A Random Walk on the Red Carpet: Rating Movies with User Reviews and PageRank

Derry Tanti Wijaya
National University of Singapore
School of Computing
Computing 1, Law Link Singapore 117590
+65 93839509
derrytan@comp.nus.edu.sg

Stéphane Bressan[1]
National University of Singapore
School of Computing
Computing 1, Law Link Singapore 117590
+65 6516 2727
steph@nus.edu.sg

ABSTRACT
Although PageRank has been designed to estimate the popularity of Web pages, it is a general algorithm that can be applied to the analysis of other graphs other than one of hypertext documents. In this paper, we explore its application to sentiment analysis and opinion mining: i.e. the ranking of items based on user textual reviews. We first propose various techniques using collocation and pivot words to extract a weighted graph of terms from user reviews and to account for positive and negative opinions. We refer to this graph as the sentiment graph. Using PageRank and a very small set of adjectives (such as 'good', 'excellent', etc.) we rank the different items. We illustrate and evaluate our approach using reviews of box office movies by users of a popular movie review site. The results show that our approach is very effective and that the ranking it computes is comparable to the ranking obtained from the box office figures. The results also show that our approach is able to compute context-dependent ratings.

Categories and Subject Descriptors
H.3.3 [**Information Search and Retrieval**]: Retrieval models, Selection process; I.2.7 [**Artificial Intelligence**]: Natural Language Processing – Text Analysis; H.2.8 [**Database Applications**]: Data mining

General Terms
Algorithms, Measurement, Performance, Experimentation.

Keywords: Opinion Mining, Ranking, PageRank.

1. INTRODUCTION
The success of Web 2.0 can be sized by the increasing popularity of forums and media in which users and organizations express and share views on anything and everything. Reviews can be found on individual blogs or specialized review sites such as cnet or tripadvisor®, for instance. Focused reviews are available on the World Wide Web for items as varied as consumer electronics hotels, public schools and election candidates. Name it; folks have reviewed it!

Unfortunately, the abundance of reviews also makes it challenging for users to compare and rank different items. Although quantitative ratings are often given with textual reviews, these ratings differ in scale, in notation, in criteria, etc. from website to website and hence are hard to aggregate. Some reviews are not even rated. In addition, quantitative ratings have been found to be insufficient in reflecting the opinions of the corresponding textual reviews [1, 2]. The ability to automatically rank different items based on their textual reviews will definitely be beneficial. This is the purpose of our paper.

In the field of information retrieval, given a graph where edges are weighted by the probability of traversing from one vertex to another, to rank the vertices we can simulate a random walk on the graph. As a walker proceeds in this random walk from vertex to vertex, he visits some vertices more often than the others. The vertices can therefore be ranked according to their scores: the probabilities that a walker will arrive at the vertices after the random walk.

To simulate such random walk and compute the scores of the vertices, we can represent the graph by its adjacency matrix and compute the fix point of the product of the matrix with itself [3] or approximate the computation of the fix point with PageRank [4] which introduces 'fatigue' to the random walker.

In this paper, we use the idea of random walk using PageRank to rank movies according to the opinions expressed in their textual reviews.

PageRank has been designed to rank Web pages. Unlike hypertext documents where edges are explicitly available (hyperlinks), there are no obvious edges that can be derived from movie reviews to build a graph where movies are vertices. Comparing movies based on general textual similarities of their reviews may not be entirely appropriate as movies tell different stories. The general textual similarities also do not reflect differences in the opinions expressed about the movie. We could then consider looking for sentences like "movie A is better than movie B", which make direct comparison between movies. Yet, such sentences are rarely found in individual reviews. Instead, in the reviews we commonly find sentences such as "I had a great time" and "the movie was horrible" which are expressing an opinion about the movie by means of adjectives.

CIKM'08, October 26–30, 2008, Napa Valley, California, USA.

[1] This work was partially funded by the National University of Singapore ARG project R-252-000-285-112, "Mind Your Language"

"Great" suggests a positive opinion. "Horrible" suggests a negative opinion. Whether an adjective expresses a positive or negative opinion is referred to as its semantic orientation. Other researchers have studied the semantic orientation of adjectives to infer opinion [5, 6, 7]. We also very commonly find sentences such as "this movie is good and funny" or "this movie is boring but has a good ending" in the reviews. The collocation of adjectives in such sentences forms, reinforces and amends the opinion expressed.

However, although some adjectives may have some positive or negative universal semantic orientation (e.g. "good", "excellent", "bad", "poor") other adjectives' orientation may not be known or depend on context [8, 9, 10, 11]. The design of effective context-dependent methods is generally considered a challenge in natural language processing [9]. We propose to use PageRank with a graph where vertices are adjectives and edges represent collocation (i.e. context-dependent clues on the semantic relationships between adjectives). Starting from a set of known adjectives (i.e. adjectives that have positive or negative universal semantic orientations), PageRank propagates their semantic orientations to other vertices whose orientations are not yet known and computes the semantic orientation scores. We can then rank movies by computing their individual scores from the semantic orientation scores of the adjectives in the movies' reviews. The higher the score of its adjectives, the more positive the opinions expressed about a movie: the higher the rank of the movie.

Furthermore, the semantic orientation of an adjective may depend on further facets of its context. For example, the adjective "funny" may have a positive semantic orientation when used in the review of a comedy movie: "the movie is so funny I had a good laugh", but may have a negative semantic orientation when used in the review of an action movie: "the villain looks a bit funny it was weird". We can therefore build the graph of adjectives for different context and granularity: we can build a single graph based on all reviews, we can build a graph by genre, or a graph by movie (we could also build graph by authors, by date or any other facets of context). In this paper we present the results for graph built from all reviews, graph built from reviews grouped by genre (e.g. comedy, action, horror etc.) and graph built from reviews grouped by movie.

In summary, we propose a practical context-dependent ranking procedure that can rank movies directly from their user reviews with no other resource required. The procedure is threefold. We first propose a simple yet effective technique for constructing a weighted graph of adjectives from the reviews. We use part-of-speech tagging, collocation and pivot words such as conjunctions (e.g. "and") and adverbs (e.g. "however") to create the graph. We refer to this graph as the sentiment graph. The graph is then used to compute semantic orientation scores of individual adjectives using PageRank. The scores of the individual adjectives from all the movie's reviews are combined to get the movie's score. The movies are ranked according to their scores.

We illustrate and evaluate our approach using reviews of recent box office movies by users of a popular movie review site. To measure the effectiveness of our ranking we use different metrics such as *average ranking error*, *percentage of overlap*, and *percentage of rank overlap*. We also look into the granularity of ranking and measure its effectiveness with regard to the *information loss* that a coarser ranking incurs. The results of this extensive performance evaluation demonstrate that the method we propose is very effective and can produce ranking comparable to the ranking induced from the box office figures and also show the limitations of user ratings. The results also confirm the context-dependence of the method.

Naturally, the approach can be straightforwardly applied to other items and reviews such as hotels, books and so on. Although we do not report these results here, we have conducted further experiments in other such domains that confirm the general effectiveness of our proposed approach.

Our contribution is fourfold. Firstly, from a ranking perspective, we contribute by making it possible to rank items using PageRank applied on a different graph other than the graph where the vertices are the items to rank. We use a related graph constructed from smaller components (adjectives) that express opinions about the items. This makes it possible to rank items based on opinions. Secondly, from the opinion mining perspective, we contribute by using PageRank algorithm to rank items context-dependently. Thirdly, we contribute by introducing *information loss* as a novel metric for measuring ranking. Lastly, we contribute a practical, effective and perhaps even predictive method for ranking items based on opinions expressed in their reviews.

The rest of the paper is organized as follows. In section 2, we survey related works. In section 3 we present our proposed method. In section 4 we present results of our experiments and we conclude in section 5.

2. RELATED WORK

2.1 Determining Semantic Orientation

Sentiment analysis and opinion mining are the generic natural language processing and text mining tasks involved in the processing of documents that express views and reviews in order to identify attitudes. One specific instance of opinion mining is the rating and ranking of items based on textual reviews. Its subtasks consist of determining and quantifying semantic orientation (positive or negative). The semantic orientation of an item, the feature of an item or the review for an item is usually aggregated from the semantic orientation of terms in the reviews: words, word senses or words of certain classes. The semantic orientation of terms is determined using starting set of terms whose semantic orientation is known and the terms' context in the review (collocation and pivot words such as conjunctions and adverbs, for instance), in some corpus (the Web), or in some known ontological resource like WordNet.

The authors of [12] propose to determine the semantic orientation of adjectives in texts. They use conjunctions (e.g. "and", "but") to derive the semantic orientation of adjectives. For instance "and" connects adjectives of the same orientation and "but" connects adjectives of different orientation. A clustering algorithm partitions adjectives into positive and negative clusters based on the conjunctions that links them.

The authors of [13] propose to determine the semantic orientation of word senses. They construct a lexicon called SentiWordNet where each word sense is associated with three scores, an objective score, a positive score and a negative score, to represent its semantic orientation. They use WordNet synsets and lexical relations together with a machine learning classifier to determine the scores. The same authors in [14] use PageRank on WordNet for the same task.

The author of [5] proposes to determine the semantic orientation of reviews by determining and aggregating the semantic orientation of

phrases in the reviews that contain adjectives and adverbs. He quantifies the semantic orientation of a phrase using its collocation with positive and negative adjectives and adverbs in Web documents as retrieved by search engines. The review is then classified as "recommended" if the average semantic orientation of its phrases is positive. It is classified as "not recommended" otherwise.

The authors of [8] propose to determine the semantic orientation of features of items (for instance: the battery life of a cell phone, the user friendliness of its menus, etc.). For this they determine and aggregate the semantic orientation of words in reviews. They leverage the observation that opinions with the same semantic orientation are commonly expressed in consecutive sentences, unless words such as "but", "however" articulate the successive sentences. If such words appear, the orientation is changed. The semantic orientation of a starting set of words is used to infer the orientations of other words. If the overall score of words expressed on a feature *f* in the sentence *s* is positive (resp. negative), then the semantic orientation of the opinion on *f* in the sentence *s* is positive (resp. negative).

The authors of [8] argue that semantic orientation should be context-dependent. It must capture usage in context. For instance the adjective "sharp" may be positive or negative depending on the item being reviewed.

Our approach quantifies the semantic orientation of opinions about the items in order to rank the items. We use collocation and pivot words such as conjunctions and adverbs to construct the sentiment graph. We use PageRank algorithm and a starting set of adjectives to quantify semantic orientations of adjectives in the graph. We aggregate these semantic orientation scores of adjectives to determine the final scores of the items. We rank the items according to their scores. Our approach is context-dependent.

2.2 Information Loss

In this paper, in order to measure the effectiveness of the proposed approach and its variants, we study different granularity of ranking.

Based on the quantitative scores of items obtained, we may either rank the items individually or we may group the items based on their scores: i.e. the highest *m* items, second highest *m* items, third highest *m* items, etc., and then rank the groups.

m can take the integer value between 1 to N_{item}, where N_{item} is the total number of items. $m = 1$ means we rank the items individually (finest granularity), $m = N_{item}$ means we group all items into one group and assign this group a rank (coarsest granularity).

We introduce coarser granularity ranking ($m > 1$) because we believe users may often be more interested in knowing which group of movies is good, which group of movies is medium, and which group of movies is bad instead of the individual ranking of each movie.

We use the metric introduced by [15] to measure the coarseness of grouping in terms of information loss. In this model, information loss for a given item is proportional to the size of the interval of scores of items in its group. Total information loss is the sum of information loss of all items in the data set. We oppose ranking effectiveness to the information loss that a coarser ranking incurs. As far as we know, this is a novel way of measuring ranking.

3. PROPOSED METHODS

The procedure we propose for ranking items based on the text of their reviews is threefold. We first construct a sentiment graph from the collocation of adjectives, taking into account pivot words such as conjunctions and adverbs. Then we compute the semantic orientation scores of individual adjectives using PageRank algorithm and a starting set of known adjectives. Finally we aggregate the semantic orientation scores of adjectives in all the reviews of an item to compute the item's semantic orientation score and ranking.

3.1 Sentiment Graph

The sentiment graph is constructed as follows. We define three variants of the method, depending on whether we construct a sentiment graph from reviews grouped by item, a sentiment graph from reviews grouped by genre, or a sentiment graph from all reviews. We refer to these variants as individual_ byGenre_, and all_ respectively.

The text of the reviews to be processed is first tagged using a part of speech tagger to identify adjectives. It is also segmented into sentences. We use Brill's part-of-speech tagger [16] and Ratnaparkhi's sentence splitter [17].

We then extract adjectives from the text of the reviews. The adjectives constitute the vertices of the graph. Here, we have assumed that the adjectives in the reviews are related to the movie. There may be other adjectives in the reviews that may not be related to the movie. For example the adjective "terrible" in the sentence "I watch this movie in a terrible cinema". However, we believe that such usage of adjectives (which is not related to the movie in review) is infrequent; therefore its effect can be minimized when we take a large number of reviews.

There exists an edge between two vertices if the corresponding adjectives occur in the same sentence (i.e. if they collocate). The weight of the edge is commensurate to the number of sentences in which the two adjectives collocate. Collocation between adjectives indicates either reinforcement or amendments of semantic orientations between the adjectives.

We obtain a graph $G_{pn}=<N, E_{pn}>$ with N its set of vertices and E_{pn} its set of weighted edges. The weight $W_{pn}(i, j)$ of the edge between the vertices i and j is the number of collocations. W_{pn} is a $|N| \times |N|$ matrix called the adjacency matrix.

For example, given the sentences "the camera is small but smart", "Although the camera is small, I think it is quite smart", "the camera is small but affordable", "I think it is good that the camera is affordable", and "it is small and has poor quality", G_{pn} is shown in figure 1 (the number in the square brackets indicate the weight of the edge).

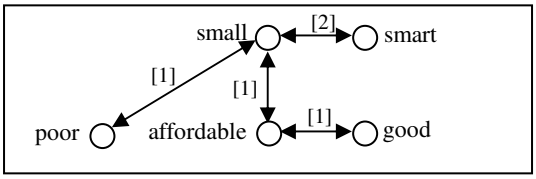

Figure 1. The graph G_{pn}

If two adjectives occur in a sentence where they are separated by words like "but", "although" or articulated in simple constructions such as "even if …, …" we refer to this situation as negative

collocation. Negative collocation between adjectives indicates amendments of semantic orientations between the adjectives.

We obtain a graph G_n=<N, E_n> with N its set of vertices and E_n its set of weighted edges. The weight $W_n(i, j)$ of the edge between the vertices i and j is the number of negative collocations. W_n is G_n's adjacency matrix.

For example, given the sentences "the camera is small but smart", "Although the camera is small, I think it is quite smart", "the camera is small but affordable", "I think it is good that the camera is affordable", and "it is small and has poor quality", G_n is shown in figure 2 (the number in the square brackets indicate the weight of the edge).

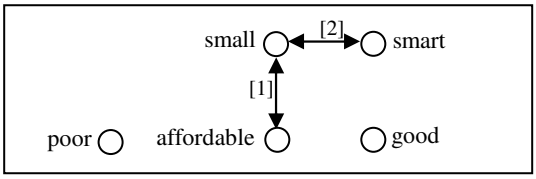

Figure 2. The graph G_n

If two adjectives are negatively collocated to the same adjective in different sentences, we treat them as being positively collocated. For example, if we have two sentences: "the camera is small but smart" and "the camera is small but affordable" the adjectives "smart" and "affordable" are considered positively collocated. We compute the co-citation matrix W_c of W_n [18]. Positive collocation between adjectives indicates reinforcement of semantic orientations between the adjectives.

The final sentiment graph G is a structure <N, E> with N its set of vertices and E its set of weighted edges (self-loops removed) where

$$W(i, j) = W_{pn}(i, j) - W_n(i, j) + W_c(i, j)$$

W is the adjacency matrix of our sentiment graph.

For example, given the sentences "the camera is small but smart", "Although the camera is small, I think it is quite smart", "the camera is small but affordable", "I think it is good that the camera is affordable", and "it is small and has poor quality", G is shown in figure 3 (the number in the square brackets indicate the weight of the edge).

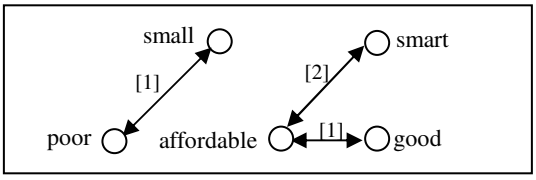

Figure 3. The graph G

3.2 PageRank

PageRank algorithm is applied to the sentiment graph obtained in 3.1 to compute the semantic orientation scores of adjectives.

We define two sets containing known adjectives with positive and negative semantic orientation respectively. We assign non-zero initial semantic orientation scores to these adjectives. These semantic orientation scores will be propagated to other adjectives during the course of PageRank application on the graph. In this manner, the semantic orientation scores of unknown adjectives can be computed.

The set *Good_Adjectives* is the set containing known adjectives with positive orientation. The set *Bad_Adjectives* is the set containing known adjectives with negative orientation.

The vertex in the graph is assigned a non-zero initial semantic orientation score if the corresponding adjective is in the set *Good_Adjectives* or *Bad_Adjectives*, and is assigned zero initial semantic orientation score otherwise: i.e. we construct a vector a_p^0 = <a_1… $a_{|N|}$> in which a_i is 1/|Good_Adjectives ∩ N| if the corresponding adjective is in *Good_Adjectives* and 0 otherwise, and we construct a vector a_n^0 = <a_1… $a_{|N|}$> in which a_i is 1/|Bad_Adjectives ∩ N| if the corresponding adjective is in *Bad_Adjectives* and 0 otherwise.

PageRank [4] computes the fix point or stable state of the product of an adjacency matrix with itself and a damping factor (the probability, at any step, that a walker will continue walking). This is similar to a random walk in the graph defined by the matrix, for a walker getting fatigued and switches to a random vertex according to the damping factor.

The input to PageRank algorithm is the adjacency matrix W normalized into W_{norm} where $W_{norm}(i, j) = W(i, j) / \Sigma_{\forall k \in N} W(k, j)$ and a vector a^0 (a_p^0 or a_n^0), which is the initial semantic orientation scores assigned to the vertices (adjectives).

In the formula below, α is the damping factor and e represents the probability that a random walker will choose a random vertex when it gets tired. As in [4] this probability is set to be equal for all the vertices. PageRank algorithm iteratively computes the semantic orientation scores of the vertices (adjectives), i.e. the vector a:

$$a^k = \alpha W_{norm} a^{k-1} + (1- \alpha) e$$

As in [4], we set α to be 0.85. e = <e_1…$e_{|N|}$> is constant across iterations. We set e_i =1/|N| for any i as in [4].

When we use a_p^0, we propagate the semantic orientation scores of known positive adjectives to other adjectives in the graph. Correspondingly, when we use a_n^0, we propagate the semantic orientation scores of known negative adjectives to other adjectives in the graph. Therefore, depending whether we use a_p^0 or a_n^0, we obtain methods that compute positive or negative semantic orientation scores of the adjectives in the graph, respectively.

We refer to these methods as _Positive and _Negative, respectively.

The vertices (adjectives) can also be ranked according to their semantic orientation scores to produce context-dependent ranking of adjectives.

The positive (resp. negative) score of each item is computed as the sum of positive (resp. negative) scores of adjectives from all its reviews. The sum considers duplicates, i.e. an adjective that appears twice in the reviews will contribute its score twice towards the total sum. In this paper we use sum to combine the scores of the adjectives to compute the score of the item. Other aggregate function is certainly possible and can be explored in the future work.

3.3 Proposed Methods

Depending on how we construct a sentiment graph, we define three variants: (1) individual_: we construct a sentiment graph from reviews grouped by individual item, (2) byGenre_: we construct a sentiment graph from reviews grouped by genre, and (3) all_: we construct a sentiment graph from all reviews.

Depending on our input to PageRank, we define two variants: (1) _Positive: we input the matrix W and the vector a_p^0 to PageRank to compute positive semantic orientation scores of adjectives, (2) _Negative: we input the matrix W and the vector a_n^0 to PageRank to compute negative semantic orientation scores of adjectives.

Using the computed positive and negative semantic orientation scores, we define another variant called _PositiveNegative which computes positive semantic orientation score minus negative semantic orientation score.

Therefore in total we propose 9 methods: (1) individualPositive, (2) individualNegative, (3) individualPositiveNegative, (4) byGenrePositive, (5) byGenreNegative, (6) byGenrePositiveNegative, (7) allPositive, (8) allNegative, and (9) allPositiveNegative.

4. EXPERIMENTS

In our experiment, we define the set *Good_Adjectives* to contain 19 adjectives: "good" and its synonyms: "excellent", "brilliant", "well", "better", "best", "worthy", "worth", "nice", "great", "perfect", "positive", and negative antonyms: "not bad", "not horrible", "not terrible", "not awful", "not worse", "not worst", "not negative".

We define the set *Bad_Adjectives* to be the exact mirror image of *Good_Adjectives* (i.e. it contains 19 adjectives: "not good", "not excellent", "not brilliant", etc.)

We illustrate and evaluate our approach using reviews of box office movies written by users of a popular movie review site. We pick 50 movies randomly from box office list of November 2007 to February 2008. For each movie, we download all its users' reviews. For each movie we note its box office figure, its overall quantitative user rating, and its genre. The movies are of genre action, animation, children, comedy, drama, foreign film, horror, musical, romance, science fiction, chick flick, crime, political, or psycho.

In our experimental data, we have quantitative user rating for each user and each movie (on a scale of 1 to 10 stars). These ratings are averaged to provide an overall user rating for the movie. However, there is evidence [1, 2] that such rating for measuring reviews is not reliable. The unreliability of user quantitative rating is attributed to its inconsistency [1] and its too coarse granularity [2]. Similar qualitative textual reviews can yield very different quantitative ratings from users. In the most extreme case, the users do not understand the rating system and give a 1 instead of a 10. Choosing a number between 1 and 10 to quantify one's opinion is subjective and difficult [1]. The coarse granularity of the rating scale (1 to 10 stars) for measuring reviews has the underlying assumption that the opinion in textual review is perfectly classified (summarized) into the 10 classes of the star rating [2]. Yet the findings in [2] clearly indicate that the actual text contains significantly more information than the ratings. The loss of information due to the mapping from textual reviews to the coarser star rating is irretrievable. Inconsistency and coarse granularity are the paradox of user rating because to reduce one will mean to increase the other. For example, although it is easier to be consistent when choosing between "good" or "bad instead of choosing a number from 1 to 10; the 2 classes of rating (coarser granularity) loses more information than the 10 classes of rating (finer granularity). We further investigate the effectiveness of user ratings in our experiments.

In our experimental data, we have also objective figures that represent the opinions of the general audience: i.e. the box office figures. The box office figure is the gross income of the movie, which is the number of tickets sold (indicates the audience's decision to watch the movie) times the price of the ticket (indicates the audience's willingness to pay).

If we assume that reviewers are representative of the general audience and that reviews are representative of the general audience's opinions, then the box office figures should be a suitable source of reference ranking to measure performance in our experiments.

We recognize that box office figures may not be the only robust and objective source of reference ranking; it is however an important and valuable one, especially from the marketing point of view.

We construct the sentiment graph and run PageRank on the graph. PageRank computes semantic orientation scores of each vertex (adjective) in the graph. We sum the semantic orientation scores of adjectives in the reviews of an item to determine the semantic orientation score of the item. We rank the items according to their scores.

4.1 Measuring Ranking Performance

In this paper we present three *metrics* for *measuring* ranking performance.

The first metric is *Percentage of Overlap* [19] which is the size of the overlap between two top-*k* lists: i.e. how many movies in the top-*k* list of box office ranking are in the top-*k* list of our ranking. We normalize this measure by dividing it with *k* to get the *Percentage of Overlap*. The bigger the overlap, the better is our ranking in matching the box office ranking. *k* can take a value between 1 to N_{item}, where N_{item} is the total number of movies.

The second metric is *Average Rank Error*. For each movie, we compute the difference between the rank we produce for the movie and the movie's box office rank. *Average Rank Error* is the average of these rank differences. The smaller the average, the better is our ranking in matching the box office ranking.

The third metric is *Percentage of Rank Overlap* which is the percentage of movies out of the total number of movies that have the *same* numerical rank in our ranking as in the box office ranking. This is a stricter measure than the *Percentage of Overlap* metric [19] which does not care about the actual numerical ranks. The bigger the rank overlap, the better is our ranking in matching the box office ranking.

4.2 Evaluating Ranking Performance

In this paper we present two *methods* for *evaluating* ranking performance.

We can evaluate the ranking of the entire data or we can evaluate the ranking of just the subset (top-*k*) of the data. We call this

method of evaluating ranking *Top-k*, for k = 1 (we evaluate the ranking of the top 1 movie), k = 2 (we evaluate the ranking of the top 2 movies), to k = 50 (we evaluate the ranking of the entire data).

We can evaluate the ranking of individual movies or we can evaluate the ranking of the groups of movies. We call this method of evaluating ranking *Granularity-g*, for g = 1 (we evaluate the ranking of individual movies), g = 2 (we group movies by 2 based on their scores (i.e. highest 2 movies, second highest 2 movies, third highest 2 movies, etc.), assign each group a rank, and evaluate the ranking of the groups), to g = 50 (we group all movies in one group, assign this group a rank, and evaluate this coarsest ranking). We measure ranking effectiveness with regard to the information loss that a coarser ranking incurs.

A combination of the first and second method is possible. For example, we can consider grouping the movies then evaluating top-*k* lists of each group or we can consider grouping the movies then evaluating the ranking of the top-*k* groups only. However we do not explore it in this paper due to space consideration.

4.3 Experimental Results

We present results of evaluating our ranking against the box office ranking. We use *Top-k* and *Granularity-g* method for evaluating performance. For each of the evaluation, we present metrics for measuring ranking performance. We also present interesting result for the ranking of adjectives of each genre.

4.3.1 Top-k

We compare the top-*k* list of our ranking with the top-*k* list of the box office ranking, for *k* = 1 to *k* = 50.

For each *k*, we present percentage of overlap and average rank error.

Percentage of overlap is measured as the size of overlap (the number of movies in the top-*k* list of our ranking which are in the top-*k* list of the box office ranking), divided by *k*.

Average rank error is measured as the average of rank differences between the ranks of movies in the top-*k* list of the box office ranking and their ranks in our ranking.

In figure 4, we present the percentage of overlap between the top-*k* list of our ranking and the top-*k* list of the box office ranking.

We see that all our methods (except individualNegative, byGenreNegative, and allNegative) perform better (higher percentage of overlap with box office ranking) than the ranking from user ratings. individualPositive and byGenrePositive perform the best, achieving more than 70% of overlap with box office ranking for almost all *k*.

In figure 5 we compare the average rank error between the ranks of movies in the top-*k* of box office ranking and their ranks in our ranking.

From figure 5 we can see that all our methods (except individualNegative, byGenreNegative, and allNegative) perform better (lower average rank error) than the ranking from user ratings. individualPositive and byGenrePositive perform the best.

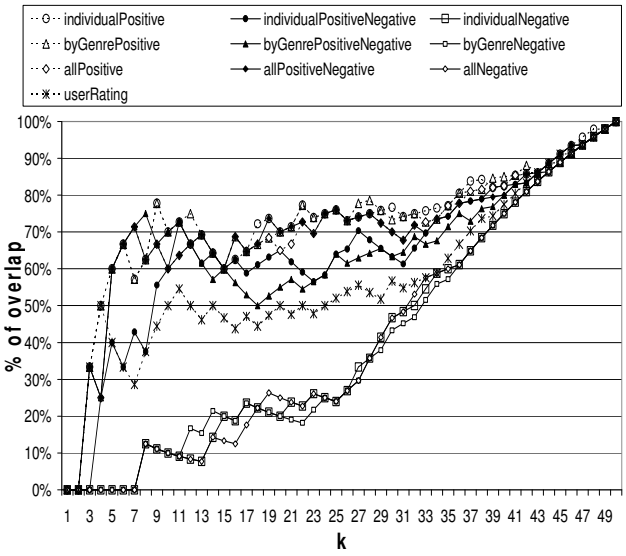

Figure 4. Percentage of Overlap in top-*k* Movies

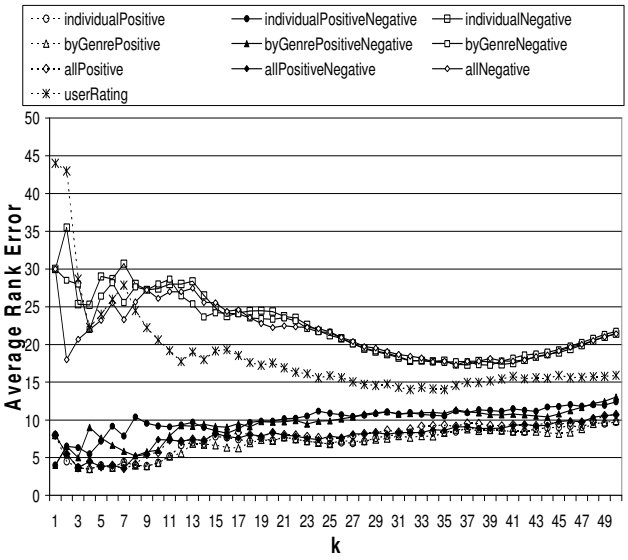

Figure 5. Average Rank Error in top-*k* Movies

From these results, we observe that methods which use negative semantic orientation scores are not as effective as those with just the positive semantic orientation scores. This highlights a question on how best to perceive and use the negative semantic orientation scores: i.e. non-negative orientation may not always mean positive orientation, and positive and negative semantic orientation scores may not always combine in a linear fashion.

From these results, we also observe that methods which use reviews grouped by genre (byGenre_) perform better than the methods which combine all the reviews (all_). This maybe because, when we combine all reviews, we lose information on the genre context of the adjectives. Such context information maybe important in determining the semantic orientations of adjectives which depend on genre: e.g. funny – which maybe

positive in comedy genre but negative elsewhere, scary – which maybe positive in horror genre but negative elsewhere, etc.

From these results, we also observe that all our methods (except individualNegative, byGenreNegative, and allNegative) perform better than user ratings in inferring the box office ranking, which we believe to be an objective measure for ranking opinions about the movies. Our results confirm similar observations in [1, 2].

4.3.2 Granularity-g

We group movies at different granularity g, for g = 1 (one movie in a group) to g = 50 (all movies in one group). After grouping, movies in the same group are assigned the same rank. Hence, different grouping results in different ranking.

For each g, we present percentage of rank overlap and average rank error with the information loss incurred from coarser ranking.

Percentage of rank overlap is measured as the percentage of movies out of all movies in our dataset which has the same numerical rank in our ranking as in the box office ranking.

Average rank error is measured as the average of rank differences between the rank we produce for each movie and the movie's box office rank.

Information loss is measured as the sum of information loss of each movie. The information loss of each movie is the range of box office figures in its group divided by the maximum range of box office figures of all movies in our dataset.

In figure 6 we present the percentage of rank overlap vs. information loss at different granularity g when we compare our ranking to box office ranking.

In figure 6 we can see that all our methods (except individualNegative, byGenreNegative, and allNegative) perform better (higher percentage of rank overlap for the same granularity g) than the ranking from user ratings. individualPositive and byGenrePositive perform the best.

In figure 7 we present the average rank error vs. information loss at different granularity g when we compare our ranking with the box office ranking.

When g = 1 (each movie is a group of its own), the information loss is zero and the average rank error is maximum. As we group movies (g > 1), the average rank error decreases more rapidly than the increase in information loss. This shows that grouping movies can improve the ranking result greatly without incurring too much information loss.

When we zoom in on figure 7, we see that all our methods (except individualNegative, byGenreNegative, and allNegative) perform better (lower average rank error for the same granularity g) than the ranking from user ratings. individualPositive and byGenrePositive perform the best.

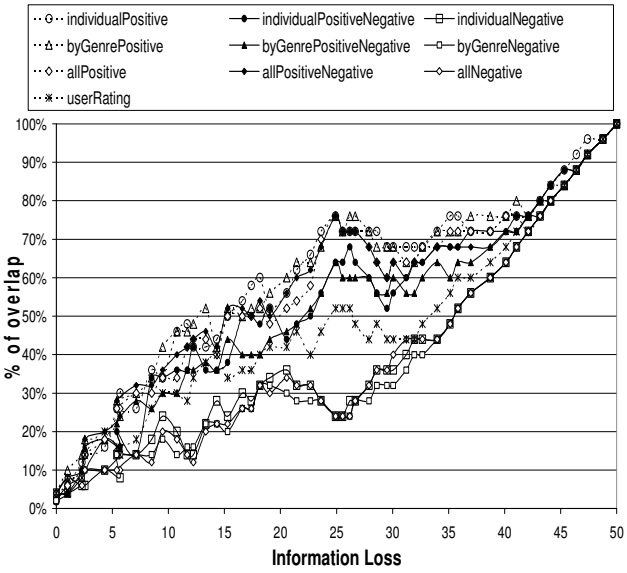

Figure 6. Percentage of Rank Overlap vs. Information Loss at Different Grouping Granularity *g*

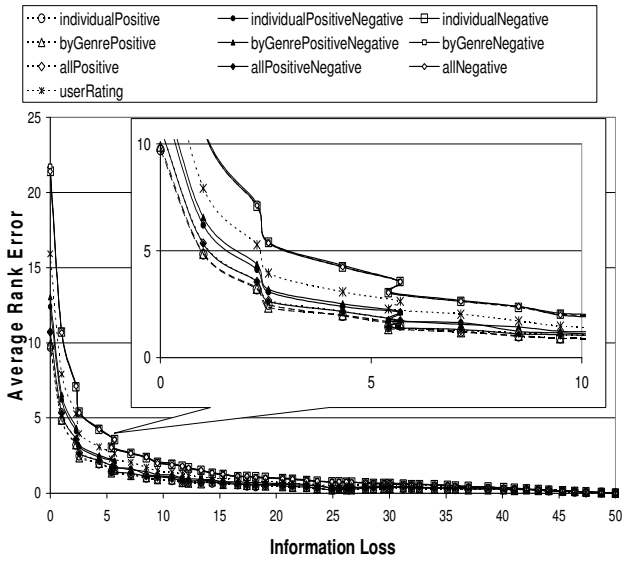

Figure 7. Average Rank Error vs. Information Loss at Different Grouping Granularity *g*

4.3.3 Sensitivity to Starting Adjectives

Here we present the results of evaluating the sensitivity of our method to the starting adjectives we define in *Good_Adjectives* and *Bad_Adjectives* sets. We present results of ranking by one of our best performing method (individualPositive) as compared to the box office ranking.

In figure 8, we present the results of running individualPositive method with different *number* of starting adjectives extracted from our *Good_Adjectives* set.

From figure 8, we can see that different number of starting adjectives do not vary the ranking results much in terms of their

percentage of overlap with the box office ranking. Even when we only use 1 starting adjective, our method still works in producing high quality ranking. From k = 6, the percentage of overlap with the box office ranking is always higher than 60% for various number of starting adjectives.

In figure 9, we present the results of running individualPositive method with different *subsets* of starting adjectives extracted from our *Good_Adjectives* set.

From figure 9, we can see that different subset of starting adjectives do not vary the ranking results much in terms of their percentage of overlap with the box office ranking.

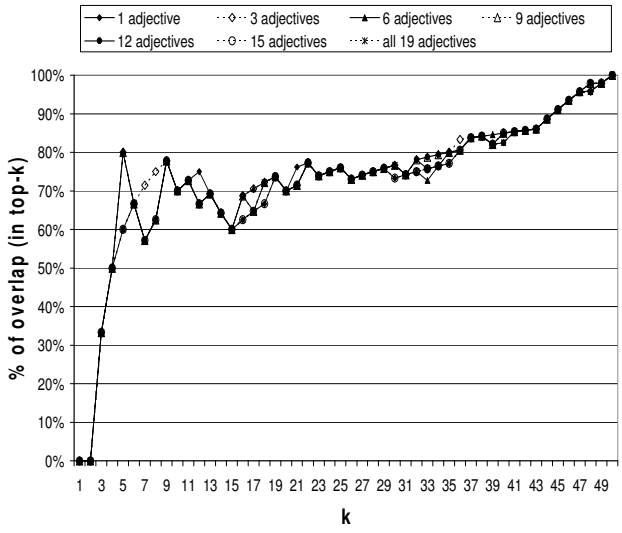

Figure 8. Percentage of Overlap in top-*k* Movies (individualPositive method)

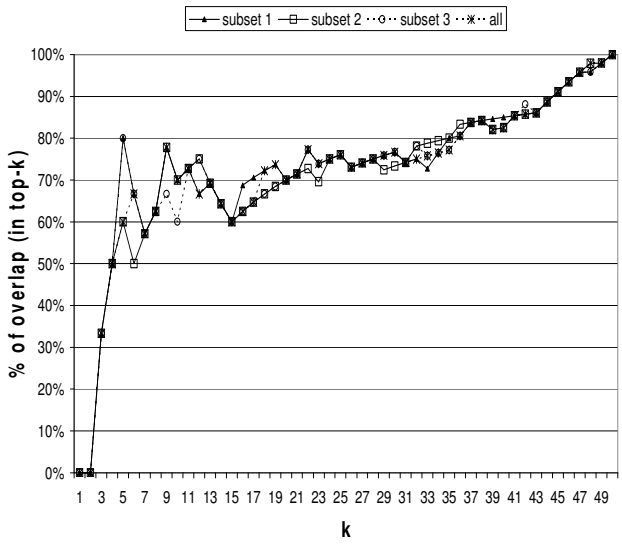

Figure 9. Percentage of Overlap in top-*k* Movies (individualPositive method)

These results show that our method is not overly sensitive to the number or the choice of starting adjectives. The sentiment graph itself may have contained enough information on the semantic orientations of its vertices (adjectives).

4.3.4 Ranking of Adjectives

The interesting thing about our proposed methods is that, not only they can produce the ranking of the movies; they can also produce the context-dependent ranking of the adjectives in the reviews.

Here we present some interesting results from the ranking of the adjectives when we conduct our method: byGenrePositive using only one positive adjective: "good" in our starting set.

Using only the adjective "good" as its starting adjective, our method is able to find that the adjective "great" has also a universal positive semantic orientation in all the genres: i.e. "great" is ranked in the top 1% of adjectives with highest positive orientations, in all the genres.

Among other interesting top 1% adjectives with highest positive orientations are: "funny" (in comedy, chick flick, animation and children genres), "stupid" (in the comedy genre), "animated" (in animation and children genres), "musical" (in the musical genre), "political" and "flawed" (in the political genre), "original" (in the psycho genre), "enchanted" and "fairy" (in the children genre), "real" (in the drama genre), "young" and "British" (in the romantic genre). Some of these adjectives have either ambiguous orientations or orientations that are genre-specific.

For example, the adjectives "flawed" and "stupid" have ambiguous semantic orientations: i.e. they can have positive or negative semantic orientations depending on how they are used in the sentence. Our method is able to identify that these adjectives have positive orientations in the political and comedy genre, respectively. Further investigation to the actual reviews reveals interesting usage of the adjective "flawed" in the political genre: "… a rather affectionate look at a flawed man who felt compelled to right what was wrong", "Wilson Hanks, a flawed and fun loving Congressman from the piney woods of East Texas…", and the interesting usage of the adjective "stupid" in the comedy genre: "I like a stupid movie where I do not have to think in and just sit back", which suggest the positive orientations of the adjective "flawed" and "stupid" in the sentences.

Further, "political", "musical", and "animated" are adjectives whose usage and orientations maybe specific only to the political, musical and animation genre respectively. Our method is able to identify that these adjectives have indeed positive orientations in their respective genres even when these adjectives are not in our starting set of positive adjectives.

Lastly, another interesting issue to explore is whether or not the adjectives actually reflect the audience demands for what will be considered good movies for a particular genre. For example, among the top 1% of adjectives (with highest positive semantic orientations) in the romantic genre is the adjective "British". Indeed, British romantic movie has done continuously well in topping the box office list with movies such as "Bridget Jones' Diary" and "Four Weddings and a Funeral". Another example is the adjective "animated" that is ranked among the top 1% of adjectives with highest positive semantic orientations in the children genre. Indeed, animated movies in children genre have done very well in the past with movies such as "Shrek", "Finding Nemo", and "Toys Story". We are interested in exploring this issue further in our future work.

5. CONCLUSIONS

Our contributions include a novel and practical procedure, a comprehensive performance analysis and a methodology for performance evaluation that includes metrics novel in this application domain.

We propose a novel and practical context-dependent ranking procedure that can rank items directly from the text of their reviews. The method uses simple contextual relationships such as collocation, negative collocation and coordination by pivot words such as conjunctions and adverbs to construct a sentiment graph. From a small starting set of adjectives whose orientation is universally known, the orientation of other adjectives in the reviews and the items reviewed can be computed and ranked using the PageRank algorithm. The method has several variants whether it uses positive or negative starting adjectives and whether the sentiment graph is constructed for individual items, by genre, or globally. From the ranking perspective, we have contributed by making it possible to rank items using PageRank applied to a different graph other than the graph where the vertices are the items to rank. We use instead a related graph constructed from the smaller components (terms, adjectives) that express opinions about the items. From the opinion mining perspective, we contribute by using PageRank algorithm to design context-dependent ranking of items based on opinions in their reviews.

We instantiate and evaluate our procedure and its variants using reviews of box office movies. While the design of effective context dependent methods is generally considered a hard topic in natural language processing [9], we show that our method is very effective and produces a ranking comparable to the one of the box office. Our best performing method uses positive starting adjectives and a sentiment graph constructed for individual items (individualPositive). This interestingly happens to be the simplest method. We also show that our method is not overly sensitive to the number or choice of starting adjectives. We compare our method and the box office ranking with the ranking induced from user ratings and show, if it was necessary, the limitation of the latter. This highlights the practical application of our proposal.

Our performance evaluation examines the usage of several metrics for measuring ranking used by different authors. We show how the notion of information loss can be used in evaluating coarseness when measuring ranking effectiveness at different granularities. This is a novel metric in evaluating ranking.

From the experiments, we find that ranking based on negative semantic orientation scores does not give as good a performance as ranking based on positive semantic orientation scores. This highlights the question on how best we should perceive the negative orientation scores. We find that positive and negative orientation may not combine well in a linear fashion and that non-negative orientation may not always mean positive orientation. In future, more combination of positive and negative orientation can be explored.

Lastly, the practical thing about our methods is its applicability to many more domains, to rank items based on their reviews. With such automated ranking of items based on reviews, companies could track public response to their products or services, politicians could track public opinions, and so on. In future, we will explore the application of our methods to many more domains.

6. REFERENCES

[1] Dave, K., Lawrence, S., and Pennock, D.M. 2003. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. WWW 2003.

[2] Ghose, A., Ipeirotis, P. G., and Sundararajan, A. 2007. Opinion mining using econometrics: A case study on reputation systems. In Proceedings of the 44th Annual Meeting of the ACL.

[3] Gondran, M. and Minoux, M. 1984. Graphs and Algorithms. John Wiley and Sons.

[4] Brin, S. and Page, L. 1998. The anatomy of a large-scale hypertextual Web search engine. Computer Networks and ISDN Systems, 30(1-7):107–117.

[5] Turney, P.D. 2002. Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. Proceedings of the 40th ACL.

[6] Hu, M. and Liu, B. 2004. Mining Opinion Features in Customer Reviews. AAAI-2004.

[7] Whitelaw, C., Garg, N., and Argamon, S. 2005. Using appraisal taxonomies for sentiment analysis. In Proc. Second Midwest Computational Linguistic Colloquium (MCLC).

[8] Ding, X. and Liu, B. 2007. The Utility of Linguistic Rules in Opinion Mining. SIGIR.

[9] Liu, B., Hu, M., and Cheng, J. 2005. Opinion Observer: Analyzing and Comparing Opinions on the Web. Proceedings of the 14th international WWW conference (Chiba, Japan, May 10-14, 2005).

[10] Wiebe, J., Wilson, T., and Bell, M. 2001. Identifying collocations for recognizing opinions. In Proceedings of ACL/EACL 2001 Workshop on Collocation.

[11] Wilson, T., Wiebe, J., and Hoffmann, P. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In HLT/EMNLP 2005.

[12] Hatzivassiloglou, V. and McKeown, K.R. 1997. Predicting the semantic orientation of adjectives. Proceeding of the 35th Annual Meeting of the ACL and the 8th Conference of the European Chapter of the ACL.

[13] Esuli, A. and Sebastiani, F. 2006. SentiWordNet: A Publicly Available Lexical Resource for Opinion Mining. In Proceedings of LREC-06, 5th Conference on Language Resources and Evaluation.

[14] Esuli, A. and Sebastiani, F. 2007. PageRanking WordNet synsets: An application to opinion mining. In Proceedings of the 45th Annual Meeting of the ACL (Prague, CZ).

[15] Byun, J., Kamra, A., Bertino, E., and Li, N. 2007. Efficient k-anonymity Using Clustering Techniques. Proceedings of International Conference on Database Systems for Advanced Applications (DASFAA).

[16] Brill, E. 1995. Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part of Speech Tagging. Computational Linguistics.

[17] Reynar, J. and Ratnaparkhi, A. 1997. A Maximum Entropy Approach to Identifying Sentence Boundaries. ANLP 1997: 16-19.

[18] Zhou, X. and Pu, P. 2002. Visual and Multimedia Information Management. Springer.

[19] Bar-Ilan, J., Mat-Hassan, M., and Levene, M. 2006. Methods for Comparing Rankings of Search Engine Results. Computer Networks 50 (1448-1463).