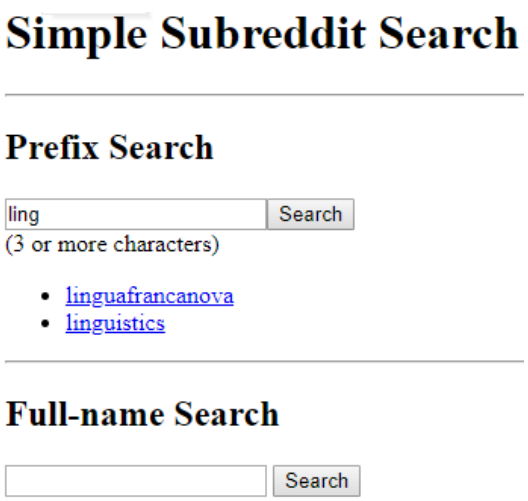
IR Reddit Website Writeup

Kendra Chalkley
CS535 Information Retrieval

1 Introduction

The goal of this project was to create a website which users could browse to learn about language used on subreddits as described by LDA generated topics. The site includes three page types: the initial search page, the subreddit pages, and the topic pages. These pages are populated from a LevelDB database and generated in flask. The following sections elaborate on each of the page types and elements within the pages as well as how each section could be improved in the future.

2 Home Page



The home page of the site allows users to navigate directly to a subreddit of interest by typing the full name of the subreddit into 'Full-name Search' field. Alternatively, using the 'Prefix Search' returns all subreddits that start with the same 3 or more characters, in case the user cannot recall the exact name of the subreddit they are interested in.

The query bar only accepts exact matches to subreddit names and is case sensitive. It is intended for users who know what they are looking for and don't need any extraneous information distracting them from their information needs. Because successful use of the query bar will return only one result, the page automatically redirects to the appropriate subreddit page upon finding a match between user input and database keys.

The prefix field requires input of at least three characters and returns a (case sensitive) list of all subreddit names with that prefix. For example, a prefix search for 'dep' returns:

```
depaul
depechemode
dependa
dependent_types
depravedmemes
depressed
depressing
depression
depression_de
depression_help
depressionregimens
depthtub
```

and 'Dep' returns:

```
Depersonalization
Depresion
Depressed_Writing
Depressed_supporters
DepressingStories
DepthHub
```

Ideally, this field would not be case sensitive. Subreddit capitalization is consistently written by reddit.com in the same way it was when the subreddit was created, but duplicate reddits distinguished only by capitalization are not allowed. (Note that /r/depression and /r/Depresion are also distinguished by a spelling difference.) LevelDB, however, does not provide an easy method of including case folding in its key search. Thus, implementing

case-agnostic key search would require either losing the capitalization of the original name and saving all sub names with the same capitalization pattern, or combining and reordering the results of 2^n prefix searches for every query where n is the length of the prefix. Note that not all searches would be over the whole search space, and LevelDB's key search is extremely fast, so the latter option may be feasible even though it initially seems absurd.

In the future, the home page can include a link to a more complex 'advanced search' option for users who are familiar with the site's language features, and who would like to enter the features they are looking for without browsing options iteratively, as the current page requires.

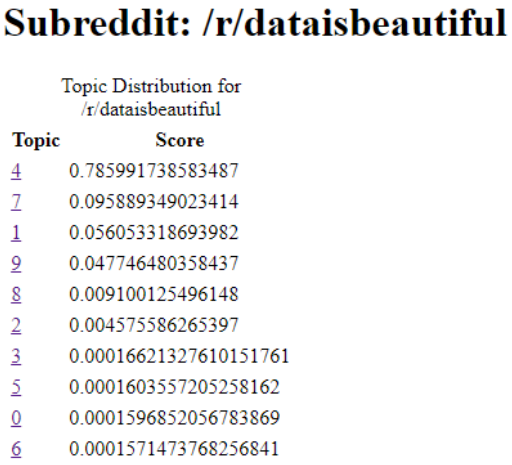
Other features I can include in the future would be:

- a list of interesting reddit suggestions for users with information needs that are not specific to a specific subreddit but more general information gathering about the types of reddits and language categories
- replace prefix search with a substring search so that 'narcissism' and 'raisedbynarcissists' can be returned by the same search
- a topic search feature in which the user provides a word or set of words and the site displays information about related topics based on the combined prevalence of words in the topics
- links to summary pages describing topics and subreddit distribution to give users a better understanding of the dataset and features.

3 Reddit Pages: /r/

Users can access reddit pages by an exact matching query from the home page, by clicking on a link in prefix search results, or in the example subreddits portion of a topic page. For each subreddit in the dataset, there exists a page which lists the distribution of topics in that subreddit in descending order with links to each topic description page.

These pages will be dramatically improved with the inclusion of more information. Ideally this page would show summary information about volume and length of posts, when the sub has been active, and what words contributed to its topic scores. If not the latter, topics could be represented in a more meaningful way than just by their numerical order.



In addition to providing more information, my initial plan for the site included the ability to select features from subreddit description pages to include in an advanced search feature. For example, if a user looking at the /r/dataisbeautiful page wanted to see pages which also had topic 4 as their most prevalent topic but had a relatively low score in topic 7, they would be able to click a button which would add those topics and positive or negative weights to a collection of search features that they would then be able to review and submit to receive a ranked list of subreddit matches. The goal of this feature would be to allow users not only to find pages which are similar to the each other, but also to learn something about the relationships between linguistic habits of various groups. (See also this article on reddit arithmetic.)

While I initially thought that just 'high' or 'low' scores for topics would be sufficient features to group similar subreddits by, the results in the example subreddits sections of the topic pages suggest this might not actually be effective. Some subreddits with very few posts have unusually high topic scores, which would throw off results. As such, subreddit similarity measures should be represented either by a complete vector of exact subreddit scores or a combination of binned topic scores (high-/medium/low) and the summary information

mentioned earlier in this section (post volume, average post length etc.).

4 Topic Pages: /t/

Topic: 9

This topic's most common words are:

Word	Score
amp	0.06557392980236
nbsp	0.027273266581114
code	0.0047369530502600005
like	0.004187490240037
codes	0.0035963995773690004
psn	0.003299346541788
pokemon	0.0032133570016090003
new	0.002904642947971
file	0.0026959690349450003
free	0.0025769393784690004

Subreddits which have the highest values fo this topic are are:

/r/

- ledootgeneration
- hailcabbage
- Fanfictionsofreddit
- jakirojakirojakiro
- Ryaisms
- awwwtf
- fairytailcirclejerk
- whereidthesodago
- AncientCivilizations
- futanari_Comics

Topic pages are accessible from links on subreddit pages. They show the top 10 most frequent words in the topic and a list of 10 subreddits which have a high concentration of that topic. Overall, the topics were not clearly descriptive of subreddit content. A few potential ways of addressing this are discussed below.

Topics used in this project were generated by training an LDA model on reddit submissions, after having removed Spark's 'Standard English Stopwords' from each post. The number of non-dictionary words is one of the issues which makes topics hard to interpret. There are many of these words in the most common words for the topics, including abbreviations, and in the case of topic 2 even a rather lengthy url. On the one hand, this is to be expected given the corpus, but on the other, a better tokenizer or further restricting the input data to a more limited set of posts and subreddits could have drastically improved results.

Topic: 2

This topic's most common words are:

Word	Score
gt	0.05297343772463801
c	0.033506293961553
lt	0.03171737484276
de	0.018954704390877
sticker	0.017807175646214
keys	0.015616201812056002
katowice	0.014206435072528
http://steamcommunity.com/market/listings/730/sticker	0.012470267525364
que	0.011753380397343001
k	0.011717508672464001

Subreddits which have the highest values fo this topic are are:

/r/

- TheShirt
- erk
- cheapcsgotrading
- PlazaCForalNavarra
- karmaparajaime
- GlobalOffensiveTrade
- h1z1market
- TomasVillegas
- BitcoinVzla
- MexicoCraftBeer

The resulting topics are also not very balanced, as topic 4 seems to be the most prevalent topic in a majority of subreddits. The balance could be improved by adjustment of the document concentration hyperparameter or addition of more stopwords.

The subreddit topic distributions used in this dataset were based on an LDA model that used posts as input documents, but the distributions themselves were based on subreddits as the sum of all words in their posts. This could also have had a negative effect on the analysis, as words could have much different tf-idf among post documents as opposed to subreddit documents, and perhaps the model should have been trained using subreddits as documents. However, I chose this architecture based on the nature of internet forums; posts are more likely to be from a single topic, and forums/subreddits should be a combination of topics because they are a combination of posts. Given the results, perhaps this conceptualization of the LDA process is out of line with the mathematical reality of the model. In the end, more iteration and parameter tuning is neces-

Topic: 4

This topic's most common words are:

Word	Score
like	0.009333334800165
get	0.007006275135742
one	0.006561737375237
time	0.005943152037368
know	0.005594165736021
really	0.004760405377542
want	0.004630715253509001
even	0.003946919130159
also	0.003889150607254
people	0.0038333512998360002

sary to determine the best model.

Given that the results of LDA were so distant from what I had hoped for, perhaps, instead of using topic features to describe subreddits, I could have used dictionary features. This would have avoided the awkward non-word problem that the topics had. I chose the topics because, theory-wise, that is closer to what I actually wanted to model: a subreddit as a combination of topics, as opposed to a collection of sentiment dictionaries.

In the end, there are a lot of ways to describe language. I chose the LDA model because it was both one I had worked with recently and so could implement more easily, and also because it is the model that corresponded best to what I expected to find in the reddit data. Potentially, however, a different language model might be more useful to retrieve the information that I was looking for. Perhaps future iterations of this site will include both more varied and more accurate language modeling, and provide more complex search features to enable the user to navigate the additional language features fluidly.