DIVTEXT

TEXT RECOMMENDATION SYSTEM BASED ON RESULT DIVERSIFICATION

Bahaeddin ERAVCI Emre YILMAZ Izzeddin GUR Mehmet GUVERCIN

Agenda

- Problem: diverse text recommendation
- Motivation
- Applied methodology
- Experiments on Jester Joke dataset
- Experimental results

PROBLEM DESCRIPTION

• We have user queries, and we want to find similar but diverse documents to offer

Similar Artists



1	Bob Dylan
2	Radiohead
3	Led Zeppelin
3	The Rolling Stones
5	Pink Floyd
6	David Bowie
7	The Who
8	John Lennon

• There is a tradeoff between similar and diverse documents



Jester 4.0

Jokes for Your Sense of Humor

Home

About Register (Optional) Login (Optional)

Is there truth behind all humor, or is it the other way around?

Jester uses a collaborative filtering algorithm called Eigentaste to recommend jokes to you based on your ratings of previous jokes.

Update:

Jester now uses Eigentaste 5.0, an algorithm that improves upon Eigentaste. In addition, user registration has been made optional.

To learn more about Eigentaste, go here.

Instructions:

After telling us where you heard about Jester, click on the "Show Me Jokes!" button. You'll be given a set of 8 jokes to rate. After that, Jester will begin recommending jokes that have been personalized to your tastes.

Please rate the jokes by clicking on the rating bar on the bottom on the screen. Click to the left of the rating bar if the joke makes you wince or to the right if it makes you laugh uncontrollably, and anywhere in between if appropriate. If you have seen or heard a joke before, please try to recall how funny it was to you the first time you heard it and rate it accordingly.

Note: Some of the jokes in our database may be considered by some to be offensive. If you are likely to be offended by mild ethnic, sexist, or religious jokes, please do not continue. Thank you.

Where did you hear about Jester?

Show Me Jokes!

MOTIVATION

- o Big data
 - The fastest increasing quantity on this planet is the amount of information we are generating
- Large availability
- Partial knowledge about data
 - if you don't know the alternatives you don't know to search for exactly



MOTIVATION

• So, we need results that are both *similar* to the query yet *different* from each other i.e. **diversity**

• This gives a chance

- to do exploratory search
- see different perspectives of the query
- for a better user satisfaction

METHODOLOGY

- Given a user profile
- Goal: find similar but diverse k documents for each user

Steps of methodology:

- 1) Extract document-term matrix
- 2) Generate user profiles
- 3) Get top k documents from diverse system
- 4) Evaluate results based on the user satisfaction

- Clearing punctuation marks
- Removing stop words
- Stemming
 - Tartarus Snowball
- Removing unnecessary words
- Constructing document-term frequency matrix
- o 1115 terms 100 documents

• To extract queries for users

- Get vector of top m documents that user most likes
- Take the union of document vectors

$$q = \bigcup_{i=1}^{m} d_i$$

• Calculate $s(q, d_i)$ for all i

•
$$s(q, d_i) \stackrel{\text{\tiny def}}{=} \frac{2 * q \cap d_i}{|q| + |d_i|}$$

• Put most similar one into RelDocs

- Calculate how much the document d_i is diverse
 - $div(di, RelDocs) \stackrel{\text{def}}{=} \min[1 s(d_i, d_j)] \quad \forall dj \in RelDocs$
- Calculate the relevance of the document d_i
 - $\operatorname{Rel}(d_i, q, \operatorname{RelDocs}) = \alpha * s(q, d_i) + (1 \alpha) * \operatorname{div}(d_i, \operatorname{RelDocs})$

- α is the parameter to adjust similarity and diversity
- If $\alpha = 1$, then only similar documents will be recommended
- Put document with maximum Rel into RelDocs
- Stop after k iterations
- Report RelDocs

- At the end of the step 3 we have k recommended documents
- Evaluate user satisfaction(unweighted)
 - Sum user ratings for k documents
 - Without ranking
- Evaluate user satisfaction(weighted)
 - \sum (user rating) *(reverse rank)
 - Reverse rank = k rank + 1

EXPERIMENTAL SETUP

• Dataset:

- 100 jokes are rated by 7200 users with scale [-10,10]
- α is incremented from 0.0 to 1.0 by 0.1
- Min, mean, max satisfaction is calculated for each α
- **o** k = 10
- m = 10

EXPERIMENTAL RESULTS



EXPERIMENTAL RESULTS



CONCLUSION

- Diverse results give better user satisfaction with respect to similarity search
 - User satisfaction increases 20x
- User satisfaction increases until an optimum alpha value
 - Optimum alpha value is found 0.6 for this dataset
- Diversity even works for the most unsatisfied users

FUTURE WORK

- Recommendation system based on collaborative filtering
- Learn α for each user
- Using clustering analysis for diversity

Q & A