Review Opinion Diversification - Exhaustive Coverage Ranking

Arjun Ahuja
Institute Institute of Technology
Hyderabad, Telangana 502285
cs14btech11004@iith.ac.in

Safwan Mahmood
Institute Institute of Technology
Hyderabad, Telangana 502285
es14btech11017@iith.ac.in

ABSTRACT

We study the problem of exhaustive coverage ranking for reviews in a setting where there exists a huge collection of reviews, and reviews may belong to more than one category. We present two approaches to diversify search results that aims to minimize the risk of dissatisfaction of the average user. For the second approach we will further use the algorithm for diversifying results based on a query given in [2].
Furthermore, we use generalized classical IR metrics introduced in [2] to take into account our diversifying strategy, these include NDCG-IA, MRR-IA, and MAP-IA, to explicitly account for the value of diversification.

1 INTRODUCTION

Ranking the reviews has been a well known problem in Information Retrieval and there has been a lot of well known proposals to solve this problem. Though these methods give us the top best reviews, they fail to catch all the essence of all the reviews i.e. fail to diversify the reviews.In this report we have come up with one approach and modified another to fit our needs to solve the objective.
We have a large collection of reviews consisting of both positive, negative and neutral opinions.

2 VECTOR REPRESENTATION

We use the tf-idf scores for each word in review and get a vector for each of the reviews. The vector for each review was found to be very sparse, as we were using a dictionary of around 100000 reviews which had around 20000 unique words but not many unique words per document.
If the collection size is large enough(We used several examples some involving less number of reviews some involving more) then we apply a dimensionality reduction trick called the PCA referenced from [1]. We found out that as the collection size increase, PCA's performance improves.
We don't suggest using dimensionality reduction if the collection size is small, as the program becomes quite slow ,with PCA becoming the bottleneck of the whole program.

3 CATEGORIZING - K MEANS

The collection of reviews is categorized into their respective categories using K-Means. We use the silhouette scores along with the elbow method to get the best clustering and optimum number of clusters. Every review falls into their respective category clusters. The number of categories in the collection is reflected by number of clusters obtained from the optimal silhouette score.
We devise a method to automate the process of finding the elbow to calculate an approximate elbow, which might change depending on the number of reviews in the collection.
We find the elbow in graph of silhouette score vs number of cluster. When we start to approach a particular value(the value of silhouette scores starts decreasing with lesser magnitudes i.e. $\frac{\partial s}{\partial n}$, where n is the number of clusters and s is the silhouette score obtained, becomes very less for a particular value of number of clusters. We choose that value as the value of number of clusters.)

4 REVIEW SCORE CALCULATION

We define :
Vector $C : N_c x N_d$ where N_c : No of categories, N_d : Dimension of review vector.
Vector $R : N_r x N_d$ where N_r : No of reviews
Scores $S : N_r x N_c$ where N_r : No of reviews.
To calculate the probability of a review belonging to ith category: It is the distance between the review and the ith category centroid.

4.1 Algorithm

$for(i : review)$:
    $for(j : categories)$:
        $S[i][j] := dist(R[i], C[j])$

After getting the scores of a review belonging to each class, we can take two approaches.

5 APPROACH 1 : WEIGHTED SCORES

We take the weighted average of the scores of all the category scores giving maximum weight-age to the greater category scores. We store the new values for each review and sort them in descending order of the scores.
$for(i : review)$:
    $for(j : categories)$:
        $Z[i] += S[i][j]$

5.1 Top-k Values

We return the top-k reviews according to the above calculated Z score.

5.2 Other Ways

- Take average of all category scores.
- Use a learning algorithm to get best fit weights by training.

6 APPROACH 2 : USING INTENT AWARENESS ALGORITHM

The algorithm is referenced from as mentioned in [2]. The algorithm requires the use of a query. In our scenario we won't be dealing with queries. Instead of explicitly defining a query we define the probabilities related to that query, i.e, P(c|q), which means to find

probability of a category given some query. As it is a diversification problem, we can assume this probability to be same for each query. We define P(c|q) ,probability of category given a query as

$$\frac{1.0}{numberOfClusterCenters}$$

where number Of cluster centers are returned by K-Means, that is we define each category to have the same probability for complete diversification. The rest of the algorithm follows from as mentioned in [2]

7 RESULTS

We compare both the approaches on various evaluation metrics.DCG,NDCG,MA are classical metrics while NDCG-IA, MAP-IA are introduced in [2]. NDCG-IA is NDCG penalized with P(c|q), assume n to be the number of cluster centers obtained in KMeans process i.e.

NDCG-IA = $\sum_{c=1}^{n} P(c|q) \times NDCG(Q, c)$

which is NDCG considering only one category, we can implement this into our own program using 1 is the initialization for category c and 0 for all other categories, which is different from previously mentioned value of $\frac{1.0}{n}$
similarly for MAP-IA we have

MAP-IA = $\sum_{c=1}^{n} P(c|q) \times MAP(Q, c)$

7.1 Plotting Precision-Recall Curves

Approach 1: Plot for the approach 1. **Approach 2:** Plot for the approach 2.

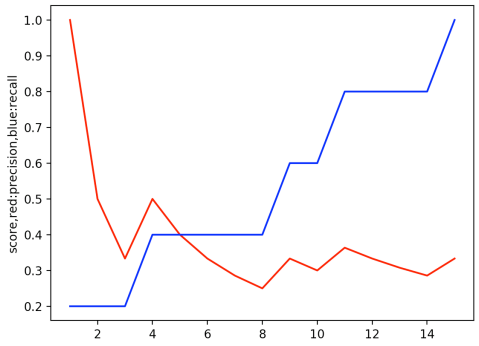

Figure 1: Approach 1: Precision vs Recall

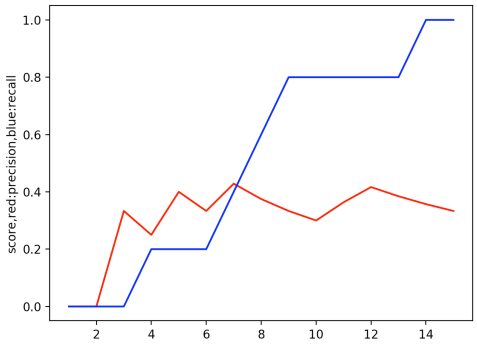

Figure 2: Approach 1: Precision vs Recall

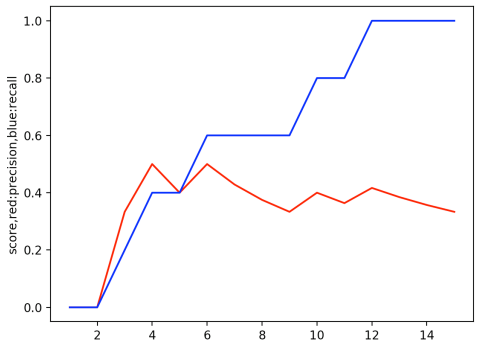

Figure 3: Approach 2: Precision vs Recall

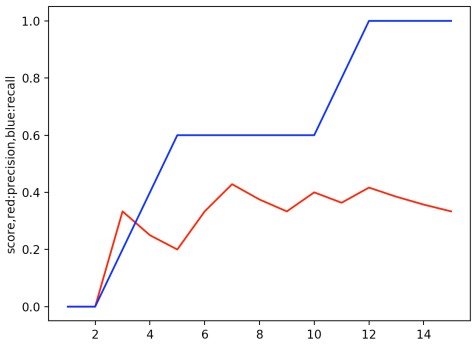

Figure 4: Approach 2: Precision vs Recall

7.2 Tables

We compare the metric values like DCG, NDCG, MAP, NDCG-IA, MAP-IA. As NDCG-IA and MAP-IA are only defined for approach 2, its values for approach 1 are same as that for NDCG and MAP respectively. Refer Table 1. for the comparison of values between approach 1 and 2.

Table 1: Approach 1 vs Approach 2

DCG	NDCG	MAP	NDCG-IA	MAP-IA
1.43	0.485	0.5	0.485	0.5
1.31	0.44	0.44	0.72	0.76

8 CONCLUSIONS

- We provided two approaches to get diversifying exhaustive reviews. We see that in case of approach two(IA select) NDCG values are less than what we achieved in approach 1. But the approach 2 have rather high NDCG-IA values, which depicts the degree of diversification.We see that approach 2 performs better diversification when compared to approach one.
- While this also depicts that the top-k results may not be the most diverse results as for approach 2 we obtained NDCG is low but NDCG-IA as a high value.
- We also get that as NDCG and NDCG-IA scores for approach 1 are always same because of their independence with the category probability, but we can modify that algorithm to include category probability by modifying our scoring strategy by assigning scores to a point based on its distance from only one category and then finding the score, but his could be tested in future work, this method can also be used for Approach 2.

9 REFERENCES

[1] Herve Abdi and Lynne J. Williams. -. Principal component analysis. (-).

[2] Sreenivas Gollapudi Alan Halverson, Rakesh Agrawal and Samuel Ieong. 2009. Diversifying Search Results. (2009).