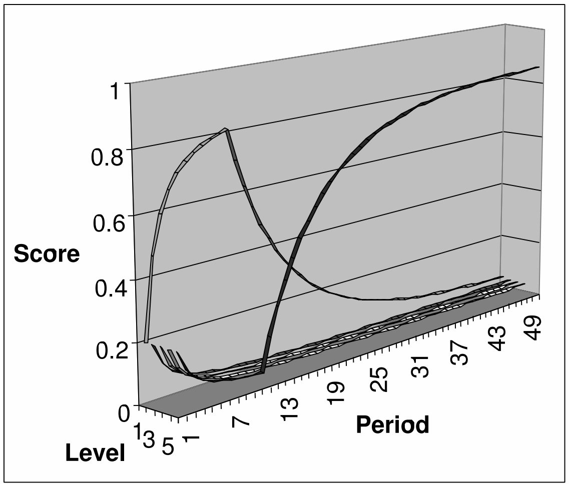

Fig. 8. Constant base rate and the evolution of an agent's reputation with a sequence of 10 mediocre (L1) and 40 excellent (L5) ratings

ratings, the score for L5 converges to $S(\text{L5}) = 0.855$, and the scores for the other levels converge to 0.036.

Now to the case of dynamic base rates. For simplicity we assume that the community consists of a single agent who is rated as before, i.e. 10 mediocre ratings followed by 40 excellent ratings. After each period, the base rate is updated to the score of the previous period. This is illustrated in Fig.9.

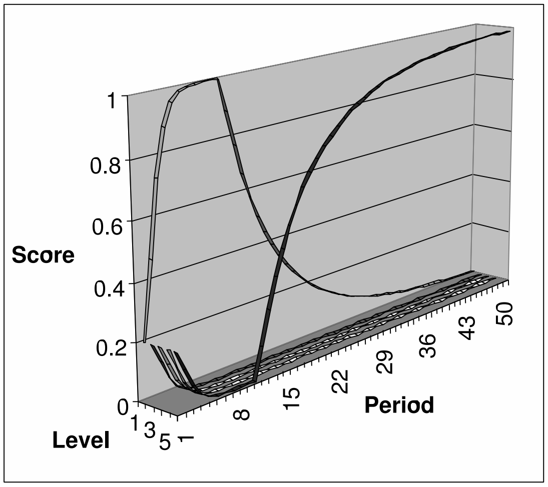

Fig. 9. Dynamic base rate and the evolution of an agent's reputation with a sequence of 10 mediocre (L1) and 40 excellent (L5) ratings

As expected an abrupt change occurs between period 10 and 11. However this time, the reputation score for L5 converges to $S(\text{L5}) = 1$, and the reputation score for the other levels converge to 0.

With a dynamic base rate, the convergence values are not determined by the longevity factor. This example unnatural because the community only consists of a single agent. However, convergence values from the longevity factor will be independent from the longevity factor for any community size.

C. Example 3: Evolution of Point Estimates

We consider the following sequence of ratings:

Periods 1 - 10:	L1 Mediocre
Periods 11 - 20:	L2 Bad
Periods 21 - 30:	L3 Average
Periods 31 - 40:	L4 Good
Periods 41 - 50:	L5 Excellent

The longevity factor is $\lambda = 0.9$ as before, and the base rate is dynamic. The evolution of the scores of each level as well as the point estimate are illustrated in Fig.10.

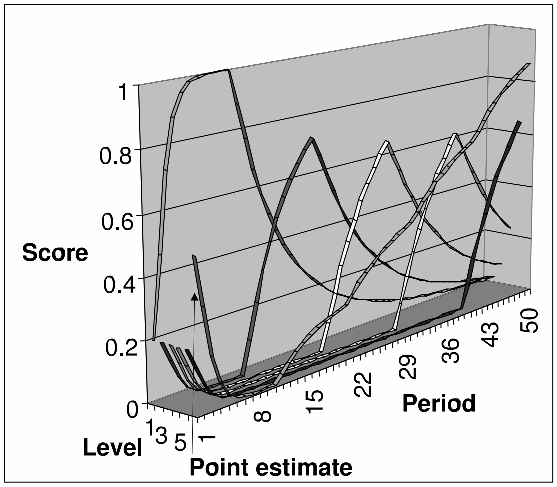

Fig. 10. Scores and point estimate during a sequence of varying ratings

In Fig.10 the multinomial reputation scores change abruptly between each sequence of 10 periods. The point estimate first drops as the score for L1 increase during the first 10 periods. After that the point estimate increases relatively smoothly during the subsequent 40 periods. Because of the dynamic base rate, the point estimate will eventually converge to 1.

V. Example Reputation System Architecture

As a simple example of how a reputation system can be implemented in a general level we describe a simple reputation toolbar which can be installed on any browser. This allows the reputation score of any Web page to be visualised to the user, as well as the user to rate Web sites and Web pages. The toolbar communicates with a centralised server which keeps the reputation vectors of all Web pages. A Web page can be rated by the user with a discrete set of different levels, as described above. This architecture is illustrated in Fig.11

While the browser is fetching a Web page, the reputation toolbar will query the reputation server about the reputation score of that Web page or Web site. This is provided as a reputation score. The user is also invited to rate the same Web site through the toolbar. This rating is sent to the reputation server, and taken into account when computing the reputation score in the future.

The computation of the reputation score is always done by the server, and only the scores are sent to the toolbar to be visualised in some form.

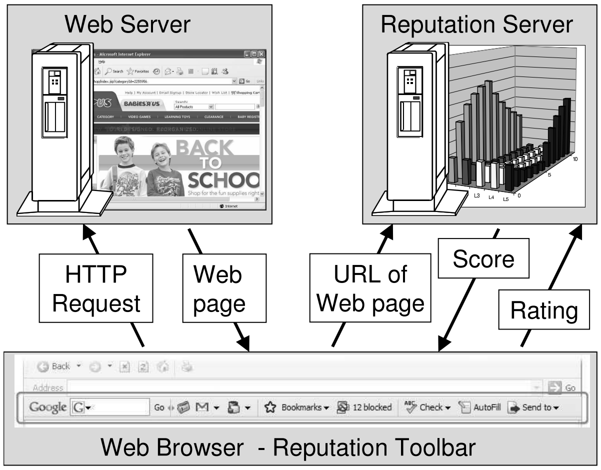

Fig. 11. Network architecture for reputation toolbars

The toolbar and the reputation server is currently being developed as part of a project on online trust management at Queensland University of Technology. The functionality of the reputation toolbar of Fig.11 can very well be integrated with a traditional search engine toolbar. The reputation scores can then be taken into account for computing ranking when searching Web resources, or can be presented as a separate score for each search query result. In the latter case, the reputation server and the search engine do not need to be co-located. The reputation score can simply be fetched as part of a search query, either by the search engine itself, or by a shell on the client machine.

The addition of a reputation system to the traditional search engine will allow the implementation of the *critical surfer model*, which represents an improvement over the current *random surfer* and the *intentional surfer* models.

The random surfer model is implemented by the traditional PageRank algorithm originally used by Google[11], and reflects the probability that a random surfer would access an given Web page. The increasing usage of `rel="nofollow"` in Web pages will have the effect that scores computed by the PageRank algorithm no longer reflect the real structure of the Web, and no longer reflect a true random surfer model. The random surfer follows any link, whereas search engines only follow those that are *not* marked by `rel="nofollow"`. A likely development is that most outgoing hyperlinks will be marked in this way in a selfish manner in order not to suffer decreased scores. The search engines will then face the problem of scarcity of cross links between Web sites, making the computed scores increasingly unreliable.

As a substitute for the hyperlinks, search engines need to use other types of evidence. A ranking can for example be based on the link to every page that people actually visit, and this is called the intentional surfer model. By encouraging people to use toolbars, search engines can get precisely that information. A toolbar provides some value-added functionality to users, such as displaying the PageRank of every page the user visits. In return for this functionality, the search engine is informed about every single Web page that the user visits.

The critical surfer model is based on critical ratings and feedback about web pages. The critical surfer model is implemented by taking expressions of approval or disapproval of particular Web pages into account when ranking Web pages in search results. This is currently not possible with existing search engine technology, but would be possible by integrating reputation systems with search engines.

VI. Discussion and Conclusion

The Dirichlet distribution provides a flexible basis for constructing reputation systems. Reputation scores can be represented as point estimates, or as multinomial probabilities. Either or both can be used, depending on the needs of the application. Although the Dirichlet distribution might seem mathematically complicated, the computation of the distribution itself never actually has to be done to accumulate ratings or to compute reputation scores. In fact, the multidimensional Dirichlet distribution itself would provide poor usability and human interpretation. Instead, reputation scores represented as point estimates and multinomial probabilities, which are simple to compute, will provide very good usability.

The strength of Bayesian reputation systems is that they provide a statistically sound basis for computing reputation scores, and we have shown that it provides a very flexible framework for constructing reputation systems.

References

[1] P. Resnick, R. Zeckhauser, R. Friedman, and K. Kuwabara, "Reputation Systems," *Communications of the ACM*, vol. 43, no. 12, pp. 45–48, December 2000.

[2] L. Rasmusson and S. Janssen, "Simulated Social Control for Secure Internet Commerce," in *Proceedings of the 1996 New Security Paradigms Workshop*, C. Meadows, Ed. ACM, 1996.

[3] A. Jøsang, R. Ismail, and C. Boyd, "A Survey of Trust and Reputation Systems for Online Service Provision," *Decision Support Systems*, vol. 43, no. 2, pp. 618–644, 2007.

[4] A. Jøsang and R. Ismail, "The Beta Reputation System," in *Proceedings of the 15th Bled Electronic Commerce Conference*, June 2002.

[5] A. Jøsang, S. Hird, and E. Faccer, "Simulating the Effect of Reputation Systems on e-Markets," in *Proceedings of the First International Conference on Trust Management (iTrust)*, P. Nixon and S. Terzis, Eds., Crete, May 2003.

[6] A. Withby, A. Jøsang, and J. Indulska, "Filtering Out Unfair Ratings in Bayesian Reputation Systems," *The Icfain Journal of Management Research*, vol. 4, no. 2, pp. 48–64, 2005.

[7] A. Gelman *et al.*, *Bayesian Data Analysis, 2nd ed.* Florida, USA: Chapman and Hall/CRC, 2004.

[8] A. Jøsang, "Trust-Based Decision Making for Electronic Transactions," in *Proceedings of the 4th Nordic Workshop on Secure Computer Systems (NORDSEC'99)*, L. Yngström and T. Svensson, Eds. Stockholm University, Sweden, 1999.

[9] L. Mui, M. Mohtashemi, C. Ang, P. Szolovits, and A. Halberstadt, "Ratings in Distributed Systems: A Bayesian Approach," in *Proceedings of the Workshop on Information Technologies and Systems (WITS)*, 2001.

[10] L. Mui, M. Mohtashemi, and A. Halberstadt, "A Computational Model of Trust and Reputation," in *Proceedings of the 35th Hawaii International Conference on System Science (HICSS)*, 2002.

[11] L. Page, S. Brin, R. Motwani, and T. Winograd, "The PageRank Citation Ranking: Bringing Order to the Web," Stanford Digital Library Technologies Project, Tech. Rep., 1998.