Microblog Retrieval: Real Time Adhoc Retrieval

CSC849 Term Project
Fall 2016

Xuan Zhang
916409525

- INTRODUCTION
- MY APPROACH
- EXPERIMENTS & ANALYSIS
- CONCLUSIONS

INTRODUCTION

- Twitter is becoming more popular nowadays, and it is one of biggest social media around the world.

- However, microblog are different from traditional information retrieval.

- First, tweets are very short, less than 140 words.

- Second, information on microblogging websites rapidly updates.

- Third, short tweets have distinct features and structures.

MY APPROACH

- Analysis of the Given Materials

- My Expansion Method

Analysis of the Given Materials

- 7 folders, not cover all the documents. (total 2704533 tweets).

- Not all the relevant documents included in the given materials

- 11 queries don’t exist relevant feedback in “qrels”. (Query: 11, 12, 15, 18, 24, 49, 50, 53, 63, 75, 76)

- 5 queries’ relevant feedback not in the given materials (Query: 30, 55, 58, 66, 80)

- Effective queries: 110 - 11 - 5 = **94 queries**

Analysis of the Given Materials

- Relevant Feedback: **total 1650** relevant feedback tweets in the given materials for these **94 queries**

- Average: 1650 / 94 = 17.55 relevant tweets per query

- Actually, the median value is 12 relevant tweets per query

- Due to the limited resource, I think P@10 or P@15 is more meaningful than P@30 in this project, though many paper preferring P@30.

MY APPROACH

- I use Indri-5.11 and trec_eval.9.0 and my own programming in Java in this project.

- I do query expansion based on tf-idf score:

- For each documents: weight(t, d) = [1 + log10(**tf(t,d)**)]* log10(N/**df(t)**)

- We can see that tf(t, d) is related to the relevant documents, but df(t) is related to the all documents, relevant or non relevant documents.

- So, we can deal with them **separately**.

MY APPROACH

- I use Indri to get the top 20 pseudo relevant tweets based on indri scores. And then extract these tweets from microblog2011 and build inverted index for each query.

- Thus, we can get tf(t, d).

MY APPROACH

- Traverse all the files in microblog2011, from 7 folders I get 7 files, computing the document frequency for each term in each folder (inverted index 1 without position list). (Because the files are too big, So I use 7 files for each folder respectively.)

- And in the relevant documents, if we need one certain term, the program would get the document frequency by adding the 7 doc Freq together (inverted index 2 without position list).

MY APPROACH

- When I get weight(t, d) for each document on the term, I add them together and then get the final score for that term.

- Rank the scores of different terms, get the top term to do the expansion.

- From this project, I selected the top 10 terms based on their tf-idf score for each query.

EXPERIMENTS & ANALYSIS

- Ideal case (System Limitation)
- Formal Run
- Small Modification

Ideal Case

- According all the relevant documents in “qrels” and in our given documents (1650 tweets totally), I build the inverted index for each query, and do the expansion.

- Definitely in the real life formal run would not be better than this “best performance” Ideal Case in my system. But we can know the limitation of using Indri in this project.

Ideal Case

	Original	Ideal Case
P@5	0.3830	0.8191
P@10	0.3511	0.6319
P@15	0.3170	0.5340
P@20	0.2920	0.4585
P@30	0.2358	0.3596
MRR	0.6326	0.9840
MAP	0.1855	0.3755

Formal Run

- Step 1: Use Indri to get top 10 tweets for each query.
- Step 2: Extract them from microblog2011 and build inverted index with position list, and get document frequency information from the big inverted index without position list. Get the tf-idf score for each term.
- Step 3: get the new query with the expanded terms for each query
- Step 4: use this new query to run indri again and get evaluation.

Formal Run

- Now I find out there are a couple of queries that cannot be easy to get any true relevant documents. (Query: 16, 33, 46, 100, 85, 93). Why?

- Some of them are only 1 relevant feedback, and we need external evidence. Some of them should be added proximity operator in indri.

Small Modification

- For example: Query 100, we should use
- #filreq(#less(requested_id 3473774846411571 2) #syn(#1(Republican National Committee) rnc))
- For Query 85, we should use
- #filreq(#less(requested_id 3476483255304192 0) #combine(#1(Best Buy) improve sales))
- For Query 93, we should use
- #filreq(#less(requested_id 3493602675340492 8) #combine(#1(fashion week) in #syn(NYC #1(new york))))

Small Modification

- For Query 16: release of #1(Known and Unknown)
- I use external evidence, I download the wikipedia for this wikipedia, and get the top 5 frequented words: rumsfeld bush president we know
- I add the top 1 words “rumsfeld” to the original query, and then the results from indri returns the only 1 relevant feedback of this query.

Add External Evidence

- I use the wikipedia in the google search result, and only copied the old version which is before the query date.

- If there is no wikipedia page in the google search result, I will use the first relevant page and pay attention that the date should be before the query date.

Formal Run

	Original	Expansion based on Indri	Expansion on Indri+External
P@5	0.3830	0.4000	0.3957
P@10	0.3511	0.3559	0.3585
P@15	0.3170	0.3154	0.3106
MRR	0.6326	0.5615	0.5752

ANALYSIS

- For expansion from Indri, the improvement is not obvious, only P@5 improve 4%, and some other features such as P@15 and MRR even worse than original queries.

- I think this is because even in the top 10 tweets from indri, the precision is not high, only 0.3511. When I do expansion, there is lots of non relevant tweets involving.

ANALYSIS

- Using external evidence in my method is useless for improving the precision at any level.
- I guess it is because of the following reasons:
- 1. Frequently used terms are non-relevant terms, such as “we”, “they”.
- 2. The relevant terms expanded from external are not included in the limited tweets in the project.
- 3. Expansion terms from external are duplicated as expansion from indri, such as Query 2: “qatar”, “cup”, “world”.
- 4. I should download at least 10 pages for each query, only 1 page is not enough.

Conclusions

- By the means of my expansion method, no matter from tweets itself or from external, the improvements is not obvious.
- It might be improved by other methods:
- Using more proximity operator in the original queries.
- Using large amount of external evidence.

Thanks!