Towards a Theory of Trust Based Collaborative Search

Yacov Yacobi
Microsoft Research
One Microsoft Way
Redmond, WA, 98052, USA

ABSTRACT

We developed three new theoretical insights into the art of hierarchical clustering in the context of web-search. A notable example where these results may be useful is Trust Based Collaborative Search, where an active user consults agents that in the past performed a similar search. We proceed with this as an example throughout the paper, even though the results are more broadly applicable. The first result is that under plausible conditions, trust converges to the extremes, creating clusters of maximal trust. The trust between any two agents, whose initial mutual trust is not maximal, eventually vanishes. In practice there is uncertainty about data, hence we have to approximate the first result with less than maximal trust. We allow clustering tolerance equal to the uncertainty at each stage. The second result is that in the context of search, under plausible assumptions, this uncertainty converges exponentially fast as we descend the clustering tree. The third observation is that Shannon's cryptography may help estimate that uncertainty.

Keywords

Trust, browsing, collaboration, search, cryptography.

Categories and Subject Descriptors

H.3.3 [Information Search and Retrieval]: Clustering

General Terms

Theory

1. INTRODUCTION

We present a few new theoretical insights into general hierarchical clustering. The results are relevant for example, to collaborative filtering, social web search, and general hierarchical data clustering based on similarity. We use Trust Based Collaborative Search, where an active user consults agents that in the past performed a similar search as an example throughout the paper, even though the results are more broadly applicable.

The first result is that under plausible conditions that we motivate, trust converges to the extremes, creating clusters of maximal trust. The trust between any two agents, whose initial mutual trust is not maximal, eventually vanishes.

In practice there is uncertainty about data, hence we have to approximate the first result with less than maximal trust. We allow clustering tolerance equal to the uncertainty at each stage. The second result is that in the context of search, under plausible assumptions, this uncertainty converges exponentially fast as we descend the clustering tree. This conclusion follows from a novel "Hindsight" thought experiment, in which we compare the uncertainties of the real hierarchical clustering process to an imaginary process going in the opposite direction in the clustering tree (hence "hindsight"). The uncertainties of the latter are not bigger than those of the former. This leads to interesting bounds on uncertainties and hence on clustering tolerances.

The third observation is that Shannon's cryptography may help estimate that uncertainty. A query plays the role of a cryptogram, the search engine is the cryptanalyst, and the user's intention is the clear text. Shannon's "unicity distance" is the length of the search. It is needed to quantify the clustering-tolerance.

Prior art: Our trust matrices are similar (but not identical) to those of [11] and even to PageRank [2]. A broader exposition of trust-theory appears in our (yet unpublished) paper [8] (but without its application in search). Clustering using Information Theoretic entropy was done successfully in [3]. It is worth noting that when using derivatives of conditional entropy as a metric for clustering, in practice we can use only computable entropies, as defined in [22]. We do not use it here, but it is an option for future work in this area (we use a simpler metric for now). [6] pointed to the empirical significance of having smaller clusters of recommenders who highly trust each others. [9] discusses trust in the context of certification chains. [12] discusses convergence to consensus (our convergence is a simple case). The connection to cryptography relies on insight from [18] and [19].

The structure of the rest of this paper is as follows: Section 2 is a detailed overview of this work. Section 3, presents the first result: that trust converges to the extremes. Section 4 presents the "hindsight" thought experiment, that help quantify the optimal clustering tolerances. Section 5 explains how to use the results of section 4 in a hierarchical clustering algorithm.

2. OVERVIEW

2.1 Objects and their representations

Objects include pages, clusters of pages, queries, agents that in the past performed a given query, and clusters of such agents. An object is represented by a vector of probabilities of attributes. These vectors are a "common denominator" enabling the comparison of any object to any object. In this paper, the gap between two objects is the angle between the corresponding vectors . The angle is normalized to the interval [0,1]. The similarity is one minus the gap. The similarity between vectors i and j is denoted s_{ij}. We discuss vectors of similarities (not to be confused with the vectors representing objects).

2.2 Hierarchical clustering

A collaborative browsing process involves two stages of clustering: The ordinary hierarchical clustering of pages (stage-2) and hierarchical clustering of agents that performed a given query in the past, based on their subsequent browsing navigation histories (stage-1). The active user navigates the hierarchical clustering of agents (that performed "her" query in the past), ending in a cluster of pages that she then likewise navigates to a result. The path that the active user navigates is also an object. We engineer each of stage-1 and stage-2 to be a tree. The seam between the stages is not a forest; two distinct agents can go to the same cluster of pages. Within a tree, we identify a path leading from the root to any node with that node, and represent it by the vector of probabilities of attributes of that node.

In each node in the hierarchical clustering, there are two stages in the computation of similarity, s_{ij}. In both phases the gap between vectors representing objects is the angle between the vectors. In the local phase, it is computed independent of other vectors. It is followed by a transitive-averaging phase (see section 3), where the "opinions" of peers are taken into account. This is done iteratively using (similarity) matrix multiplications. These iterations should not be confused with layers (=levels) in the hierarchical trees.

2.3 A Theory of similarity

In general trust-theory [8], behavior is a random variable, and the gap between two behaviors is (a simple derivative of) their (Shannon's) conditional entropy. The relevant behavior for collaborative browsing, given initial query, is past navigations of the trees defined by the query. We therefore equate trust between agents with similarity of the vectors that represent them, and use the results of trust-theory directly [8]. However, here we use a different metric for the gap between vectors. Whereas in [8] we used conditional entropy, here we use the angle between vectors. The result of [8] is independent of the metric. For the sake of self containment we give the main result of [8] in section 3.

There is empirical evidence [6] that in trust-based reputation model for virtual communities, it pays to restrict the clusters of agents to small sets with high mutual trust. We propose and motivate a mathematical model, where this phenomenon emerges naturally. In our model, we separate trust values from their weights. We engineer the weight matrix to be stochastic (that is the natural definition of normalized weighted averaging), but unlike others (e.g. [11]), we do not demand that the overall trust matrix be stochastic. Each trust value can have any value in the Real interval [0,1]. We motivate this separation using real examples, and show that in this model, trust converges to the extremes, agreeing with and accentuating the observed phenomenon. Specifically, in our model, cliques[1] of agents of maximal mutual trust are formed, and the trust between any two agents that do not maximally trust each other, converges to zero. We offer initial practical relaxations to the model that preserve some of the theoretical flavor.

2.4 Uncertainties and tolerances

At each stage we accept members who are less than perfectly matched to other members. The gap in similarity that we allow equals our uncertainty about the data.

There are two sources of uncertainty. *Result uncertainty* and *measurement uncertainty*. There is uncertainty about results because a partially navigated path allows many optional results. Measure uncertainty is due to inaccurate and imprecise[2] probabilistic data.

2.5 Propagation of uncertainty

We distinguish gradual from instantaneous cluster buildup. Each of these methods may be used to create hierarchical clustering. In the former, cluster members that were already accepted, are not judged again under new data. They "interview" candidates under new data. In the instantaneous cluster buildup, everybody is judged at once under the current data. It is easy to see that uncertainty about similarity propagates in the gradual process. We show (in section 4.4) that under certain assumptions, there are relations between uncertainties of distinct layers in the hierarchical tree even when using instantaneous cluster buildup.

We estimate uncertainties under the following assumptions: (i) similarities between cluster members are close to 1 (it follows from general Trust Theory of section 3 that these are the interesting cases). (ii) The multiplication of two uncertainties is negligible. (iii) The gap metric is such that the uncertainty about the gap between two vectors is roughly

[1] We use the term "clique" in the theoretical part, and the term "cluster" in the practical part, where we allow tolerances >0.

[2] Accuracy relates to the gap between the expected measured value and the true value. Precision relates to the variance in the measured values.

the sum of the uncertainties in the directions of each of these vectors[3].

2.6 The Hindsight Thought Experiment

Consider a hierarchical clustering search tree. It can be either stage-2 (hierarchical clustering of pages), or stage-1 (hierarchical clustering of agents). Uncertainty declines as we descend the tree (as usual, leaves are assumed at the bottom). This is certainly true about result-uncertainty: the actively navigated path becomes more specified as we descend. However, it is plausible that measurement uncertainty does not increase either, and may decrease.

This suggests that the best approach is to do instantaneous clustering at each level in the tree. We compare this real process, where we descend the tree, to an imaginary process ascending the tree. Since data quality does not deteriorate as we descend the tree, the best approach for the imaginary process is to use gradual cluster buildup.

Thought Experiment: Create a full real path (all the way to a result) using instantaneous cluster buildup. Traverse the path from the bottom up, doing gradual cluster buildup. Analyze uncertainties for each of the processes at a given node. Find the implications of the fact that the imaginary uncertainty is not bigger than the real one.

From this we conclude an upper bound on the real uncertainty. Specifically, we show in section 4 that the uncertainty, and hence the clustering-tolerance, is at least divided by a constant bigger than 1 at each layer in the tree, compared to the layer immediately above it.

2.7 Connection to Shannon's cryptography

It turns out that to estimate the tolerance of a node we have to know its height in the tree. Theoretically, we can build the whole tree and measure it, but there may be a shortcut. Shannon's Cryptography may become useful in this process. A query plays the role of a cryptogram, the search engine is the cryptanalyst, and the user's intention is the cleartext. Shannon's unicity distance is the length of the search. From that length we can compute the height of a node.

3. TRANSITIVE TRUST

3.1 General:

We propose a slight modification to existing mathematical model [11]. In the new model (unlike the old) trust converges to the extremes, agreeing with the empirical evidence ([6]). We speculate, that this model can further improve the results. In [8] we defined the local trust using the conditional entropy of the vectors representing the behaviors of users. In this paper trust is synonymous with similarity between representative vectors, where the metric is the absolute value of the angle between vectors. The exposition in the rest of this section is independent of how we define the local trust. It is quoted verbatim from our not yet published paper [8], for the sake of self-containment of this paper.

[3]This assumption holds when the metric is the absolute value of the angle between two independent vectors.

3.2 Trust Matrix:

Consider a set $\{1, 2, ...n\}$ of agents. In a $n \times n$ trust matrix $T = (\alpha_{ij} t_{ij})$, entry (i, j) is the trust of agent i in agent j, denoted t_{ij}, weighted by some weight factor $0 \leq \alpha_{ij} \leq 1$, where for all i, $\sum_{i=1}^{n} \alpha_{ij} = 1$. α_{ij} can be interpreted as the relative relevance of the opinion of a peer (for example, a peer may be fully trusted but claim little confidence about some specific evaluation; the condition $\sum_{i=1}^{n} \alpha_{ij} = 1$ is the usual meaning of normalized weighted average). Occasionally we use the uniform weight $1/n$ as an example, but our claims hold for any convex combination. When the discrete time $\tau = 1, 2, ...$ is necessary for the explanation we write $t_{ik}(\tau)$ instead of t_{ik}. We assume that for all i and τ, $t_{ii}(\tau) = 1$. [4] It is natural to normalize the trust values to the interval $0 \leq t_{ij} \leq 1$, since we expect $0 \leq t_{ij}(\tau) t_{jk}(\tau) \leq 1$. This is similar to EigenTrust [11], but with important difference. We do not engineer T so that for all i, $\sum_{j=1}^{n} \alpha_{ij} t_{ij} = 1$ (there is no "budget" of trust; an agent who fully trusts one agent can trust other agents as well) every t_{ij} can have any value in the Real interval $[0, 1]$).

Definition 1. A maximal trust matrix is a trust matrix where every agent has trust=1 in every agent (i.e. for uniform weight, the matrix T is all $1/n$).

Interpretation of right eigenvector of T: Consider the $n \times n$ trust matrix, T of agents $1, 2, ..n$. Assume a candidate 0 to this set. The agents $1, 2, ...n$ are existing set members. Each of them "interviews" the candidate to determine a local trust value. Interviewer i has local trust value t_{i0} in candidate 0. Let $t = (t_{10}, t_{20}, ...t_{n0})^{t}$. Right multiplying Tt yields the transitive trust values after one iteration. The right eigenvector, corresponding to eigenvalue 1, is the stable transitive trust values of the existing set members in the candidate.

Interpretation of left eigenvector of T : Row i represents the trust of agent i in each of the set members, and column j represents the trust of each set member in agent j. Let $t(\tau)$ denote a row vector whose entry $j = 1, 2, ...n$, is the aggregate trust of existing set members in existing agent j at discrete time τ. Then $t(\tau+1) = t(\tau) \cdot T(\tau)$. Therefore a left eigenvector that corresponds to eigenvalue 1, is a stable trust vector representing the overall trust of the set in each of its existing members[5].

3.3 Modes of clique build-up

In the theoretical part we use the term "cliques" and in the practical part, where we allow tolerances bigger than zero, we switch to the term "clusters." Along the time axis,

[4]When trust is based on similarity, as is the case with trust-based collaborative Web search, we do not have to assume $t_{ii} = 1$. It follows from the definitions.

[5]If T is a maximal trust matrix, then after one iteration $t(\tau)$ is necessarily a consensus. Since $T^2 = T$, T is also a projection. It projects onto $\mathcal{U}$ along $\mathcal{V}$, where $\mathcal{U}$ is all the consensus vectors, and $\mathcal{V}$ is all the vectors whose components add up to zero. The minimal polynomial of T is $x^2 - x$, whose 2 roots are the eigenvalues $\lambda_0 = 0$ and $\lambda_1 = 1$. Every consensus vector is an eigenvector corresponding to λ_1.

the process is dynamic; cliques may grow, then split (when facing new data).

Definition 2. We use the term gradual clique build-up when referring to a dynamic process, using right multiplication, $t(\tau+1) = T(\tau)t(\tau)$, where current data applies to current candidates to a clique, $t(\tau)$, but existing clique members, that were accepted under older data, are not judged again under the newer data. If all the agents represented by $T(\tau)$ are evaluated using the data available at time τ, then we call it instantaneous clique build-up.

So, $T(\tau)$ has different interpretation, depending on the mode of clique build-up. In the instantaneous clique build-up, the data at time τ is used in all the entries of the matrix, and in the gradual mode of clique build-up, entries are added gradually, and once added they are not re-evaluated under newer data. the left (right) eigenvector corresponding to eigenvalue 1 is the stable solution in the instantaneous (gradual) clique build up.

Remark 1. In the gradual buildup the initial meaning of the right vector is clear, leading to a unique eigenvector. In the instantaneous buildup we define the initial value of the left vector as follows: $s = (s_{10}, s_{20}, ...s_{n0})$, where for every $1 \leq j \leq n$, s_{j0} is the similarity of agent j to the (path leading to the) father node of the cluster that we currently build. All initial values s_{ij} are obtained locally (i.e. by evaluating similarity between vectors v_i, v_j, regardless of other vectors).

3.4 The Perron-Frobenius Theory

The part of the theory that we actually use here appears e.g. in [17], Theorem 1.1, part (e). For a concise summary of the theory see also Th. 1.3.1 in Andries Brouwer's notes[6]. The books [20] and [21] are also useful. Let T be any matrix over $\mathbb{R}$ (a vector is a special case). $T > 0$ means that every entry of T is positive (the notation $T \geq 0$ should also be interpreted likewise). A matrix $T \in \mathbb{R}^{n\times n}$ is primitive if $(\exists k)[T^k > 0]$. It is irreducible if $(\forall i,j)(\exists k)[(T^k)_{ij} > 0]$ (the corresponding digraph is strongly connected, i.e. $\exists$ path from any node i to any node j). We present here only the part of the theory that we need now.

Theorem 1. (Perron-Frobenius): Let $T \in \mathbb{R}^{n\times n}$ be irreducible. There exists $\theta_0 \in \mathbb{R}$ such that $\theta_0 = \rho(T)$ is the spectral radius of T, and if $0 \leq S \leq T$ and σ is any eigenvalue of S then $|\sigma| \leq \theta_0$. Furthermore, $|\sigma| = \theta_0$ if and only if $S = T$.

Remark 2. Let T represent a max-trust clique. As such T is stochastic, hence its spectral radius is $\rho(T) = 1$. For any $S < T$, $\rho(S) < 1$ (by the above clause of the Perron-Frobenius Theorem). So, $\lim_{k\to\infty} \rho(S^k) = 0$. This is true not only when using uniform weights 1/n, but for any convex combination.

[6] http://www.win.tue.nl/~aeb/srgbk/node4.html

Remark 3. A trust matrix which is not max-trust has only the zero vector as eigenvector.

Conclusion 1. Cliques of maximal trust are formed (they may overlap). The trust between two agents that do not maximally trust each other converges to zero (because any matrix S that includes both, is $S < T$).

3.5 Practical relaxation

In practice we have to do useful things with less than perfect trust. We create near max trust clusters with clustering tolerances that we analyze in the next section. We do this initial crude analysis assuming just one iteration of the trust matrix. The motivation is that a max-trust matrix reaches a stable solution (eigen pair with eigen value 1) after one iteration.

4. QUANTIFYING UNCERTAINTY

4.1 Preliminaries

As before, let s_{ij} denote the measured similarity between vectors v_i and v_j. $s_{ij} = a_{ij} + e_{ij}$, where a_{ij} is the true similarity, and e_{ij} is the error (synonymous with uncertainty). When the level (synonymous with layer), m, in the tree is important we use the notation $s_{ij}(m)$ (and likewise with the other variables). The expected transitive uncertainty at the output of level m is denoted $\delta(m+1)$, and the expected local uncertainty of level m is denoted $e(m)$. We distinguish gradual from instantaneous transitive uncertainty, using $\widetilde{\delta}()$ for the former and $\delta()$ for the latter.

Assumptions: (i) $e_{ij} \approx e_i + e_j$ (this holds when the metric is the absolute value of the angle between independent vectors); (ii) $e_i e_j \approx 0$; (iii) $a_{ij} \approx s_{ij} \approx 1$; (iv) For large n, $\frac{1}{n}\sum_{j=1}^{n} e_j(m) \approx 0$.

4.2 Gradual buildup

Let agent 0 be the candidate, and let i be and "interviewer." For $1 \leq i \leq n$:

$$s_{i0}(m+1) = \frac{1}{n}\sum_{j=1}^{n} s_{ij}(m)s_{j0}(m).$$

In this expression, s_{ij} is the similarity between vectors v_i and v_j. The uncertainty related to variable v_j, $1 \leq j \leq n$, vanishes for large n, but constant uncertainties related to v_i and v_0 stick out. One can easily observe that $E[e_{i0}(m+1)] \approx E[e_0(m)] + E[e_i(m)]$. Here $e(m) = E[e_0(m)]$ is local error, and $E[e_i(m)]$ comes from the previous layer. So errors accumulate. A sequence of errors like this grows faster than a Fibonacci sequence, i.e. it grows exponentially fast in m. So, $\widetilde{\delta}(m) = \Theta(\exp(m))$

4.3 Instantaneous buildup

We compute expression which is similar to the one computed above for gradual buildup, with the difference that all the data is local (current), hence there is no accumulation of errors from previous layers. This implies that the error at the output is $\delta(m) = \Theta(e(m))$.

4.4 The Hindsight Thought Experiment

For a general description see the overview in 2.6. We proceed to compare the real instantaneous process indexed by τ, to the imaginary gradual process going in the opposite direction, and indexed by m. When the two processes point to the same cluster, $m + \tau = h$, where h is the height of the hierarchical clustering tree, i.e. the length of the search. However, it is more convenient to use the same index m for both (going in the imaginary direction).

Lemma 2. Going in the real direction (opposite to m), uncertainty declines (at least) exponentially fast[7].

Proof. $\widetilde{\delta}(m+1) = \Theta(\exp(m))$, and $\delta(m+1) = \Theta(e(m))$. $\widetilde{\delta}(m+1) \leq \delta(m+1)$ implies $e(m) = o(\exp(m))$. $\square$

Let $\alpha, \beta > 1, \gamma$ be constants s.t. $e(m) = e(1) \cdot \alpha \cdot \beta^m$, and $\delta(m) = \gamma \cdot e(m)$. Suppose we measure result-uncertainty using entropy, and leaves are equiprobable. If a parent cluster has n leaves, and its child cluster has $m < n$ leaves, then $H_n = \log n$. If uncertainty declines by a factor β after each layer then $\frac{H_m}{H_n} = \beta^{-1}$. It means that the logarithm of number of leaves is divided by $\beta > 1$. This is an extremely fast convergence. It may explain empirical evidence about search convergence (it happens very fast or not at all).

4.5 Shannon's Theory

We know by now that $e(m) = e(1) \cdot \alpha \cdot \beta^m$, but we do not know m. The height of the tree is $h = m+\tau$, where τ is the (known) length of the path that the active user navigated so far. If we could find h, we would know m. An estimation of h could be gleaned by creating the whole tree and computing an average distance from the leaves, but this seems impractical. Alternatively we may be able to use Shannon's theory [19]. We view the query as a cryptogram, the search engine as the cryptanalyst, and the user's intention as the cleartext. By "query" we mean extended query, which includes the initial query and the path that the active user navigated so far. Shannon defined "unicity-distance" as the amount of cryptogram needed until there is a unique solution. This is h.

5. USING THE TOLERANCES IN THE CLUSTERING ALGORITHM

The (real) algorithm involves only instantaneous clustering. At each layer of the tree, each node is built in two stages: local and transitive-averaging. For the current non-detailed description, we will only specify the clustering tolerances for each of them. Let $\theta_L(m)$ and $\theta_T(m)$ denote the local and transitive tolerances, respectively, at layer m.

Local clustering tolerance: As in the previous section, we use $e(m)$ to denote the expected local uncertainty at level m. We set $\theta_L(m) = e(m)$.

[7] Recall that the notation $f(x) = \Theta(g(x))$ means that for large enough x, the function $f(x)$ is both upper and lower bounded by $g(x)$ times some constants.

Transitive clustering tolerance: We use the notations of section 4.3. In order to get improved sieving compared to the local phase we need to set $\theta_T(m) < \gamma \cdot e(m)$. How much smaller, is an open problem (most likely to be determined empirically).

At each layer, the hierarchical-clustering algorithm does local then transitive clustering with the above tolerances. Upon descending a tree from one layer to the next, it divides the clustering tolerance by $\beta > 1$.

6. CONCLUSIONS

We presented a few theoretical insights into the art of web Search. More work is needed to quantify key parameters (α, β, γ of section 4.4). Experimentation is needed to determine the usefulness of this theory.

ACKNOWLEDGEMENT: I thank Jim Kajiya, who encouraged and supported this research, Yuval Peres, who pointed me to the Perron-Frobenius Theory, and Sara Wyckoff, for educating me about anthropological aspects of trust.

7. REFERENCES

[1] A. Kraskov, H. Stogbauer, R.G. Andrzejak, and P. Grassberger, Hierarchical Clustering Based on Mutual Information,

[2] A.N. Langville, and C.D. Meyer, Google's PageRank and Beyond: The Science of search engine ranking, Princeton University Press, 2006, ISBN - 13: 978-0-691-12202-1,

[3] R. Cilibrasi and P. M.B. Vitanyi, Clustering by compression, IEEE Trans. IT, Vol. 51, No. 4, April 2005, 1523-1545.

[4] E. Balfe, B. Smyth, An Analysis of Query Similarity in Collaborative Web Search, in D.E. Losada and J.M. Fernandez-Luna (Eds.): ECIR 2005, LNCS 3408, pp. 330-344, 2005. Springer-Verlag Berlin Heidelberg 2005.

[5] Eric J. Glover, Gary W. Flake, Steve Lawrence, Andries Kruger, David M. Pennock, William P. Birmingham, C. Lee Giles, "Improving Category Specific Web Search by Learning Query Modifications," saint,pp.23, 2001 Symposium on Applications and the Internet (SAINT'01), 2001

[6] N. Gal-Oz, E. Gudes, and D. Hendler, A Robust and Knot-Aware Trust-Based Reputation Model, In IFIPTM 2008, Joint iTrust and PST Conferences on Privacy, Trust Management, and Security, June 18-20, 2008, Trondheim, Norway.

[7] Y. Gurevich, Itay Neeman: DKAL2—A simplified and Improved Authorization Language. Feb. 2009.

[8] M. Jakubowski, R. Venkatesan, Y. Yacobi: Quantifying Trust. Submitted to FAST'09 http://www.win.tue.nl/fmweek/

[9] A. Josang, An Algebra for Assessing Trust in Certification Chains, NDSS'99.

[10] Rainer Hegselmann and Ulrich Krause, Opinion Dynamics and Bounded Confidence Models, Analysis, and Simulation, in Journal of Artificial Societies and Social Simulation (JASSS) Vol. 5, No. 3, 2002.

[11] S.D. Kamvar, M.T. Schlosser, and H. Garica_Molina: The EigenTrust Algorithm for Reputation Management in P2P Networks,

[12] W. Ren, R.W. Beard, E.M. Atkins: A Survey of Consensus Problems in Multi-Agent Coordination, 2005 American Control Conference, June 8-10, 2005. Portland, OR USA, pp. 1859-1864.

[13] C.D. Manning, P. Raghavan, and H. Schütze, An Introduction to Information Retrival, Cambridge University Press, 2008.

[14] M. Najork, S. Gollapudi, R. Panigraphy, Less is More: Sampling the Neighborhood Graph Makes SALSA Better and Faster. WSDM 2009 Barcelona, Spain.

[15] Chun-Yuan Hsiao, Chi-Jen Lu, Leonid Reyzin: Conditional Computational Entropy, or Toward Separating Pseudoentropy from Compressibility. EUROCRYPT 2007: 169-186

[16] P. Resnick and R. Zeckhauser, Trust Among Strangers in Internet Transactions: Empirical Analysis of eBay's Reputation System,

[17] E. Seneta: Non-negative Matrices and Markov Chains, Springer-Verlag, 1980, ISBN 0-387-90598-7,

[18] C. E. Shannon: A mathematical theory of communication. Bell System Technical Journal, vol. 27, pp. 379–423 and 623–656, July and October, 1948

[19] Claude E. Shannon, Communication Theory of Secrecy Systems, Bell Labs Journal 1949.

[20] Gilbert Strang, Linear Algebra and its Applications, 4th ed. Thomson, ISBN 0-03-010567-6

[21] R.S. Varga: Matrix Iterative Analysis, Prentice Hall, Inc. 1962, Library of Congress Catalog Number 62-21277.

[22] A.C. Yao: Computational Information Theory, In Complexity In Information Theory, 1988, pp. 1-15.