Summarizing Review Scores of "Unequal" Reviewers

Hady W. Lauw* Ee-Peng Lim* Ke Wang†

Abstract

A frequently encountered problem in decision making is the following *review problem*: review a large number of objects and select a small number of the best ones. An example is selecting conference papers from a large number of submissions. This problem involves two sub-problems: assigning reviewers to each object, and summarizing reviewers' scores into an overall score that supposedly reflects the quality of an object. In this paper, we address the score summarization sub-problem for the scenario where a small number of reviewers evaluate each object. Simply averaging the scores may not work as even a single reviewer could influence the average significantly. We recognize that reviewers are not necessarily on an equal ground and propose the notion of "leniency" to model this difference of reviewers. Two insights underpin our approach: (1) the "leniency" of a reviewer depends on how s/he evaluates objects as well as on how other reviewers evaluate the same set of objects, (2) the "leniency" of a reviewer and the "quality" of objects evaluated exhibit a mutual dependency relationship. These insights motivate us to develop a model that solves both "leniency" and "quality" simultaneously. We study the effectiveness of this model on a real-life dataset.

Keywords

quality, leniency, evaluation

1 Introduction

Evaluation is a fundamental activity in our life. Conferences evaluate submissions, grant agencies evaluate grant proposals, referees rate athletes, customers rate products, and so on. The importance of evaluation can be highlighted by the wrong call in the figure-skating event of the 2002 Winter Olympics (TIME, Feb. 16, 2002). According to the report, a French referee admitted to having been "pressured" into voting for the Russian team for the Gold Medal award. The damage was partially fixed by giving out a second Gold Medal to the Canadian team.

Designing a good evaluation requires solving two sub-problems: assigning a (usually small) number of reviewers to each object, and summarizing reviewers' scores into an overall score for each object. In this paper, we consider the second sub-problem of summarizing the scores of reviewers. We assume that the raw scores from reviewers are available and that reviewers may give different scores on the same object they evaluate. The goal is to determine the overall score for each object to reflect the ground truth quality of the object.

It is important to acknowledge that reviewers are not necessarily on an "equal ground" when deriving their scores, due to the difference in background, perspective, and standard. Simply averaging the scores of reviewers amounts to treating them as "equal", when in fact they are not. One quick fix is taking some weighted average as the overall score, but the weight of reviewers often is determined in a subjective manner. A moderation step may help sometimes, but only if reviewers are willing to change their scores.

If an object is evaluated by a large number of reviewers, the average score serves as an unbiased indicator of quality [9], as known from the *law of large numbers* [6]. We do not consider such evaluations. Rather, we consider evaluations where each object is assigned a "small" number of reviewers (say 3 to 7), but there could be a large number of objects to be evaluated (say hundreds or even thousands). This is typically the case if a certain expertise is required of a reviewer, such as reviewing conference submissions or grant proposals. Each reviewer gets to evaluate a small number of objects and each object gets a small and likely different set of reviewers. In this case, the simple average can be easily "manipulated" by a single reviewer.

1.1 Approach We are interested in a "data-centric approach" where an overall score is derived solely from reviewers' scores without requiring further information about reviewers (such as weighting of reviewers). Our basic assumption is that review scores can be "trusted" in that most reviewers are "honest" and exercise their best judgments, as in most practical cases. Note that "honest" reviewers may have different "biases", thus giving different scores to an object. A key idea in our approach is modeling such "biases" by the notion of "leniency", which captures the tendency of a reviewer to give a higher or lower score. Various reasons may cause

*Nanyang Technological University
†Simon Fraser University

	o_1	o_2	o_3	o_4	o_5
r_1	0.6	0.6	0.6	–	–
r_2	0.3	–	–	0.4	–
r_3	0.3	–	–	–	0.4
r_4	–	0.3	0.3	0.4	0.4
r_5	–	0.3	0.3	0.4	0.4

Figure 1: Score Data 1

	o_1	o_2	o_3	o_4	o_5
r_1	0.6	0.4	0.4	–	–
r_2	0.3	–	–	0.2	–
r_3	0.3	–	–	–	0.2
r_4	–	0.4	0.4	0.5	0.5
r_5	–	0.4	0.4	0.5	0.5

Figure 2: Score Data 2

a different degree of leniency. It is not our intention to identify these reasons. Instead, we focus on the impact of leniency on the reviewers' scores. Two insights about leniency underlie our approach.

Insight I: Networked Approach to Leniency. To determine the leniency of a reviewer, it is important to consider all the objects evaluated by the reviewer and how other reviewers evaluate those objects. A reviewer is lenient if some trend of giving a higher or lower score is observed across the evaluation of all such objects. The following example illustrates this point.

EXAMPLE 1. *Figures 1 and 2 show two sets of scores under the same reviewer/object assignment. Each entry* $e_{ij} \in [0, 1]$ *denotes the score by a reviewer* r_i *on an object* o_j*, with a dash "-" denoting that the reviewer has not evaluated the object. In both datasets,* o_1 *gets exactly the same scores from the same reviewers, i.e., 0.6, 0.3 and 0.3 by* r_1*,* r_2 *and* r_3*. Therefore, with the averaging approach,* o_1 *will receive the same overall score in the two evaluations. We claim, however, that it is more reasonable that* o_1 *receives a lower overall score in the first evaluation than in the second evaluation.*

Consider Figure 1 first. r_1 *gives 0.6 and* r_2 *and* r_3 *give 0.3 on* o_1*. We want to know whether this difference is because* r_1*'s score is too high or because* r_2 *and* r_3*'s scores are too low. Instead of considering only* o_1*, we consider how these reviewers evaluate other objects.* r_2 *and* r_3 *tend to agree with their co-reviewers on most objects evaluated (i.e.,* o_4 *and* o_5*), whereas* r_1 *tends to give a higher score than her co-reviewers (i.e., on* o_1*,* o_2*,* o_3*). While it may be possible that* r_1 *is right and all her co-reviewers are wrong, this possibility diminishes if these tendencies are observed on a larger scale. It makes more sense to trust* r_2 *and* r_3 *more because they have shown more consistency with their co-reviewers. Therefore, we believe that* r_1 *is probably too lenient on* o_1*.*

In Figure 2, r_1 *agrees with most co-reviewers, but* r_2 *and* r_3 *give a lower score than their co-reviewers. In this case, it makes more sense to trust* r_1 *more and believe that* r_2 *and* r_3*'s scores on* o_1 *are probably too low. Interestingly, this time we trust the minority (i.e.,* r_1*) instead of the majority (i.e.,* r_2 *and* r_3*)!*

Insight II: Mutual Dependency of Quality and Leniency. The quality of an object, as represented by the overall score, and the leniency of a reviewer are mutually dependent on each other. On one hand, in order to determine the quality of an object, we need to know the leniency of the object's reviewers, so that this leniency can be taken into account. On the other hand, in order to determine the leniency of a reviewer, we need to know the quality of the objects evaluated by the reviewer as the baseline of measuring the leniency.

Our approach associates leniency with a reviewer and quality with an object. It is true that a reviewer could be more lenient on one object and less lenient on another. The leniency considered here is the overall leniency of a reviewer after considering all objects reviewed. Such overall leniency makes as much sense as the overall score of an object where different reviewers may give different scores to the same object.

The rest of the paper will be organized as follows. In Section 2, we give an overview of the related work. In Section 3, we describe a framework called the *Differential Model* to solve leniency and quality based on the above insights. This is followed in Section 4 by a brief outline of two types of solutions. Section 5 reports the experimental results on a real-life dataset. Section 6 concludes this paper.

2 Related Work

Another approach to remove the bias of evaluation is to standardize reviewers' evaluation. The idea is to normalize all the scores given by a reviewer and determine the "range" used by this reviewer. The normalized scores could then be re-calibrated to a common standard [2]. To apply this approach, however, each object must be evaluated by *every* reviewer because the "ranges" of reviewers are comparable only if they have been derived from the same set of objects. This condition is not satisfied by the type of evaluations considered in our work.

The score summarization problem is related but orthogonal to the reviewer assignment problem [3, 4, 7]. The latter problem focuses on the selection of reviewers, which takes place before evaluation, and thus does not consider review scores. Our work recognizes the fact that the reviewer/object assignment is hardly perfect,

especially for a large-scale evaluation, and seeks to improve the evaluation by considering the difference of reviewers at the stage of summarizing reviewers' scores.

In an earlier work [8], we investigate how deviation in review scores may reveal the bias on the part of reviewers and the controversy on the part of objects. While reviewers may demonstrate bias due to their leniency, the approach of explicitly modeling leniency is first introduced in this paper. The controversy of an object is an orthogonal concept to the quality of an object in that a controversial object can be of either high quality or low quality, and therefore cannot be used to measure the quality of an object.

3 Differential Model

Score data is the collection of raw scores $e_{ij} \in [0,1]$ that reviewer r_i assigns to object o_j. It is represented as an adjacency matrix as shown in Figures 1 and 2.

Given a score data, we seek to determine the leniency l_i of each reviewer r_i and the quality q_j of each object o_j. Their values are interpreted as follows. Higher q_j implies higher quality. A reviewer with $l_i > 0$ tends to inflate her score. A reviewer with $l_i < 0$ tends to deflate her score. $l_i = 0$ implies an unbiased reviewer. Here, we describe our approach to determine l_i and q_j.

Suppose that we know the quality q_j, we determine the leniency l_i as in Equation 3.1. For each object o_j evaluated by r_i, $\frac{e_{ij}-q_j}{e_{ij}}$ tells the degree to which the given score e_{ij} has been inflated (or deflated) with respect to the object's quality q_j, normalized by e_{ij}. The case of $e_{ij} = 0$ can be avoided by replacing such e_{ij} with an appropriately small non-zero value. To determine l_i, we aggregate $\frac{e_{ij}-q_j}{e_{ij}}$ over the set of objects that r_i has evaluated. Here, we use the *average* aggregation function, which takes into account the reviewer's "behavior" across all objects evaluated. Consequently, reviewers who tend to assign higher scores would have $l_i > 0$, while those who tend to assign lower scores would have $l_i < 0$.

$$l_i = \underset{j}{\mathcal{A}vg} \left(\frac{e_{ij} - q_j}{e_{ij}} \right) \tag{3.1}$$

On the other hand, suppose that we know the leniency l_i, we can compute q_j as in Equation 3.2. For each reviewer r_i of o_j, we determine $e_{ij} \cdot (1 - \alpha \cdot l_i)$, which is the score e_{ij} adjusted to compensate for the leniency l_i. The adjustment is proportional to the base score e_{ij}. The user-determined compensation scaling factor $\alpha \in [0,1]$ controls the extent to which the scores may be adjusted to compensate for leniency. In general, a larger α would lead to a larger compensation. The score of a reviewer with $l_i > 0$ is adjusted downwards, whereas if $l_i < 0$ the score is adjusted upwards. By compensating, we attempt to estimate the score that would have been assigned by a lenient reviewer had s/he been unbiased. We aggregate the adjusted scores over all reviewers of o_j. Again, the *average* function is assumed.

$$q_j = \underset{i}{\mathcal{A}vg} \left[e_{ij} \cdot (1 - \alpha \cdot l_i) \right] \tag{3.2}$$

Since knowing l_i requires knowing q_j and vice versa, the mutually-dependent variables l_i and q_j must be determined simultaneously. Furthermore, due to the inter-connectivity of reviewers and objects, we also have to consider the leniency of all reviewers and the quality of all objects together. The pair of Equations 3.1 and 3.2 are called *Differential Model*, to emphasize our modeling of unequal reviewers.

The averaging approach, or the *Naive* approach, ignores leniency and equates an object's quality to the average of its review scores. *Naive* is a special case of the *Differential Model*. When $\alpha = 0$, no adjustment for leniency is done. This reduces *Differential*'s Equation 3.2 (to determine quality) to *Naive*'s Equation 3.3.

$$q_j = \underset{i}{\mathcal{A}vg}\ e_{ij} \tag{3.3}$$

	Quality		Leniency	
	Naive	*Differential*	*Differential*	
q_1	0.40	0.38	l_1	0.36
q_2	0.40	0.38	l_2	−0.18
q_3	0.40	0.38	l_3	−0.18
q_4	0.40	0.44	l_4	−0.18
q_5	0.40	0.44	l_5	−0.18

Table 1: Quality and Leniency for Score Data 1

EXAMPLE 2. *Table 1 and Table 2 display the quality computed using* Naive*, as well as the quality and leniency computed using* Differential*, for Figure 1 and Figure 2 respectively. For this example,* Differential *uses* $\alpha = 0.5$. Naive *ignores leniency and gives all the objects the same quality of* 0.40. *We claim that the different rankings of objects by* Differential *are more intuitive.*

Consider Table 1 (for Figure 1) first. While Naive *considers all objects to have the same quality,* Differential *considers* o_4 *and* o_5 *to have higher quality than* o_1*,* o_2*, and* o_3*. This is because* r_1 *— who has given higher raw scores than her co-reviewers on* o_1*,* o_2*, and* o_3 *— has highly positive leniency (*0.36*).* r_1*'s co-reviewers (*r_2*,* r_3 *on* o_1 *and* r_4*,* r_5 *on* o_2*,* o_3*) have negative leniency (*−0.18*), which are smaller in magnitude than* r_1*'s leniency.* Differential *adjusts the scores for leniency, with greater adjustment applied on* r_1*'s scores, resulting in the net lower quality of* o_1*,* o_2*, and* o_3 *(*0.38 *by* Differential*). Since the reviewers of* o_4 *(*r_2*,* r_4*,* r_5*) and* o_5 *(*r_3*,*

	Quality		Leniency	
	Naive	*Differential*	*Differential*	
q_1	0.40	0.47	l_1	0.11
q_2	0.40	0.38	l_2	−0.79
q_3	0.40	0.38	l_3	−0.79
q_4	0.40	0.41	l_4	0.12
q_5	0.40	0.41	l_5	0.12

Table 2: Quality and Leniency for Score Data 2

*r_4, r_5) all have negative leniency, after adjustment, o_4 and o_5 have higher quality (*0.44 *by* Differential*).*

In Table 2 (for Figure 2), it is r_2 and r_3 that are given highly negative leniency (−0.79*). The upward adjustment of r_2 and r_3's scores by* Differential *due to negative leniency pulls up the quality of objects evaluated by r_2 or r_3 ($q_1 = 0.47$, $q_4 = 0.41$, $q_5 = 0.41$ by* Differential*). Since the reviewers of o_2 and o_3 (r_1, r_4, r_5) have positive leniency, after adjustment, o_2 and o_3 have lower quality (*0.38 *by* Differential*).*

4 Exact and Ranked Solutions

We present two solutions to the *Differential Model* of leniency and quality. The *Exact* solution, treats the model as a linear system of equations to be solved for exact values of leniency and quality. The *Ranked* solution, treats the model as a ranking problem to be solved for rankings by leniency and quality. Each solution gives rise to an independent outcome, which may not be identical. *Ranked* solution is valuable as sometimes no *Exact* solution exists, but a unique ranking still exists and knowing the ranking suffices for the application. For instance, the evaluation objective may be to identify the highest quality objects (e.g., a PC chair interested in accepting the best papers).

By substituting Equation 3.1 of various r_i's into Equation 3.2 of various o_j's, and then representing the resulting equations in terms of q_j's as matrices, we get a recursive matrix equation in the general form of Equation 4.4. Q is a vector whose elements are the various q_j's, while X and Y are constants. Subsequently, we distinguish between the *Exact* solution, which solves Equation 4.4 as a linear system of equations, and the *Ranked* solution, which derives a unique ranking from an eigenvector equation modified from Equation 4.4.

$$Q = X + YQ \tag{4.4}$$

4.1 Exact Solution The *Exact* solution is the unique value of Q satisfying Equation 4.4. The matrix Equation 4.4 stands for a system of linear equations in terms of various q_j's. From linear algebra [1], we know that such a system may be in one of three situations:

Case 1: consistent and uniquely determined, there is one unique solution, which is the intersection point of the linear equations

Case 2: consistent and underdetermined, there are infinitely many solutions, which lie on the line or plane where the linear equations meet

Case 3: inconsistent, there is no solution as the linear equations do not meet

Exact solution exists only under Case 1, which produces a unique Q. This solution is given in Equation 4.5, where I is the identity matrix. For the solution to be unique, $(I - Y)$ must be invertible, which is true if and only if $det(I - Y) \neq 0$. Once Q is solved, the respective q_j's are used to solve l_i using Equation 3.1.

$$Q = (I - Y)^{-1}X \tag{4.5}$$

Failing the test $det(I - Y) \neq 0$, Equation 4.4 falls under Case 2 or Case 3. *Exact* solution does not exist for either Case 2 or Case 3.

4.2 Ranked Solution For the *Ranked* solution, we are only interested in the ranking by quality (and by leniency). Thus the value of each q_j matters less than the relative ranking among the various q_j's. We could derive such a ranking from Equation 4.6, which is modified from Equation 4.4 by adding a non-zero, real-valued scalar variable λ. Intuitively, Equation 4.6 says that any q_j (in left-hand side Q) could be expressed in terms of the quality of other objects (in right-hand side Q), after rescaling by λ. In other words, the Q that satisfies Equation 4.6 would preserve the relative ratio among q_j elements (and the ranking by quality).

$$\lambda\, Q = X + YQ \tag{4.6}$$

Equation 4.6 can be further modified into an eigenvector equation in terms of Q, which can be solved by iterative methods [1]. Subject to convergence conditions [5], Q will converge almost independently of its initial value. A similar formulation and convergence have been previously attempted in the work on PageRank [10].

In summary, we have introduced two independent solution methods derived from similar, but slightly different matrix equations. *Exact* solution produces exact values of quality/leniency, which can also be used for ranking, while *Ranked* solution produces only the rankings by quality/leniency.

5 Experiments

The objective of experiments is to verify the effectiveness of the proposed model. The dataset used is a

real-life dataset obtained from the product review site Epinions (www.epinions.com). We analyze the quality rankings to see how *Naive* and *Differential* (*Exact* and *Ranked* solutions) differ from one another. We also showcase how the results of *Differential* are more intuitive than *Naive*'s by showing a specific example. As there are no pre-determined quality and leniency, a model's "effectiveness" can only be judged based on intuition as illustrated in the early examples.

As the focus of experiments is on the effectiveness of the proposed model, we will not examine computational complexity further. For the data sizes used in these experiments, computations are generally very fast, involving a small number of iterations spanning seconds.

On this dataset, we apply *Naive* and *Differential* models. For *Differential*, we derive both *Exact* and *Ranked* solutions, setting $\alpha = 0.5$. For this α setting, both *Exact* and *Ranked* solutions exist. Hence, the three comparative solutions are *Naive*, *Exact*, and *Ranked*.

5.1 Epinions Dataset The Epinions dataset was acquired by crawling pages from the site for two days (Nov. 28–29, 2005), starting from a seed page[1]. The crawled pages represented a subset of products (objects), reviewers, and scores available from Epinions. Reviewers gave scores on the scale of 1 to 5 stars. These scores were rescaled to the range from 0.2 to 1 by division by 5 (e.g., 1 star is 0.2). We pruned the data such that each object had at least 3 reviewers and each reviewer evaluated at least 3 objects. This removed the occasional reviewers/objects and gave greater support when inferring the "behavior" of reviewers/objects. Epinions assigned each product a category. We retained only the *videos* category, which was among the most popular categories and was still of significant size after pruning. The final dataset has 172 reviewers, 165 objects, and 1157 scores. This dataset has relatively few reviewers evaluating each object (3 to 14 reviewers, with an average of 7 reviewers), which makes it suitable for the problem being considered (see Section 1).

5.1.1 Rank Comparison The objects and reviewers are ranked in descending order of quality and leniency respectively. Same values share the same rank. For example, if the three highest values are the same, they share rank 1, and the next highest is of rank 4. Here, we study how differently our model is from *Naive* at ranking by quality. We also look at whether *Exact* or *Ranked* solutions make a difference for this dataset.

[1] http://www.epinions.com/member/community_lists.html/show_~6/display_list_~true/vert_~3321654/year_~1900/sec_~community_member_list/pp_~1/pa_~1

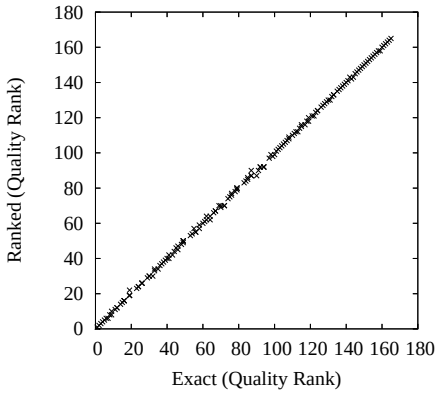

Figure 3: Quality Rank Scatterplot (*Exact* vs. *Ranked*)

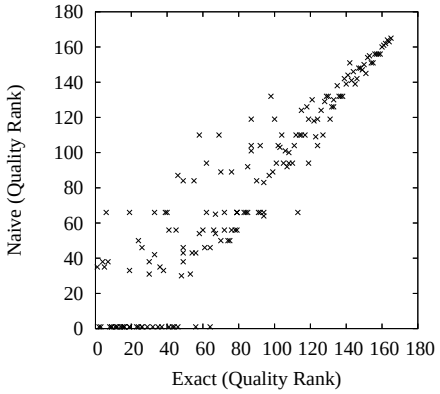

Figure 4: Quality Rank Scatterplot (*Exact* vs. *Naive*)

Exact vs. Ranked For this experiment, the *Exact* and *Ranked* solutions produce very similar rankings. Figure 3 shows a scatterplot of quality ranks. Each point on the scatterplot represents an object. Values on the x-axis are ranks given by *Exact* and values on the y-axis are ranks given by *Ranked*. If most objects are given very similar ranks, the points would line up along the diagonal, as in this figure. As leniency is mutually dependent on quality, it follows that the rankings by leniency would also be similar. Due to the similarity between *Exact* and *Ranked* here, we subsequently use only the *Exact* solution to compare against *Naive*.

Exact vs. Naive Next, we look at how *Exact* and *Naive* rank objects differently. Figure 4 gives the scatterplot of quality ranks for *Exact* vs. *Naive*. From this figure, we make these observations:

- There are significant variances around the diagonal, which means *Exact* and *Naive* give quite different ranks to several objects. For instance, there are 29 (17.6%) objects sharing rank 1 by *Naive*, which

object		Quality (Rank)	
		Naive	*Exact*
mu1016864		0.97 (35)	1.09 (1)
reviewers	e_{ij}	**Leniency (Rank)**	
			Exact
cripper	1.0		-0.05 (97)
edmaidel	1.0		-0.88 (170)
george_chabot	1.0		-0.19 (137)
icariusrex	1.0		-0.05 (95)
janesbit1	1.0		-0.11 (115)
matthewn	1.0		-0.10 (110)
munkus	1.0		-0.04 (94)
ninput	0.8		-0.61 (167)

Table 3: Profile of Object *mu1016864*

are given ranks ranging from 2 to 64 by *Exact*. This shows that our model can differentiate the quality of objects which are highly competitive, e.g., selecting the best papers at conferences or selecting the top proposals for funding.

- There are greater rank differences at higher ranks than at lower ranks. For this dataset, there is a lower density of objects at the lower end of the quality spectrum, which makes it harder for a low quality object to displace a higher- or lower-ranked object even after adjustment for leniency.

5.1.2 Case Example The previous comparison has focused on the overall difference in rankings. Here, with the help of a specific example, we showcase how our model is more intuitive than *Naive*.

Table 3 shows the profile of the object *mu1016864*. It shows the quality (and rank) given by *Naive* and *Exact*. To understand how the object is ranked, the table lists the object's reviewers, the scores they give (e_{ij}), and for *Exact* also their leniency (and ranks). We claim that intuitively the higher quality rank given by *Exact* (rank 1) is more justified than the rank given by *Naive* (rank 35). *Exact* shows that all the reviewers are negatively lenient. Particularly, *edmaidel* (rank 170) and *ninput* (rank 167) are among the least lenient (out of 172 reviewers). *Naive* does not take leniency into account, while *Exact* does. *Exact* adjusts these negatively lenient reviewers' scores upwards, which leads to the higher quality (and rank) given to this object.

To summarize, these experiments have shown that our model produces results that are different from and more intuitive than *Naive*.

6 Conclusion

In this paper, we propose the *Differential Model* to address the score summarization problem, for the scenario where each object is evaluated by relatively few reviewers. The main idea is to model the leniency of reviewers to compensate for the under- or over-estimation of quality by reviewers. Through experiments with a real-life dataset, we verify that this model is indeed more effective than the averaging approach (*Naive*).

Several avenues exist for future work. Since there are no pre-determined quality and leniency, the "performance" measurement in this paper mostly rely on intuition. Conducting a user evaluation of the results might further verify the effectiveness of the proposed model. The problem addressed here also touches aspects beyond the scope of computer science. For instance, it would be interesting to verify if the compensation mode adopted in this paper is consistent with the psychology of reviewers as studied in the social sciences.

References

[1] H. Anton and C. Rorres. *Elementary Linear Algebra with Applications*. John Wiley & Sons, Inc., 1987.

[2] H. R. Arkes. The nonuse of psychological research at two federal agencies. *Psychological Science*, 14(1):1–6, 2003.

[3] S. T. Dumais and J. Nielsen. Automating the assignment of submitted manuscripts to reviewers. In *Proceedings of the 15th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 233–244, 1992.

[4] J. Geller. Challenge: How IJCAI 1999 can prove the value of AI by using AI. In *Proceedings of the 15th International Joint Conference on Artificial Intelligence*, pages 55–61, 1997.

[5] G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, 3rd edition, 1996.

[6] G. R. Grimmett and D. R. Stirzaker. *Probability and Random Processes*. Oxford University Press, 1982.

[7] S. Hettich and M. J. Pazzani. Mining for proposal reviewers: Lessons learned at the National Science Foundation. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 862–871, 2006.

[8] H. W. Lauw, E.-P. Lim, and K. Wang. Bias and controversy: Beyond the statistical deviation. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 625–630, 2006.

[9] H. W. Marsh and L. A. Roche. Making students' evalutions of teaching effectiveness effective. *American Psychologist*, 52(11):1187–1197, 1997.

[10] L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank citation ranking: Bringing order to the Web. In *Stanford Digital Library Technologies Project*, 1998.