Ranking Online Reviews Based on Their Helpfulness: An Unsupervised Approach

Alimuddin Melleng **Anna-Jurek Loughrey** **Deepak P**
Queen's University Belfast, UK
alimuddinmllg@gmail.com, a.jurek@qub.ac.uk, deepaksp@acm.org

Abstract

Online reviews are an essential aspect of online shopping for both customers and retailers. However, many reviews found on the Internet lack in quality, informativeness or helpfulness. In many cases, they lead the customers towards positive or negative opinions without providing any concrete details (e.g., *very poor product, I would not recommend it*). In this work, we propose a novel unsupervised method for quantifying helpfulness leveraging the availability of a corpus of reviews. In particular, our method exploits three characteristics of the reviews, viz., relevance, emotional intensity and specificity, towards quantifying helpfulness. We perform three rankings (one for each feature above), which are then combined to obtain a final helpfulness ranking. For the purpose of empirically evaluating our method, we use review of four product categories from Amazon review[1]. The experimental evaluation demonstrates the effectiveness of our method in comparison to a recent and state-of-the-art baseline.

1 Introduction

Reviews are an essential aspect of information that allows users to obtain insight into a product of interest before purchasing. Typically, users write their reviews in order to express their satisfaction or dissatisfaction about purchased items or services. Products and sellers with more positive reviews tend to gain more new customers than products or sellers without reviews or with many negative reviews. This is because customers feel more confident buying products that have been recommended by other buyers. Popular products could have hundreds or thousands of reviews, which makes it impossible for the customers to read all of them. Moreover, it is not easy for a user to prioritize reading the most informative reviews since there are often no such ranking options. Some websites rank reviews based on the posting date or rating star, for example, Trustpilot.com and Reviews.io. Amazon uses a crowdsourcing mechanism, a voting system, to gather feedback on review helpfulness, and then rank them based on the overall votes they received (Amazon.com). A user can vote for a review as being helpful or unhelpful. Amazon was estimated to receive a revenue of about $2.7 billion by providing simple question "was this review helpful to you?" (Spool, 2009).

Although such a voting system is helpful for customers, it has several limitations due to the inherent character of the voting process. There are number of reasons: 1) not all reviews get the helpfulness vote; 2) the helpfulness voting does not work for cold star review (i.e., a new user or a new review will have much less votes) (Singh et al., 2017); 3) reviews receiving helpfulness votes would tend to gather more vote due to the snowball effect (e.g., phenomena such as social proof (Cialdini, 1987)).

In this work we hypothesise that helpfulness of a review should be assessed based on three characteristics, namely relevance (whether the review discusses the key features relevant to a specific product), emotional intensity (level of emotions expressed within a review) and specificity (level of details discussed in a review). We motivate the importance of each of those features later in the paper. We then propose an unsupervised helpfulness ranking method that does not depend on the helpfulness votes and only takes under consideration the content of the review and the star rating. We demonstrate that our proposed method outperforms the state-of-the-art review ranking techniques, through an extensive empirical evaluation.

The paper is organised as follows. In the next section we present an overview of the work that has been carried out in this space. Following this, we provide the motivation and technical details of

[1] http://jmcauley.ucsd.edu/data/amazon/

Proceedings of Recent Advances in Natural Language Processing, pages 959–967
Sep 1–3, 2021.
https://doi.org/10.26615/978-954-452-072-4_109

the proposed method. Finally, the results of the experimental evaluation are demonstrated followed by the discussion.

2 Related Work

Several approaches to automatically determining the helpfulness of online reviews have been explored in the past. In majority of the existing work, supervised machine learning models have been employed considering the problem as a predictive task (i.e. predict whether/how useful a review is) (Martin and Pu, 2014; Krishnamoorthy, 2015; Liu et al., 2017; Malik and Hussain, 2017; Singh et al., 2017; Wu et al., 2017; Enamul Haque et al., 2018; Lee et al.; Alsmadi et al., 2020). With supervised approaches, various types of features such as linguistic features (Krishnamoorthy, 2015; Malik and Hussain, 2017; Wu et al., 2017) or textual features (i.e. polarity, subjectivity, entropy and readability) (Singh et al., 2017; Lee et al.; Siering et al., 2018) are first extracted from the reviews, with machine learning methods used over such data to train a predictive model. In a few papers, unsupervised learning based approaches have been used to rank reviews based on their helpfulness or relevance (Tsur and Rappoport, 2006; Wu et al., 2011; Woloszyn et al., 2017). It is very apparent that the majority of work has been focused on using supervised machine learning and unsupervised learning has not been well explored in this space. Supervised learning methods depend on large, annotated datasets to train the model. Unfortunately, most of the publicly available online reviews datasets do not have labels related to their helpfulness. This makes the unsupervised learning based approaches much more attractive and hence it is the focus of our work.

A review ranking method based on unsupervised learning was proposed by Tsur and Rappoport (2006). The authors first created a corpus of core dominant terms for the reviews representing the key aspects relevant to a specific product. Dominant terms were obtained by computing the frequency of all terms in a reviews collection and re-ranking them by their frequency in the reference to the British National Corpus, a baseline corpus. They named the corpus as $virtual\ core$ (VC) review and represented it as feature vectors. Following this, they ranked the reviews according to their distance from the virtual core review vector. They assumed that the smaller the distance between a review and the virtual core review, the more relevant/helpful the review is.

Wu et al. (2011) proposed a ranking method to detect low quality reviews by using link analysis techniques. Three ranking algorithms have been implemented in their study which are (1) PageRank algorithm (Page et al., 1999), (2) HITS algorithm (Kleinberg et al., 2011), and (3) Length algorithm. First, they construct a graph for each review of a product where the vertexes are sentences in a review. Two directional edges between two vertexes are induced if they are similar according to specific POS tag i.e., nouns, adjectives, and verb. They compute the centrality scores of sentences using the PageRank and the HITS algorithms. A score for each review was obtained by summing all the centrality scores of all the sentences in a review and then rank the review based on the high centrality scores. The Length algorithm was used to rank all reviews based on total number of words. They count the number of words for each review and rank it based on the high score. The authors conjecture that high-quality review should contain more words than poor reviews. Two baseline methods were used for comparison in their experimental evaluation. From the evaluation, it could however be observed that their results were only slightly inferior in comparison to the baselines.

Inspired by the work proposed in (Martin and Pu, 2014), Woloszyn et al. (2017) developed MRR (Most Relevant Review), a novel unsupervised algorithm to rank reviews based on their estimated relevance. MRR algorithm consists of three steps: (1) First, they construct a graph of reviews for each product where the nodes are the reviews and the edges are defined based on the similarity between pairs of reviews. Two similarity scores are considered: cosine similarity between TF-IDF vectors computed for each review, and similarity between rating scores of reviews (i.e., rating scores from 1 to 5 given by reviewers), (2) This is followed by graph pruning that works by removing all edges with the similarity scores lower that the minimum threshold value, (manually set as β=0.85533), (3) Finally, the centrality scores are calculated for each review using PageRank algorithm. The authors hypothesise that the more central reviews should be considered as most relevant. Two state-of-the-art unsupervised learning (Tsur and Rappoport, 2006; Wu et al., 2011) and two supervised learning methods (i.e., one of the method use the same features

as (Wu et al., 2011)) were adopted in the experimental evaluation for comparison. Although, their results were lower than those obtained by supervised learning methods, they outperformed the two unsupervised learning based approaches.

In this work, we propose a new unsupervised method for ranking online reviews based on their helpfulness. Apart from the relevance (as in case of the existing unsupervised techniques), our method also considers the emotional intensity and the specificity of the reviews while assessing their helpfulness; this makes it unlike any of the approaches discussed above. For the text representation, we apply the Roberta state-of-the-art language model as opposed to TF-IDF used by the existing unsupervised methods.

3 Methodology

The key novelty of the proposed method is that it incorporates three different characteristics of online reviews while ranking them according to their helpfulness. We hypothesise that the helpfulness of a review should be determined based on the following features:

a) *Relevance*. Relevance indicates how well a review matches with customer's specific information needs (Liu et al., 2019). In other words, a helpful review should discuss the key features of a product, which are important for the future buyers (e.g., "The camera is easy to use, it is compact and perfect for travelling."). Review's relevance has been modelled by the existing work (Wu et al., 2011; Woloszyn et al., 2017) using graph composed of all reviews, their similarities and various centrality measures. It was assumed that the reviews that are the most central within the graph contain the most relevant information about the product. In our work, we take a similar approach, however, instead of graphs we used a simpler pair similarity based method.

b) *Emotional Intensity*. We hypothesise that emotions play an important part in a review process as they allow customers to express their feelings and experiences through opinions. Therefore, a good review should contain a good balance of both, facts and emotions. The relationship between helpfulness of online reviews and emotions have been explored by Malik and Hussain (2017) where they studied which emotions are important for helpfulness prediction. Martin and Pu (2014) used emotions to detect helpful reviews by applying different classification models (i.e., SVM, Random Forest, and Naïve Bayes) and demonstrated that their approach outperformed methods using POS tagging features. Emotion information has not been considered by any of the existing unsupervised methods. In this work, we propose to consider the level of emotions contained within a review as one of the factors in determining their helpfulness.

c) *Specificity*. A review of a product will be considered as useful/informative if it discusses various features of the products. In other words, instead of just expressing satisfaction/dissatisfaction from a product (e.g. "I hate this camera and would not recommend it"), it is much more helpful if the review explains what good or bad there is about the product (e.g. "The battery life is too short and the zoom is rather poor."). The greater number of different features is mentioned in a review, the more informative the review is for any potential buyer/customer. It should be noted that there is a distinct difference between the relevance and the specificity. With relevance, we assess whether the key characteristic of a product was discussed. While with specificity, we evaluate the level of details that was provided while discussing different features of a product. Following this reasoning, we propose to consider the number of different entities mentioned in the reviews while ranking the reviews based on their helpfulness. Such a specificity feature has also not been considered by any of the existing work.

Apart from the aforementioned characteristics, we also consider the star rating of the reviews in our ranking process. It has been demonstrated in the literature that the application of star rating is beneficial when evaluating the helpfulness of a review (Tsur and Rappoport, 2006; Schuff and Mudambi, 2010; Singh et al., 2017).

The pseudocode of our proposed methods is presented in Algorithm 1. The input to the method is a collection of reviews related to the same products. Each review contains the review text (r) and the star rating associated with this review (s). In the first step of the algorithm, the input reviews are ranked

separately on the basis of their relevance, emotional intensity and specificity. For the relevance ranking, we create a product-specific "summary document" (sum), which contains all individual reviews collated together. The summary document and each individual review are then converted into vectors using the RoBERTa pre-trained language model (Liu et al., 2019). For this part, any other embedding model (such as Word2Vec or Glove) can be considered. We used the RoBERTa model as it has recently received state-of-the-art results on many NLP benchmark datasets (Liu et al., 2019). Following this, the cosine similarity between each individual review and the summary document is calculated as its $relevance\_score$. It is worth noting that the proposed relevance ranking method is much simpler and faster than those of the baseline, which uses graphs to model similarity between reviews. With the second ranking, the reviews are ranked based on their emotional intensity. To identify different emotions in the reviews we used the DepecheMood++ (Araque et al., 2018) lexicon that contains 187942 words with 8 emotions intensity value for each word; this could be replaced with any emotion lexicon. For each review, we first identify all words which are present in the lexicon. Following this, all the intensity values assigned to those words in the lexicon are added together. The final $emotion\_score$ assigned to each review is the accumulation of intensity value by summing all emotion words within this review.

Finally, for the specificity ranking, we first apply name entity recognition and extract entities from the reviews using the NLTK library[2]. We calculate the $specificity\_score$ for each review as the sum of all entities that it contains. All the reviews are then sorted separately based on the three scores. As the outputs of the aforementioned steps, we obtained three rankings of the reviews, which were constructed based on the relevance, emotional intensity, and specificity of the reviews (lines 15-17).

As mentioned earlier, we also consider the star rating in our ranking method as it is considered as an good indicator of reviews helpfulness (Tsur and Rappoport, 2006; Schuff and Mudambi, 2010; Singh et al., 2017). We process the star rating by calculating the absolute deviation. The use of star rating deviation as a feature has been demonstrated in (Jindal and Liu, 2008; Lim et al., 2010; Jiang et al., 2013; Xu, 2013; Savage et al., 2015; Saumya and Singh, 2018) and some of the authors apply absolute deviation for the star rating (Danescu-Niculescu-Mizil et al., 2009; Mukherjee et al., 2013a,b; Runa et al., 2017) . First, we calculate the average of all star ratings of a product review (line 20). In the next step, for each review r_i, we calculate its *absolute deviation (AD)* from the average star rating as per Eq 1.

$$\begin{aligned} AD_i &= |s_i - avg| \\ RAD_i &= (1-\alpha) * AD_i \end{aligned} \tag{1}$$

where s_i is star rating for review r_i, typically between 1 and 5. Finally we calculate the *rating absolute deviation (RAD)* (line 22) as per equation 1, where α is used to balance the impact of the star rating on the final ranking and its value has been adopted from (Woloszyn et al., 2017), α = 0.867168.

The RAD value will be further included in the final ranking process together with the other three rankings as explained below. For combining the three rankings (i.e., relevance, emotional intensity and specificity), we applied the $z\text{-}score$ minimization method (Standard score, 2021). First, the mean (μ) and the standard deviation (σ) of the three ranking positions are computed for each review $r_i \in R$. In the next step we calculate the $z\text{-}score$ distance matrix calculating the $z\text{-}score$ for each review and every possible ranking position according to the following formula (lines 21-26):

$$z\text{-}score = |(p - \mu)/\sigma| \tag{2}$$

where p is the proposed ranking position.

The intuition behind this is to find the most statistically best ranking position by minimizing the aggregate z-score distance globally. The idea is from (Du et al., 2019) where they used exhaustive process for all possible features combination to find the best combination for helpfulness prediction. However, instead of using exhaustive process, we use a faster approach. The rows of matrix represent the number of reviews for each product and the columns represent the number of possible positions in the ranking (i.e., this is a squared matrix). Each cell of the matrix (c_{ij}) contains a $position\_score$ calculated for review r_i and position j using equation 2. The $z\text{-}score$ tells us how far each of the proposed ranking positions is from the mean position of the review. We further add the previously calculated RAD value to the $z\text{-}scores$

[2]https://www.nltk.org/book/ch07.html

Algorithm 1 The proposed algorithm for ranking online reviews based on their helpfulness

Require: List of reviews and their star ratings $R = \{(r_i, s_i)\}_{i=1..n}$ related to a single product
Ensure: The reviews ranked according to their helpfulness

1: $join\_review$ = join all reviews in R
2: sum = convert $join\_review$ into Roberta embedding
3: **for** each review r_i in R **do**
4: r_i_embed = convert r_i into Roberta embedding
5: $r_i\_relevance\_score = CosineSimilarity(sum, r_i\_embed)$
6: **for** each word w_j in r_i **do**
7: **if** w_j in $DepecheMood$++ **then**
8: $emotion\_scores[j]$ = sum all emotions intensities of w_j from $DepecheMood$++
9: **end if**
10: **end for**
11: $r_i\_emotion\_score = sum(emotion\_scores)$
12: $r_i\_specificity\_score$ = count number of entities in r_i
13: **end for**
14:
15: $rank_1$ = rank R based on $\{r_i\_relevance\_score\}_{i=1...n}$
16: $rank_2$ = rank R based on $\{r_i\_emotion\_score\}_{i=1...n}$
17: $rank_3$ = rank R based on $\{r_i\_specificity\_score\}_{i=1...n}$
18: $rank\_combine$ = combine all ranking ($rank_1$,$rank_2$,$rank_3$)
19:
20: avg_star = average of all star ratings $\{s_i\}_{i=1...n}$
21: **for** each r_i in R **do**
22: $RAD_i = (1 - \alpha) * |s_i - avg\_star|$
23: **for** j in $len(R)$ **do**
24: $position\_score[i][j] = \alpha * |z - score| + RAD_i$
25: **end for**
26: **end for**
27:
28: **for** $column$ in len(position) **do**
29: $sum\_score = 0$
30: **for** row in len(position) **do**
31: $sum\_score = sum\_score + position[row][column]$
32: **end for**
33: $total\_score = (sum\_score) - min(position[row][column])$
34: **end for**
35: select column where $total\_score$=max(total_score)
36: assign review at the position where $position\_score$=min($position\_score$)
37: delete the column and row and repeat step 28-37 until convergence

in the matrix. The final step is to find out which set of ranking positions of the reviews gives the lowest total z-$score$ distance. For this purpose we use an iterative solution (lines 28-37) which is explained below. For each column, we sum its values and subtract the minimum value from this column, obtaining a score referred to as $total\_score$. Then, we select the column with the maximum $total\_score$. After that, we find the minimum value in that column. The corresponding review is then assigned to the position. The next step is to delete the column and row and repeat the same for the rest of reviews until all the positions are filled. For instance, if the largest $total\_score$ is at column 4 and the minimum $position\_score$ on that column belongs to $review_1$, then assign the $review_1$ to that position, i.e 4 which is now the re-ranked position of $review_1$.

4 Experimental Evaluation

4.1 Datasets

For the purpose of this study, we use dataset from Amazon[3] reviews (from May 1996 – July 2014) for four categories of products, namely (1) Electronics, (2) Books, (3) CDs Vinyls and (4) Movies TV products, with raw data size of 1.48 GB, 9.46 GB, 1.33 GB, and 1.93 GB respectively. In this study, we only use four features: ASIN as a unique product id, ReviewText for performing the three rankings, Overall in order to include the rating star in the final ranking and Helpfulness Votes for the evaluation purposes. All the data has been processed and filtered according to the following steps. First, the product should have minimum 30 reviews. Each review should contain minimum four sentences. The review should have minimum five helpfulness votes. The details regarding the size of each dataset before and after pre-processing are listed in Table 1.

Dataset	**Before Pre-processing**	**After Pre-processing**
Books	8,898,041 rev	109,099 rev
Electronics	1,689,188 rev	8,134 rev
CDs and Vinyls	1,097,592 rev	11,448 rev
Movies & TV	1,697,533 rev	31,035 rev

Table 1: Amazon dataset

[3] http://jmcauley.ucsd.edu/data/amazon/

4.2 Baseline and Evaluation

As a baseline, we implemented the state-of-the-art unsupervised ranking method MRR (Woloszyn et al., 2017), which has been described in Section 2. This is the most recent work that has been done in this space using unsupervised learning. In the original paper (Woloszyn et al., 2017), the results were also compared with two other unsupervised approaches and supervised models and it was demonstrated that MRR outperformed others baseline (Tsur and Rappoport, 2006; Wu et al., 2011). Therefore, we only use MRR as the baseline.

For further evaluation, we also explored different variants of our proposed methods. We considered using the summary of the reviews instead of their full content. The summaries of the reviews were first obtained with the SUMY library[4] and then provided as an input to the algorithm describe in Algorithm 1. We also evaluated the performance of our method using only the relevance ranking. In this way we wanted to validate the usefulness of emotional intensity and specificity rankings in the process. Finally, we considered the performance of our method without application of the rating star.

For the evaluation, we use NDCG (Järvelin and Kekäläinen, 2002) metric. NDCG measures the quality of ranking or recommendation system using list positions. For the purpose of ranking evaluation with NDCG, we use the helpfulness vote's feature as the relevance value to determine the ranking. The relevance value for the NDCG is calculated based on the helpfulness vote obtained from Amazon using the gold standard as in (Woloszyn et al., 2017). The gold standard formula is in Eq 3 :

$$H(r \in R) = \frac{vote_{+}(r)}{vote_{+}(r) + vote_{-}(r)} \quad (3)$$

Where r is a review, $vote+$ is the number of customers who voted for the review as being helpful and $vote-$ is for the customers who votes it as being unhelpful. $H(r \in R)$ is then used as the relevance value for the NDCG.

5 Result and Discussion

The results obtained by each of the evaluated methods on each of the four datasets are presented in Tables 2-5.

Each table demonstrates the result obtained by each of the methods with and without incorporating the star rating in the process. The first row in each of the tables refers to the results obtained by the state-of-the-art (MRR) unsupervised baseline (Woloszyn et al., 2017). Rows 2 and 3 show the results obtained by our method based only on the relevance ranking and using the full text or the summary of the reviews, respectively. The last two rows refer to the results obtained by the method when all three rankings were incorporated in the process. We evaluate our ranking quality using NDCG metrics and we take four different ranking positions. Those are NDCG@3, NDCG@5, NDCG@7, and NDCG@10 where the number after the NDCG@ represent the number of reviews taken for evaluation from the top rank position.

From the results presented in Tables 2-5 we can observe that for each of the four datasets, the proposed method performed better when all three ranking were incorporated. This indicated that the emotional intensity and the specificity of a review are useful when determining its helpfulness. It can also be noted that our method obtained better results when the star rating is used when creating the final ranking. The difference is particularly apparent for the Books dataset. Finally, we can see that the proposed method performed better when applied with the full review content (Relevance(full_text)+emotion+specify) than with the summary (Relevance(summary)+emotion+specify) with three out of four datasets. The only case when using the summaries of the reviews made a positive difference is the CDs & Vinyls dataset. Looking at the overall results (Both with and without rating star) we can conclude that our proposed method performs best when each of the three rankings is performed on the full reviews' content and when the rating star is considered.

When comparing the proposed method with the baseline (MRR), we can observe from Tables 2-5 that we obtained better results according to each of the evaluation scores (NDCG@3, NDCG@5, NDCG@7, NDCG@10) in all datasets. For example, Table 2 shows our combination ranking score (relevance+emotion+specify) at NDCG@3, NDCG@5, NDCG@7, NDCG@10 are 0.982, 0.977, 0.974, and 0.972, respectively which improves by 1% from the baseline. On other datasets, the improvement is showing up to 2% compare with the baseline at NDCG@5 on Books dataset and NDCG@3 on Movies & TV dataset. To further evaluate the proposed method in comparison to the baseline, we assess whether the differences in their

[4]https://pypi.org/project/sumy/

Method	with rating star				without rating star			
	NDCG@3	NDCG@5	NDCG@7	NDCG@10	NDCG@3	NDCG@5	NDCG@7	NDCG@10
MRR	0.97	0.966	0.963	0.963				
Relevance (summary)	0.958	0.952	0.949	0.947	0.958	0.952	0.949	0.947
Relevance (full_text)	0.968	0.961	0.958	0.956	0.968	0.961	0.958	0.956
Relevance(full_text)+emotion+specify	**0.983**	**0.977**	**0.975**	**0.973**	**0.981**	0.975	0.972	0.97
Relevance(summary)+emotion+specify	0.979	0.975	0.973	0.97	0.979	0.975	0.972	0.97

Table 2: Evaluation metric Electronics dataset

Method	with rating star				without rating star			
	NDCG@3	NDCG@5	NDCG@7	NDCG@10	NDCG@3	NDCG@5	NDCG@7	NDCG@10
MRR	0.957	0.944	0.94	0.936				
Relevance (summary)	0.959	0.948	0.943	0.939	**0.959**	**0.948**	**0.943**	**0.939**
Relevance (full_text)	0.958	0.946	0.94	0.937	0.958	0.946	0.94	0.937
Relevance(full_text)+emotion+specify	**0.969**	**0.957**	**0.952**	**0.946**	0.958	0.945	0.94	0.936
Relevance(summary)+emotion+specify	0.968	0.957	0.951	0.946	0.957	0.945	0.939	0.935

Table 3: Evaluation metric Books dataset

Method	with rating star				without rating star			
	NDCG@3	NDCG@5	NDCG@7	NDCG@10	NDCG@3	NDCG@5	NDCG@7	NDCG@10
MRR	0.961	0.947	0.94	0.935				
Relevance (summary)	0.96	0.945	0.939	0.935	0.96	0.945	0.939	0.935
Relevance (full_text)	0.962	0.948	0.941	0.938	0.962	0.948	0.941	0.938
Relevance(full_text)+emotion+specify	0.968	0.957	0.952	0.949	0.967	0.955	0.949	0.947
Relevance(summary)+emotion+specify	**0.97**	**0.96**	**0.955**	**0.951**	0.967	**0.956**	**0.951**	**0.948**

Table 4: Evaluation metric CD & Vinyls dataset

Method	with rating star				without rating star			
	NDCG@3	NDCG@5	NDCG@7	NDCG@10	NDCG@3	NDCG@5	NDCG@7	NDCG@10
MRR	0.953	0.941	0.936	0.931				
Relevance (summary)	0.949	0.936	0.928	0.923	0.949	0.936	0.928	0.923
Relevance (full_text)	0.948	0.931	0.926	0.921	0.948	0.931	0.926	0.921
Relevance(full_text)+emotion+specify	**0.968**	**0.956**	**0.95**	**0.945**	**0.963**	**0.951**	0.945	**0.942**
Relevance(summary)+emotion+specify	0.966	0.954	0.949	0.944	0.961	0.95	0.945	0.94

Table 5: Evaluation metric Movie & TV dataset

performances are statistically significant using the T-test. According to 0.05 significance level, the difference was statistically significant in 11 out of 16 cases. As the 16 cases we consider four different performance measures. (NDCG@3, NDCG@5, NDCG@7, NDCG@10) calculated for each of the four datasets. The results obtained by our method on the books and CDs & Vinyls datasets are numerically superior in all four cases. As for electronic dataset, only NDCG@3 and NDCG@5 are statistically different, while on Movie & TV dataset, only one result that shows the difference in statistic, it is NDCG@10.

6 Conclusion

This paper addresses the problem of online reviews ranking according to their helpfulness. We propose an unsupervised method, which first ranks the reviews based on their relevance, emotional intensity and specificity and then combine them in order to obtain the final helpfulness ranking. The performance of the method on four datasets that were created for the purpose of this study was evaluated using the NDCG metric. It was demonstrated that the method outperformed the state-of-the-art unsupervised online review ranking method proposed in (Woloszyn et al., 2017) in every case. In the future, we want to improve our ranking system by applying different features and ranking method. Some features such as linguistic features, positive and negative emotion, or topic sentences may be explore in the ranking system. Moreover, different combination ranking method such as Schulze (Schulze, 2018) or Borda count (Emerson, 2013) or another ranking method could be explored to improve the performance.

References

Abdalraheem Alsmadi, Shadi Alzu'bi, Bilal Hawashin, Mahmoud Al-Ayyoub, and Yaser Jararweh. 2020. Employing Deep Learning Methods for Predicting Helpful Reviews. *2020 11th International Confer-*

ence on Information and Communication Systems, ICICS 2020, pages 7–12.

Amazon.com. Amazon's Top Customer Reviewers.

Oscar Araque, Lorenzo Gatti, Jacopo Staiano, and Marco Guerini. 2018. DepecheMood++: a Bilingual Emotion Lexicon Built Through Simple Yet Powerful Techniques.

Robert B Cialdini. 1987. *Influence*, volume 3. A. Michel Port Harcourt.

Cristian Danescu-Niculescu-Mizil, Gueorgi Kossinets, Jon Kleinberg, and Lillian Lee. 2009. How opinions are received by online communities: a case study on amazon. com helpfulness votes. In *Proceedings of the 18th international conference on World wide web*, pages 141–150.

Jiahua Du, Jia Rong, Sandra Michalska, Hua Wang, and Yanchun Zhang. 2019. Feature selection for helpfulness prediction of online product reviews: An empirical study. *PloS one*, 14(12):e0226902.

Peter Emerson. 2013. The original borda count and partial voting. *Social Choice and Welfare*, 40(2):353–358.

Md Enamul Haque, Mehmet Engin Tozal, and Aminul Islam. 2018. Helpfulness prediction of online product reviews. *Proceedings of the ACM Symposium on Document Engineering 2018, DocEng 2018*.

Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems*, 20(4):422–446.

Bo Jiang, B Chen, et al. 2013. Detecting product review spammers using activity model. In *Proceeding of International Conference on Advanced Computer Science and Electronics Information (ICACSEI 2013)*, pages 650–653. Citeseer.

Nitin Jindal and Bing Liu. 2008. Opinion spam and analysis. In *Proceedings of the 2008 international conference on web search and data mining*, pages 219–230.

Jon M Kleinberg, Mark Newman, Albert-László Barabási, and Duncan J Watts. 2011. *Authoritative sources in a hyperlinked environment*. Princeton University Press.

Srikumar Krishnamoorthy. 2015. Linguistic features for review helpfulness prediction. *Expert Systems with Applications*, 42(7):3751–3759.

Pei Ju Lee, Ya Han Hu, and Kuan Ting Lu.

Ee-Peng Lim, Viet-An Nguyen, Nitin Jindal, Bing Liu, and Hady Wirawan Lauw. 2010. Detecting product review spammers using rating behaviors. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 939–948.

Haijing Liu, Yang Gao, Pin Lv, Mengxue Li, Shiqiang Geng, Minglan Li, and Hao Wang. 2017. Using Argument-based features to predict and analyse review helpfulness. *EMNLP 2017 - Conference on Empirical Methods in Natural Language Processing, Proceedings*, pages 1358–1363.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. (1).

M. S.I. Malik and Ayyaz Hussain. 2017. Helpfulness of product reviews as a function of discrete positive and negative emotions. *Computers in Human Behavior*, 73:290–302.

Lionel Martin and Pearl Pu. 2014. Prediction of helpful reviews using emotions extraction. *Proceedings of the National Conference on Artificial Intelligence*, 2:1551–1557.

Arjun Mukherjee, Vivek Venkataraman, Bing Liu, and Natalie Glance. 2013a. What yelp fake review filter might be doing? In *Seventh international AAAI conference on weblogs and social media*.

Arjun Mukherjee, Vivek Venkataraman, Bing Liu, Natalie Glance, et al. 2013b. Fake review detection: Classification and analysis of real and pseudo reviews. *UIC-CS-03-2013. Technical Report*.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab.

Dao Runa, Xianguo Zhang, and Yongxin Zhai. 2017. Try to find fake reviews with semantic and relational discovery. In *2017 13th International Conference on Semantics, Knowledge and Grids (SKG)*, pages 234–239. IEEE.

Sunil Saumya and Jyoti Prakash Singh. 2018. Detection of spam reviews: a sentiment analysis approach. *Csi Transactions on ICT*, 6(2):137–148.

David Savage, Xiuzhen Zhang, Xinghuo Yu, Pauline Chou, and Qingmai Wang. 2015. Detection of opinion spam based on anomalous rating deviation. *Expert Systems with Applications*, 42(22):8650–8657.

David Schuff and Susan Mudambi. 2010. What Makes a Helpful Online Review? A Study of Customer Reviews on Amazon.com1 By:. 34(1):185–200.

Markus Schulze. 2018. The schulze method of voting. *arXiv preprint arXiv:1804.02973*.

Michael Siering, Jan Muntermann, and Balaji Rajagopalan. 2018. Explaining and predicting online review helpfulness: The role of content and reviewer-related signals. *Decision Support Systems*, 108:1–12.

Jyoti Prakash Singh, Seda Irani, Nripendra P. Rana, Yogesh K. Dwivedi, Sunil Saumya, and Pradeep Kumar Roy. 2017. Predicting the "helpfulness" of online consumer reviews. *Journal of Business Research*, 70:346–355.

Jared Spool. 2009. The Magic Behind Amazon's 2.7 Billion Dollar Question—UX Articles by UIE.

Standard score. 2021. Standard score - Wikipedia.

Oren Tsur and Ari Rappoport. 2006. R EV R ANK : a Fully Unsupervised Algorithm for Selecting the Most Helpful Book Reviews. *Artificial Intelligence*, pages 154–161.

Vinicius Woloszyn, Henrique D.P. Dos Santos, Leandro Krug Wives, and Karin Becker. 2017. MRR: An unsupervised algorithm to rank reviews by relevance. *Proceedings - 2017 IEEE/WIC/ACM International Conference on Web Intelligence, WI 2017*, pages 877–883.

Jianwei Wu, Bing Xu, and Sheng Li. 2011. An unsupervised approach to rank product reviews. *Proceedings - 2011 8th International Conference on Fuzzy Systems and Knowledge Discovery, FSKD 2011*, 3:1769–1772.

Shih-Hung Wu, Yi-Hsiang Hsieh, Liang-Pu Chen, Ping-Che Yang, and Liu Fanghuizhu. 2017. Temporal Model of the Online Customer Review Helpfulness Prediction. pages 737–742.

Chang Xu. 2013. Detecting collusive spammers in online review communities. In *Proceedings of the sixth workshop on Ph. D. students in information and knowledge management*, pages 33–40.