Ranking Comments on the Social Web

Chiao-Fang Hsu, Elham Khabiri, and James Caverlee
Department of Computer Science and Engineering
Texas A&M University
College Station, TX 77845 USA
{drakihsu, khabiri, caverlee}@cse.tamu.edu

***Abstract*—**

We study how an online community perceives the relative quality of its own user-contributed content, which has important implications for the successful self-regulation and growth of the Social Web in the presence of increasing spam and a flood of Social Web metadata. We propose and evaluate a machine learning-based approach for ranking comments on the Social Web based on the community's expressed preferences, which can be used to promote high-quality comments and filter out low-quality comments. We study several factors impacting community preference, including the contributor's reputation and community activity level, as well as the complexity and richness of the comment. Through experiments, we find that the proposed approach results in significant improvement in ranking quality versus alternative approaches.

I. Introduction

The Social Web is one of the early successes in the emerging social computing paradigm. Prominent Social Web examples include large-scale information sharing communities (e.g., Wikipedia), social media sites (e.g., YouTube), and web-based social networks (e.g., Facebook), each centered around user-contributed content and community-based information sharing.

One of the key features driving the growth and success of the Social Web is the large-scale user participation in content annotation via user-contributed tags, ratings, and comments. While tags and ratings provide succinct metadata about Social Web content (e.g., a tag is often a single keyword), user-contributed comments offer the promise of a rich source of contextual information about Social Web content but in a potentially "messier" form, considering the wide variability in quality, style, and substance of comments generated by a legion of Social Web participants.

Our overall research goal is to leverage these comments as a form of social collective intelligence for enhanced information organization, summarization, content retrieval, and visualization. Indeed, several recent research efforts have begun steps in this direction [1], [2]. However, to support these new applications and to ensure the continued growth of the Social Web, we are first interested in understanding the quality of user-contributed comments and in mitigating the potentially negative impact of spam and low-quality comments on the sustainability of the Social Web.

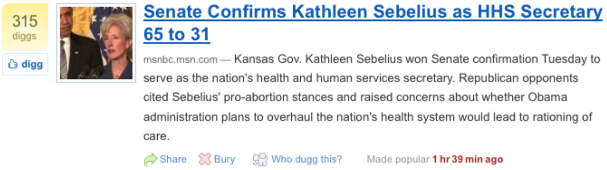
A number of recent studies have examined challenges to the quality of user-contributed content, including the quality of user-contributed tags [3], blog comments [4], user-contributed answers on Question-Answering forums [5], product reviews on Amazon [6], and so forth. In many cases, these quality assessments rely on experts external to the Social Web community (e.g., a panel of human experts declares that a blog comment is "spam" or "not-spam").

In this paper, we are interested in studying how a Social Web community itself perceives the quality of user-contributed comments within the community, so that the community is the final arbiter of quality. By studying how a community can self-regulate, we may gain insights into what a community values and how to sustain the positive growth of the community.

In particular, we propose to automatically rank the comments associated with a Social Web object (e.g., Web document, image, video) based on the expressed preferences of the community itself. By learning ranking functions for user-contributed comments, we may (i) automatically score new comments as they arise in the community; (ii) promote high-quality comments; (iii) filter out low-quality comments, so that user attention is not wasted; (iv) provide a sound basis for enhanced comment-based Social Web applications like summarization, content retrieval, visualization, and so on.

Learning to rank comments on the Social Web is challenging, however. In many cases, the comments are fairly short, lacking the structure and attributes necessary for traditional Web content and link-based quality metrics (e.g., PageRank). Additionally, the comment quality may vary from object to object and from community to community (e.g., NYTimes articles may attract insightful comments, whereas YouTube comments may attract more juvenile comments), and so the learning model should be flexible across these dimensions, so that comment quality is assessed relative to the object and its community. In addition, comments posted early may receive more social attention [7], and hence, it is important that when training a ranking model to balance the visibility of the comment with its intrinsic quality (i.e., breaking the feedback loop, so that early comments are not always preferred over later comments). Finally, it is important that a ranking model perform especially well on the top-k comments for small k, since users and applications are typically most interested in these high-quality comments.

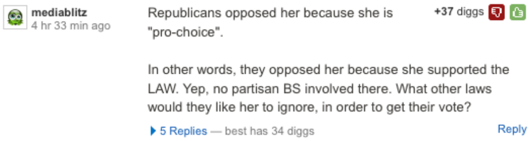
With these challenges in mind, we propose and evaluate a regression-based learning approach for automatically ranking comments based on the expressed preferences of the community itself. Concretely, we study the popular social



Senate Confirms Kathleen Sebelius as HHS Secretary 65 to 31

msnbc.msn.com — Kansas Gov. Kathleen Sebelius won Senate confirmation Tuesday to serve as the nation's health and human services secretary. Republican opponents cited Sebelius' pro-abortion stances and raised concerns about whether Obama administration plans to overhaul the nation's health system would lead to rationing of care.

Fig. 1. Example article with 315 "Diggs" (article community rating).



Republicans opposed her because she is "pro-choice".

In other words, they opposed her because she supported the LAW. Yep, no partisan BS involved there. What other laws would they like her to ignore, in order to get their vote?

Fig. 2. Example comment associated with the article in Figure 1. This comment has a comment community rating of +37.

news aggregator Digg and the socially-generated comments that Digg users can annotate news articles with. We study several factors impacting the community's preference for user-contributed comments, including the contributor's reputation and community activity level, as well as the complexity and richness of the comment. Through experiments, we find that the proposed approach results in significant improvement in ranking quality versus alternative approaches. Additionally, we study an extension to the model for balancing the visibility of a comment with its intrinsic quality.

II. Background

Commenting systems on the Social Web have been growing in popularity in the past few years, from blogs and social media sites like YouTube and Flickr to major news sites like NYTimes.com. Many of these commenting systems include a rating component, so that users can rate the comments submitted by other users.

In this paper, we study Digg, a popular social news aggregator. Digg users can submit stories to the community, rate stories that have been submitted by others (to "Digg" a story is to cast a positive vote for it), comment on stories, and engage in other typical Social Web activities (e.g., make friends with other users, track the stories that have been "Dugg" by others, and so on). With more than 27 millions visitors in the past year, Digg is one of the most successful social news aggregators, and an even more popular Web destination than NYTimes.com and CNN.com [according to statistics from compete.com].

Figure 1 illustrates an example submission to the Digg community, in this case a news article about Kathleen Sebelius. We can see that the story has received 315 Diggs (or 315 positive votes) by members of the Digg community, making this a quite popular story. We call this score the *article community rating*. Figure 2 illustrates a sample comment associated with this article. Digg users may rate each comment using a simple thumbs-up or thumbs-down rating system; in this case, the comment has an aggregate score of 37 (45 up-votes and 8 down-votes). We refer to the aggregate score assigned to a comment as the *comment community rating*. When considering all of the comments associated with a Social Web object, we can order the comments according to these community-based ratings; we refer to this as the *community preference* for these comments. Our goal is to develop automatic techniques for learning this *community preference* even in the absence of explicit community ratings.

III. Learning to Rank Comments

In this section, we present the formal model for ranking comments on the Social Web by community preference. We approach the problem of ranking comments as a regression problem.

Consider a set of k Social Web objects (e.g., Web documents, images, videos) $O = \{o_1, o_2, ..., o_k\}$. Each object o_i has a set of up to n comments associated with it $C_i = \{c_{i1}, c_{i2}, ..., c_{in}\}$. Each comment c_{ij} has a set of m features $F_{c_{ij}} = \{f_1, f_2, ..., f_m\}$. Each feature refers to some quality measure with respect to the comment. In the following section, we will explore a number of different possible feature choices.

We assume there exists some training data that has the form:

$$\{(F_{c_{1,1}}, r_{c_{1,1}})...(F_{c_{1,n}}, r_{c_{1,n}}), (F_{c_{2,1}}, r_{c_{2,1}})...(F_{c_{2,n}}, r_{c_{2,n}}), ..., \\ (F_{c_{k,1}}, r_{c_{k,1}})...(F_{c_{k,n}}, r_{c_{k,n}})\} \subset F \times \mathcal{R}$$

where the pair $(F_{c_{ij}}, r_{c_{ij}})$ corresponds to the feature set for comment c_{ij} and the comment community rating $r_{c_{ij}}$ for comment c_{ij}.

To tackle the community preference-based ranking problem, we can train a regression model over this training data. Concretely, we build the model through (i) a selection of features, as we will discuss in the following section; and (ii) the application of Support Vector Regression [8], a state-of-the-art regression model similar-in-spirit to the popular Support Vector Machine classifier that has proven successful across many domains, e.g., [9].

Support Vector Regression uses an ϵ-insensitive loss function that defines a tube with radius ϵ around the hypothetical regression function. If the data is placed within this tube, the loss function can be regarded as 0. By introducing the positive slack variables ξ_i and ξ_i^*, the SVR regression can be formulated as the constrained optimization problem:

$$Minimize \quad \frac{1}{2}w^T w + C\sum_{i=1}^{l} \xi_i + \xi_i^*$$

$$Subject\ to \begin{cases} r_i - w^T\phi(F_{c_{ij}}) - b \leqslant \epsilon + \xi_i \\ w^T\phi(F_{c_{ij}}) + b - y_i \leqslant \epsilon + \xi_i \\ \xi_i, \xi_i^* \geqslant 0, i = 1, ..., l, \epsilon > 0 \end{cases}$$

where $\phi(F_{c_{ij}})$ is the feature mapping for each comment in the high dimensional feature space, w and b are the slope and offset of the regression line, and $C > 0$, called the regularization parameter, is a positive constant. The positive slack variables ξ_i and ξ_i^* are to measure the deviation of training samples outside the tube ϵ zone. The constrained optimization problem given by the equation can be reformulated into a dual problem formalism by introducing Lagrange multipliers. Based on the Karush-Kuhn-Tucker conditions, the function is given by:

$$f(F_c) = \sum_{g=1}^{k*n} \sum_{h=1}^{k*n} (\alpha_g - \alpha_g^*) K(F_{c_g}, F_{c_h}) + b$$

where α_g, α_g^* are the Lagrange multipliers corresponding to the training data. Note that for those comments that serve as support vectors, the $\alpha_g > 0$ and $\alpha_g^* > 0$ whereas all the other comments must have $\alpha_g = 0$, $\alpha_g^* = 0$. $K(F_{c_g}, F_{c_h}) = \phi(F_{c_g})\phi(F_{c_h})$ denotes the kernel function, which satisfies the Mercers conditions. The kernel function we used in this work is the radial basis function: $exp(\gamma * |F_{c_g} - F_{c_h}|^2)$. In practice, we use a robust SVR implementation with default parameters available as part of the LIBSVM package [10].

In the testing phase we use this model to predict a rating for the unseen comments associated with an object $S = \{s_1, s_2, ..., s_n\}$ (e.g., $S = \{30, 100, 40\}$). Based on these ratings we can determine the relative rank order of the unseen comments: $R = \{r_1, r_2, ..., r_n\}$ (e.g., $R = \{3, 1, 2\}$). Note that our goal here is not to precisely estimate the actual comment community rating for a comment. Since comments may be continually rated, a predicted rating may quickly become stale. Instead, our goal is to predict the *relative order* of comments, so that even as new ratings are made on the comments, the model will be able to capture the relative quality.

IV. Comment Representation

Given the baseline ranking model, we now turn to the choice of features to represent the comments. The quality of a ranking model is strongly influenced by the quality of the features used to model the domain. In this case, we study several factors that we hypothesize may influence comment community ratings – the visibility of the comment, the influence and reputation of the user contributing the comment, and the content of the comment itself. Note that although the following discussion focuses on Digg for clarity, the proposed model is designed for use with any collection of Social Web comments.

A. Comment Visibility

The first factor we consider is comment visibility within the community. Intuitively, if more users in the community view a comment, it is more likely to attract a larger community rating. Conversely, a comment that is viewed by very few community members (say, for a comment related to an article that is of little interest to the community), will have less capacity to attract a large community rating.

We measure the visibility of a comment through two factors: (i) the *article community rating* of the article that the comment is attached to; and (ii) the *comment posting time*, since earlier comments may have the capacity to be viewed by more community members than later arriving comments.

Figure 3 shows the average article community rating versus the average comment rating (for the top-50 comments per article) across eight top-level Digg categories. The correlation coefficient is 0.93, validating the intuition that articles with high visibility (via many article Diggs) attract more votes for their comments.

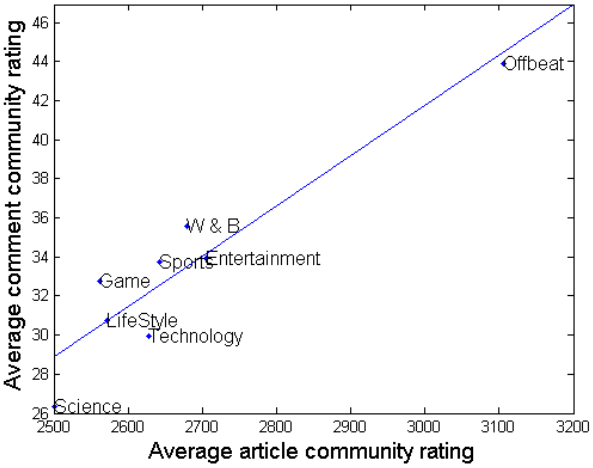


Fig. 3. Avg article rating vs. avg comment community rating (by category).

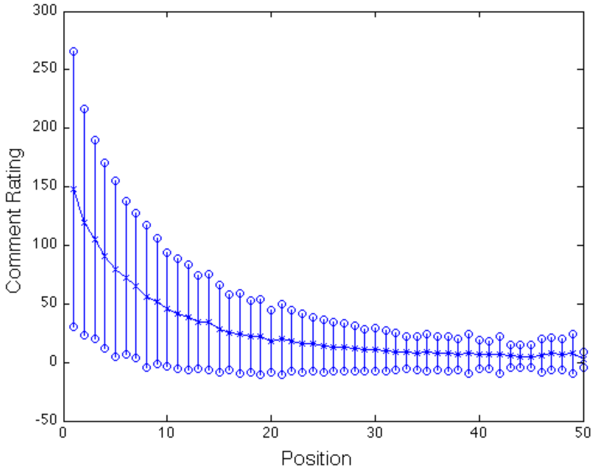


Fig. 4. Comment posting time (by position) versus comment community rating. We report the mean comment rating +/- one standard deviation.

Figure 4 shows that the mean score of comments that arrive earlier is greater than the mean score of comments arriving later, though with greater variability for early comments. In the figure, comments are arranged in order of their posting time (e.g, 1st, 2nd, ...). An early comment has greater visibility, and hence, greater capacity for a high community rating.

Recall that our overall goal is to automatically find the relative rankings of the comments associated with an article, even in cases when the community has not yet made its aggregate community preferences known. Hence, the first visibility feature (article community rating) will not necessarily be available for our prediction framework. As a result, we train the regression models with the article community rating feature to control for the article visibility bias across articles. For the testing phase we ignore the article community rating since it may not be known in practice and since all comments for an article would share the same feature value.

The second visibility feature – comment posting time – is known in the testing phase, and so we can use it as a prediction feature. Of course, it may be reasonable to try to control for comment posting time in much the same way we have controlled for the overall article visibility – so that potentially high-quality comments that happen to arrive late (and hence, may receive a low score due to low visibility within the community) are boosted to a higher position. Indeed, we study one possible "correction" factor in Section V-E.

B. User Reputation and Influence

We consider reputation and influence of the user contributing the comment. We want to know if a power user's comments will be more interesting and valuable to the Digg community. Here are some per-user features.

The first set of user-based features gives insight into each user's activity and interest level within the community:

- *Number of articles submitted*: This measures a user's activity in the community by the number of articles the user has submitted to the Digg community.
- *Community membership date*: This feature indicates how long the user has belonged to the community. For smoothing purposes, the account starting date (yyyymmdd) of each user is normalized into the range of 0 to 1, with higher values indicating newer members.
- *Category activity*: We calculate the percent of that user's article ratings to articles from each of the eight top-level Digg categories (e.g., Sports, Technology). For a comment from this user on a particular article, we take the user's category activity percentage for the article's category. The intuition is that for users who comment in an area of their expertise, their comments may have a higher likelihood of being appreciated by the community.

The second set of user-based features measures user popularity in the community:

- *Number of articles appearing on the Digg front page*: Digg uses a proprietary promotion algorithm to determine which stories submitted by its users reach the front page of Digg (and hence, reach the largest audience). A user who has had success submitting stories that reach the front page is an influential member.
- *Number of profile views*: How many times has the commenter's Digg profile been viewed?
- *Number of friends*: The number of friends of the commenter is recorded. Users with many friends may be more appreciated as commenters.

The final set of user-based features considers how well each user has participated in commenting in the past:

- *History of received comment ratings*: This feature measures the aggregate (sum) rating of a user's past comments. Does this user tend to make highly-rated comments? Or lowly-rated comments?
- *History of received comment replies*: This feature measures the number of replies that the commenter has received to past comments and can be viewed as a reflection of the interestingness of his comments.

C. Content-Based Features

The third factor we study are features related to the content of the comment itself. Since Digg and other Social Web sites attract comments from users with a wide-range of educational backgrounds, ages, and interests, the comments these users contribute may vary greatly in word choice, grammar, use of novel phrases, and so on. To capture the impact of these content-based attributes, we consider several semantic and statistical features of the comment text.

The first set of content-based features reflect some statistical properties of the text:

- *Comment length*: The first feature measures the number of words in the comment text. There may be a tradeoff between longer comments compared with the community's time and effort spent to appreciate the comment.
- *Comment complexity*: We measure the complexity of a comment by the entropy of the words in the comment. Equation 1 shows that for a comment c_j with λ number of words what is the entropy of c_j when each word has frequency p_i.

$$entropy(c_j) = \frac{1}{\lambda}\sum_{i=1}^{n} p_i[log_{10}(\lambda) - log_{10}(p_i)] \quad (1)$$

- *Number of upper case words:* This is a simple count of upper case words.
- *Comment informativeness*: Informativeness is meant to capture the uniqueness of the content in a comment relative to other comments attached to the same Social Web object. We measure the informativeness of comment c_j using a variation of the standard TFIDF approach from information retrieval, where we sum over the TFIDF values for all terms in a single comment:

$$inform(c_j) = \sum_{t_i \in c_j} \mathrm{tf_{i,j}} \times \mathrm{idf_i}$$

The tf component values terms that occur frequently within a comment: $tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}}$ where $n_{i,j}$ is the number of occurrences of the considered term in comment c_j, and the denominator is the sum of number of occurrences of all terms in comment c_j. The idf component values terms that occur infrequently across comments $idf_i = \log \frac{|C|}{|\{c:t_i \in c\}|+1}$ where $|C|$ is the number of comments and $|\{c : t_i \in c\}|$ is the number of comments in which t_i appears.
- *Category cohesion*: This feature measures the commenter's word choice with respect to the other comments within a particular category. The hypothesis is that each category has its own sub-community that uses particular jargon. Hence, comments that have high cohesion with the rest of the category are more likely to receive high

ratings. We measure category cohesion using the sum of the Mutual Information (MI) between all terms in the comment and the category (cat) of the article:

$$cohesion(c_j; cat) = \sum_{t \in c_j} MI(t, cat)$$

MI measures the amount of information each term t tells us about category cat: $MI(t, cat) = p^{'}(t|cat)p(cat)log(\frac{p(t|cat)}{p(t)})$. $p(t|cat)$ is the probability that term t appears in comments in cat. $p^{'}(t|cat)$ is a correction to $p(t|cat)$ that gives every term a non-zero probability of occurrence across all categories. Therefore we have $p^{'}(t|cat) = \alpha p(t|cat) + (1 - \alpha)p(t)$ as a smoothed probability that a comment contains term t given that it belongs to category cat. α is between 0 and 1. In practice we select a smoothing factor of $\alpha = 0.9$. $p(t)$ is the fraction of all comments containing t; and $p(cat)$ is the fraction of comments belonging to category cat. To prevent comments with more terms from receiving higher cohesion values, we also considered a version that divides cohesion by the number of terms in c_j. Experimentally, we find that this normalized version yields qualitatively similar results.

The next set of content-based features rely on NLP-style analysis of the comments:

- *Readability*: We measure the readability of a comment by its SMOG score [11], which estimates the years of education needed to understand a piece of writing.
- *Subjectivity vs. objectivity*: We measure the subjectivity/objectivity of each comment using the open source NLP tool LingPipe [12].
- *Verb+Noun count*: A simple count of verbs and nouns.

The last set of content-based features compare the comment text to the article the comment is attached to:

- *Comment-article overlap*: This feature measures the overlap between terms in the article abstract and the comment.
- *Comment-article polarity*: Finally, we measure if the polarity of each comment (positive or negative) matches the polarity of the article (using LingPipe [12]): 1 for agreement; 0 for disagreement.

V. Experiments

In this section, we evaluate the quality of the community-preference prediction model using the features described in the previous sections.

A. Data

For our dataset, we crawled the most-Dugg stories of the past 365 days in November 2008, resulting in a corpus of 9,000 Digg stories containing 247,004 comments submitted by 47,084 unique contributors. We focused our collection on these older pages since the commenting and rating activity has most likely stabilized for these stories, leading to a more reliable analysis of the comments.

B. Evaluation Method

Our evaluation is designed with three goals in mind. First, we aim to compare the learning-based ranking approach versus alternative approaches, to understand if the model does indeed capture salient features for predicting community preference. Second, we isolate the features used by the model to gain a better understanding of which comment features are good predictors of community preference. Finally, we explore an extension to the model for identifying and promoting high-quality comments that may have been overlooked.

As a baseline, we can measure the effectiveness of the learned model by comparing the predicted rank order of the comments to the ground truth rank order, as determined by the ground truth comment community ratings. Recall that it is important that the predicted comment rankings be of especially high-quality for the top-k comments for small k, since users and applications are typically most interested in these high-quality comments. Errors in ranking prediction at lower ranks are of less importance (e.g., swapping the 200th and the 201st comment). Hence, we evaluate the quality of the predictions using the well-known Normalized Discounted Cumulative Gain (NDCG) measure for evaluating the quality of top-k lists [13]. NDCG reflects this intuition by reducing the penalty of ranking errors logarithmically in proportion to the position of the comment. Formally, the discounted cumulative gain (DCG) is found for a top-k list as:

$$DCG_k = \sum_{i=1}^{k} \frac{fav_i}{\log_2(1+i)}$$

where fav_i is a favorability score for the comment at position i. We define the favorability score as its rank complement: $fav_i = N - Rank_i + 1$. For comparison across top-k lists for different articles, DCG is normalized by the *ideal* discounted cumulative gain at k. The ideal DCG ($iDCG_k$) is found by sorting the comments in order of their comment community rating and calculating DCG as above, resulting in $NDCG_k$:

$$NDCG_k = \frac{DCG_k}{iDCG_k}$$

NDCG ranges from 0 to 1, with higher-scores indicating greater agreement between the predicted rank order and the ideal rank order (based on the comment community ratings).

In all of the experiments reported here, we train and test the model using 10-fold cross validation and a 20-80 train-test split. After randomly sampling 24,000 comments from the dataset, the data is randomly split into 10 parts. We train the model over two of the parts (including the ground truth comment community rating) and then test the model over the remaining eight parts (for which the model has no access to the ground trust comment community rating). This procedure is repeated 10 times; the results are averaged over the 10-folds.

C. Experiment 1: Model Comparison

First, we compare the proposed model – denoted here as the Social Web Comment Prediction *SWCP* model – against two

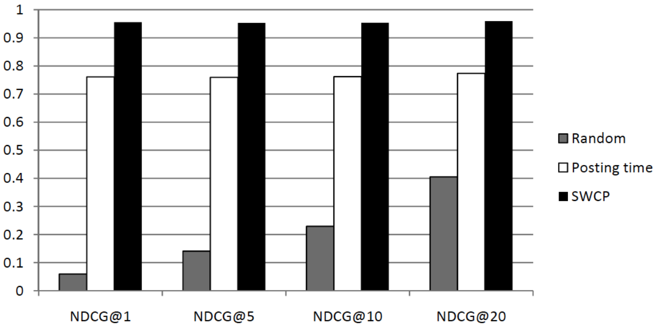


Fig. 5. Comparing the SWCP model versus alternatives.

alternatives: a random ranking model and a time-of-posting based ranking model. In the random ranking model, comment order is purely random. This simplistic model provides us with a baseline against which to compare the developed models. The second model is a time-of-posting ranking model. Recall that in Figure 4, we saw how comment posting time has a strong impact on its community rating, since earlier comments have greater visibility in the community. It might be reasonable to conjecture that posting time is all that matters. Concretely, this model assigns rank order to comments based solely on time-of-posting, i.e., comments arriving in the order $\{c_1, c_2, ..., c_n\}$ are ranked $\{1, 2, ..., n\}$.

Figure 5 shows the performance of the three models across four different NDCG k-values: NDCG@1, NDCG@5, NDCG@10, and NDCG@20. First note that both the comment-posting time model and the SWCP model outperform the random model for all NDCG metrics. Second, although the comment-posting time performs reasonably well, it alone is an insufficient determiner of comment community preference. We see that the inclusion of the user-based and comment-based features results in around a 25% improvement across all NDCG metrics. What is especially encouraging is that the model performs extremely well for the top-1 comment, meaning the model almost always correctly identifies the top-1 comment regardless of its posting time. The similarly good results for 5, 10, and 20 are also encouraging, validating the premise that comments, although a "messier" form of user-based annotation (compared to tags and ratings), do contain implicit quality signals that can be mined and used for automatic comment extraction by community preference. This has strong positive implications for the success of new comment-based applications (e.g., enhanced information organization, summarization, content retrieval, and visualization), as well as the continued success of the Social Web in the presence of growing spam and low-quality comments.

D. Experiment 2: Feature Study

Given the performance of the Social Web Comment Prediction model, what impact do the user-based and content-based features have on the prediction quality? Since evaluation of all possible feature combinations would be computationally expensive, we isolate the features in groups to better understand which features are good predictors.

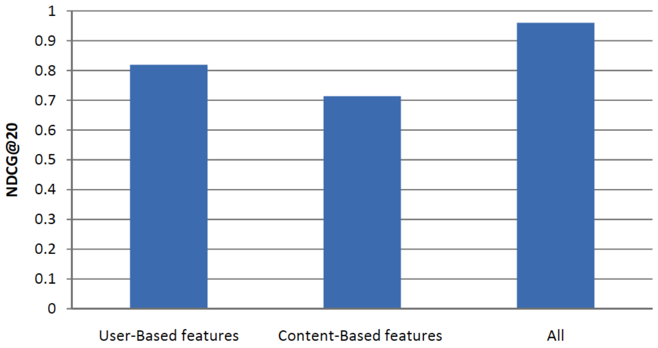


Fig. 6. Comparing feature sets.

First, we train two models – one using only user-based features (recall Section IV-B) and one using only content-based features (recall Section IV-C). Figure 6 shows the performance of the user-based model, the content-based model, and the full feature model for NDCG@20. We find qualitatively similar results for other values of NDCG@k (k=1, 5, 10). The user-based features alone do a better job than content-based feature alone, however, both approaches perform significantly less well than the full combination of features. We view the user-based features as a "prior" on the preference of the community for the user's comments. Only in combination with the actual comment text can we predict the community preference with good success. This negates the hypothesis that power users wield excessive control over comments (unlike the article promotion feature of Digg, which many presume is heavily influenced by power users).

To better understand the relative impact of particular user-based and content-based features, we next train and evaluate six models – one for each of the three user-based feature groups, and one for each of the three content-based feature groups. Table I reports the NDCG@k for k=1, 5, 10, and 20 for each of these six feature groupings.

TABLE I

Feature group	**NDCG@1**	**@5**	**@10**	**@20**
User activity and interest	0.61	0.62	0.65	0.70
User popularity	0.64	0.65	0.67	0.72
User comment history	0.66	0.69	0.71	0.73
Content statistics	0.62	0.65	0.67	0.71
Content NLP features	0.64	0.67	0.68	0.72
Comment-article	0.66	0.68	0.70	0.73

For the user-based feature, the user comment history feature group (recall that this includes the history of a user's previous comment ratings and the number of replies those comments have received) shows the strongest impact. This indicates that some users have a specialty for writing comments that are appreciated by the community; again, we can interpret this feature as a "prior" on a given comment's quality. Also note that content-based features are important; two of the top-three feature groups are content-based. We find it interesting that user activity and interest level – based on articles submitted, length of community membership, and category activity – is the single weakest performing feature group. Authoring comments that are perceived as high-quality by the community is largely independent of the user's activity level. Our hypothesis is that there are fundamentally different user types within a Social Web community: article submitters, article raters, commenters, etc. Exploring these different user types and their inter-relationship is an area deserving of further study.

In the final feature study, we explore the importance of content-based features for appropriately modeling the domain. We begin by assuming that our model has access to all user-based features. Could it be that comments are not really "messy"? And that by adding a single content-based feature we can equal the performance of the full feature model? Intuitively, this would mean that the comments contain some clear quality indicators once we factor in the "prior" for the user contributing the comment.

TABLE II

Feature group	**NDCG@1**	**@5**	**@10**	**@20**
All user-based features (A)	0.74	0.74	0.75	0.81
A + Text length	0.76	0.76	0.77	0.83
A + Upper case	0.74	0.74	0.75	0.81
A + Entropy	0.73	0.74	0.75	0.81
A + Informativeness	0.73	0.74	0.75	0.82
All features (user+content)	0.94	0.95	0.95	0.96

Table II reports the NDCG values for the baseline model considering only user-based features, plus four models that consider the baseline plus a single content-based feature only (text length, upper case, entropy, informativeness). In all, however, the content-based features are quite valuable. This indicates that comment content is complex, and that the community's preference for a comment is not driven by a simple feature. Instead, we see the need for full content analysis to capture this complexity.

E. Experiment 3: Rank Boosting

As we have seen in Figure 4, the comment posting time has a strong influence on the visibility of a comment and the resulting comment community rating. In this last experiment, we are interested in further exploring this phenomenon, as a first step toward breaking the rich-get-richer visibility cycle.

As an example, consider the four comments A, B, C, and D and their actual comment community ratings as illustrated in Figure 7. Applying a simple comment posting time ranking to these comments results in the rank order $\{A, B, C, D\}$. After applying the Social Web Comment Prediction model, we would ideally find the rank order $\{A, D, B, C\}$. This rank order is in strict order of the community ratings. Indeed, we have seen how the proposed model in this paper performs well on this problem.

It might be reasonable, however, to claim that comment D is the most preferred comment. Based on its late arrival time, but high community rating, we could assert that comment D has been most appreciated by the community relative to its smaller community visibility. This intuition motivates this last exploratory experiment.

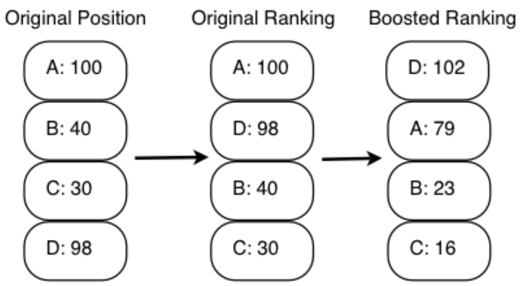


Fig. 7. Example illustrating the original time-of-posting position for each comment, the predicted ranking according to the SWCP model, and the boosted ranking using the positional boost modification.

Referring back to Figure 4, we propose to re-scale the comment community ratings for each training instance with respect to the average community rating for other comments posted in the same order position. In this way, we can evaluate a post arriving 4th (as in the example with comment D) against *all other comments* in our training data arriving 4th. The intuition is the further a comment's rating is from the average relative to other comments in the same position, then the more the comment's rating should be rewarded or punished.

Concretely, for a comment in the j'th position attached to a Social Web object i, we can define the *boosted* comment community rating $\hat{r}_{c_{ij}}$ with respect to all k comments at this same position as:

$$\hat{r}_{c_{ij}} = r_{c_{ij}} + r_{c_{ij}} \times \frac{r_{c_{ij}} - \bar{r}_{c_j}}{\sqrt{\frac{1}{k}\sum_{i=1}^{k}\left(r_{c_{ij}} - \bar{r}_{c_j}\right)^2}}$$

where $\bar{r}_{c_j}$ is the mean comment score at position j ($\bar{r}_{c_{ij}} = \frac{1}{k}\sum_{i=1}^{k} r_{c_{ij}}$) and the denominator is the variance of these comment scores. So a comment with a large rating in a position with a small average rating and small variance would be promoted to a new boosted rating.

Returning to our example, suppose the (average, variance) pairs of all comments at positions 1 to 4 are: (148, 235), (119, 193), (105, 169), and (91, 158). Applying the boosting formula results in the rank order $\{D, A, B, C\}$. Since comment D's original rating is much higher than the average rating for other comments at the same position, it is boosted from a score of 98 to 102. More importantly, comment A underperforms for its position and is penalized from 100 to 79.

In Figure 8 we compare this "boosted" version against the alternative random and time-of-comment ranking models. As in our original model, we see significant enhancement.

VI. Related Work

Our work in this paper is inspired by some previous studies of comments in message forums and newsgroups, including [14] and [15]. In particular, the Slashdot community – one of the acknowledged forebears of Digg and related social

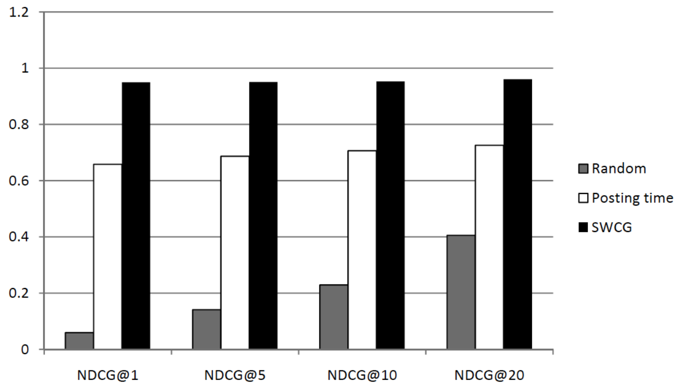


Fig. 8. Comparing the rank boosted SWCP model versus alternatives.

news aggregators – has attracted much attention. Several researchers have examined Slashdot's moderation policy for rating and filtering user-contributed comments, including [16] and [17]. Gomez et al. have studied the statistical properties of Slashdot discussion threads to identify degrees of controversial topics [18]. Other researchers have developed some machine learning approaches for semi-automating or fully automating the moderation of comments on Slashdot, including [19] and [20]. In a separate direction, Lerman has studied Digg and its article rating system in some detail, e.g., [21], [22], [23].

It is important to note that Digg and most new Social Web commenting systems differ from Slashdot in two ways. First, The eligible moderators of Slashdot are limited to a fraction of users. This is in direct opposition to the Social Web philosophy, in which all users are eligible to rate a comment. Second, Slashdot's comment rating policy restricts the ratings of a comment from -1 to 5, unlike Digg's comment rating system which is (potentially) unbounded, allowing for a wide variety of scores to be applied to comments. This purely democratic system could be potentially more problematic for sustaining the growth and quality of the community comment rating system, hence motivating the need for this work.

The Web community has recently examined "learning-to-rank" approaches for automatically ranking Web search results, e.g., [24], [25], among many others. In an earlier preliminary study, we considered a comment classification framework that assigned comments into four quality groups (e.g., "excellent", "fair") [27]. Inspired by [26], we find that a ranking-based approach is more intuitive and provides more accurate results than assigning class labels.

VII. Conclusions and Future Work

We have proposed and evaluated a regression-based learning model for automatically identifying comment quality within a Social Web community based on the community's preferences. We examined the impact of different comment features like visibility, user reputation of the comment's author, and the content of the comment itself to understand the influence of these features on the overall community's preference for comments. As part of our future work, we are interested to integrate these results as part of our broader research effort to build enhanced Social Web information management applications that leverage this social collective intelligence.

References

[1] M. Hu, A. Sun, and E.-P. Lim, "Comments-oriented document summarization: understanding documents with readers' feedback," in *SIGIR*. ACM, 2008.

[2] G. Mishne, "Using blog properties to improve retrieval," in *ICWSM*, 2007.

[3] S. Sen, M. F. Harper, A. Lapitz, and J. Riedl, "The quest for quality tags," in *GROUP '07: Proceedings of the 2007 international ACM conference on Supporting group work*. ACM, 2007.

[4] G. Mishne and D. Carmel, "Blocking blog spam with language model disagreement," in *Adversarial Information Retrieval on the Web*, 2005.

[5] E. Agichtein, C. Castillo, D. Donato, A. Gionis, and G. Mishne, "Finding high-quality content in social media," in *WSDM*. ACM, 2008.

[6] N. Jindal and B. Liu, "Opinion spam and analysis," in *WSDM*. ACM, 2008.

[7] G. Szabo and B. A. Huberman, "Predicting the popularity of online content," Nov 2008.

[8] H. Drucker, Chris, B. L. Kaufman, A. Smola, and V. Vapnik, "Support vector regression machines," in *Advances in Neural Information Processing Systems 9*, vol. 9, 1997.

[9] D. Sculley and G. M. Wachman, "Relaxed online svms for spam filtering," in *SIGIR*. ACM, 2007.

[10] C.-C. Chang and C.-J. Lin, *LIBSVM: a library for support vector machines*, 2001.

[11] G. H. McLaughlin, "Smog grading: A new readability formula," *Journal of reading*, 1969.

[12] B. Carpenter, "Phrasal queries with lingpipe and lucene," in *Proceedings of the 13th Meeting of the Text Retrieval Conference (TREC)*, 2004.

[13] B. Croft, D. Metzler, and T. Strohman, *Search Engines: Information Retrieval in Practice*, 1st ed. Addison Wesley, February 2009.

[14] D. Goldberg *et al.*, "Using collaborative filtering to weave an information tapestry," *Communications of the ACM*, vol. 35, no. 12, 1992.

[15] G. Mishne and N. Glance, "Leave a reply: An analysis of weblog comments," in *Workshop on the Weblogging ecosystem*, 2006.

[16] C. Lampe and P. Resnick, "Slash(dot) and burn: distributed moderation in a large online conversation space," in *Proceedings of the 2004 conference on Human factors in computing systems*. ACM Press, 2004.

[17] C. A. C. Lampe, E. Johnston, and P. Resnick, "Follow the reader: filtering comments on slashdot," in *CHI*. ACM, 2007.

[18] V. Gómez, A. Kaltenbrunner, and V. López, "Statistical analysis of the social network and discussion threads in slashdot," in *WWW*, 2008.

[19] A. Arnt and S. Zilberstein, "Learning to perform moderation in online forums," in *Web Intelligence*, 2003.

[20] A. Veloso, W. Meira, T. Macambira, D. Guedes, and H. Almeida, "Automatic moderation of comments in a large on-line journalistic environment," in *Proceedings of International Conference on Weblogs and Social Media*, 2007.

[21] K. Lerman, "Social networks and social information filtering on digg," in *Proceedings of International Conference on Weblogs and Social Media*, 2007.

[22] L. Krista, "Social information processing in news aggregation," *IEEE Internet Computing: special issue on Social Search*, vol. 11, no. 6, November 2007.

[23] K. Lerman and A. Galstyan, "Analysis of social voting patterns on digg," in *Proceedings of the ACM SIGCOMM workshop on Online Social Networks*, Jun 2008.

[24] X. Geng, T.-Y. Liu, T. Qin, and H. Li, "Feature selection for ranking," in *SIGIR*. ACM Press, 2007.

[25] Z. Zheng, K. Chen, G. Sun, and H. Zha, "A regression framework for learning ranking functions using relative relevance judgments," in *SIGIR*. ACM, 2007.

[26] W. W. Cohen, R. E. Schapire, and Y. Singer, "Learning to order things," *Journal of Artificial Intelligence Research*, vol. 10, 1998.

[27] E. Khabiri, C.-F. Hsu, and J. Caverlee, "Analyzing and predicting community preference of socially generated metadata: A case study on comments in the digg community," in *3rd Int'l AAAI Conference on Weblogs and Social Media ICWSM 09*, 2009.