

Recommendation Systems

UNI : akv2001
ID : 9
TITLE: Degustibus
TYPE : SOFTWARE

Tired of eating the same old thing? Degustibus is a web-based "lunch engine" designed to give novel and relevant food suggestions from restaurants in the Columbia University area. Employing K-means clustering and relevance feedback, Degustibus rotates around clusters to maximize the novelty of suggestions and the diversity of results. Selections are used to refine browsing and discover similar offerings in the neighborhood.

UNI : xz2242
ID : 12
TITLE : Simple content-based movie recommender
Type: Software Project

In this project, a simple content-based movie recommendation system is implemented. The movie datasets are collected from IMDB database. The recommendation system leverages the Naive Bayes probability model. The model is learned using user inputs and movies are recommended to the user based on the classification score.

UNI : pl2412
ID : 22
TITLE : Movie Recommendation System
TYPE : SOFTWARE

Abstract: My project is developing a Movie Recommendation System by MovieLens dataset and IMDB dataset. The system has two parts, online part and offline part. The offline part using Naive Bayes method on IMDB dataset to calculate the rating of all users to all movies in the MovieLens, and add the top N movies with the highest score to user rating history to expand the user's rating item space. The online part using collaborative filtering method on MovieLens dataset, calculating the Pearson correlation between users and calculate the predicted rating based on that. For the online part, the response time is controlled in less than 10 seconds. For the offline part, it is less than 10 hours. For the evaluation part, I will calculate the mean absolute error between the prediction and actual rating. I expect the system could retrieve movies that the user really likes.

UNI : kl2549
ID : 28
TITLE : Artist Recommendation System
TYPE : SOFTWARE

The goal of my project is: given a singer/band/musician, and the system will return a list of singer/band/musician which people like the given one also like. My project is mainly built on twitter API and Yahoo BOSS API. Because there's an extension, I'll try to include Facebook API to make the query result more "relevant". <http://www.cs.columbia.edu/~kl2549/coms6998/project/index.php>
This is the web interface for my project. The instructions are all listed on it. I make the program to send http request to twitter every 5 seconds to prevent from being banned IP. Therefore it costs some time for a new query (200*5 secs, where 200 is the sample number). I also use youtube API to display the query results. UNI : cl2894

ID : 30

TITLE : Movie Recommendation System

TYPE : SOFTWARE

Used kMeans, Cosine Similarity, Association Rules as three separate algorithm to get the most possible unseen movies for a certain user. These three results will be combined by using Threshold Algorithm to find the Top k results.

I will use cross validation to test the experiment results.

I have finished all 4 algorithms, right now I am combining them. After the combination, I will divide the data to training and testing sets to do cross validation.

Search Engines and Crawlers / Spiders

UNI : yz2364
ID : 29
TITLE : Blog Retrieval
TYPE : SOFTWARE

The project name is blog retrieval. The data set used is part of the political blogosphere compiled by Lada Adamic and Natalie Glance, including about 400 blogs. And for the retrieval part, each blog is scored according to the sum of the score of the posts in it. And the score of each post in each blog is determined by idf and normalized by the length of it. Each post is proximity-based reranked. The result returned is the blogs related to the query and they are ranked with the score. Also, a summary of the most related post would be presented as well as the url of it.

UNI: dvg2107

ID : 11

TITLE: Patent Search Engine
TYPE : SOFTWARE

My project is an implementation of a search engine for patents. I made two main changes from a standard keyword based search. The first is to have the query be a short description of the invention being considered. The second, and more substantial, is to utilize the citations in the patents to produce clusters of similar patents. The engine implements this clustering in two ways. The first is to match the query against pre-indexed clusters of patents. The second is to produce clusters for patents that match the text of the query.

UNI: yw2390
ID : 4
TITLE: Image Search on Book Covers
TYPE : SOFTWARE

When we see a book which we are interested in, instead of typing the name of the book on search engine, this system requires the image of the book's cover, and gives the title, author, price and reviews of the book. The user interface is a web page for user to upload the images. This system demonstrate how to construct a image retrieval system simply by text retrieval techniques, which we learned from this class. Sift features will be extracted from images and K-Means will be used for clustering them together.

UNI : xz2244
ID : 7
TITLE : Cruise Crawler
TYPE : SOFTWARE

The CruiseCrawler crawls information about cruise ships, including technical statistics, comments, itineraries, real-time position, etc. The used websites includes Wikipedia, Avoya Travel, Cruise Critic, etc. Some social networks like Twitter may be used to extract comments. The crawled information will be stored in MongoDB. A website is built based on this database. It is used to show the merged information from different sources. Other tentative features include: show real-time ship positions on maps; automatically label positive/negative on comments.

UNI : yz2351
ID : 26
Title: Web Spidering and PageRanking

Project Type: Software Project

Abstract: Implement a Personalized-PageRank-Spider(PPR-Spider) Class that start crawling from a given website and download all HTML pages. Output a list of url in decreasing order of PageRank value. Block-Rank Algorithm is implemented, which is the personalized page rank algorithm in this PPR-Spider.

Evaluation method: To verify the correctness, choose a small sample website with 5-6 links, compare the result of program with results manually calculated.

To see the effect of Personalized step on the Page Ranking, we can compare the graph of standard PageRank with Personalized PageRank, which will be demonstrated in the report.

UNI : rb2838
ID : 1
TITLE : Relevance FeedBack for Search Queries
TYPE : SOFTWARE

Often while searching for something, we are not able to come up with a specific name to be able to correctly describe what we are looking for or we don't exactly know what it is we are looking for at all! In such cases it is desired that the Information Retrieval system be able to help you find the content you are looking for, even with a fuzzy input query. On seeing some of the documents returned by the IR system the user can rate them as relevant or not, and thereby guide the system to find the correct documents.

UNI : jwl2140

ID : 20

TITLE : Implement TextRank in ClairLib

TYPE : SOFTWARE

There are several established techniques for text summarization, and Clairlib already has the ability to summarize documents using lextank (for multiple document summarization), and by combining sentence scores. However, one of the major techniques for summarization developed at the University of North Texas known as TextRank is missing from Clairlib. To remedy this, I implemented TextRank in Clairlib. The dataset I used is the DUC dataset and I used ROUGE to evaluate the performance of the different Clairlib summarizers. The interface is similar to the one used in LexRank, but not identical because the values are computed differently.

Summarization and Classification

UNI: lx2141
ID : 15
TITLE : Multi-Doc News Summarization System
TYPE : SOFTWARE

Description: The System tries to summarize multiple documents into a condense paragraph by evaluating and utilizing several sentence features such as sentence position, length, Tf*idf and headline information. It is well-tuned during experiments that the weight for these feature to combine a comprehensive score. Furthermore, the system eliminates the repeated similar-meaning sentences in the summary.

*The program is still under development, the current result shows that it has decent result of summarized sentences, but the eliminating similaraty still need to be improved much.

UNI: zd2140
ID : 17
TITLE : Text Summarization and a new approach to identify thematic words
TYPE : SOFTWARE

In most automatic text summarization approaches, one major task is to identify sentences that convey important information in that original text. In this project we developed an automatic single document summarization system, which will base on the statistical result of the training corpus to develop a standard score mechanism, thus identify the possible important sentences. Partially, whether a sentence contains thematic words is a major indicator of whether this sentence is important. But the traditional approaches that extract thematic word solely base on their term frequency is problematic, here we propose a new way that we utilize the Google Trend or Yahoo! Boss News API to find thematic word that we argue not only have higher possibility to identify thematic word but also make the summarization up to date and more relate to people recent concern.

UNI : zg2145
ID : 25
TITLE : News Summarization Based on Maximum Coverage
TYPE : SOFTWARE

This project is an implementation of a software for news summarization. I will use DUC as the data set.
The results will be a set of most 'important' sentences extracted from the text. These sentences should cover as many different words that appear in the text as they can. Also, different words will have different weights according to their frequency and positions.
To evaluate the results, I will compare the results generated by the software with the given summary in the original data, and use cosine similarity as the evaluation metrics.

UNI: z12241

ID : 6
TITLE : Movie Classification

Project Type: Software.

This is a movie classification project that classify movies according to movie genres. Movies classification can help people find movies they're interested in. It would save us a lot work and time if we can classify movies automatically. The features will be movie actors, actress, synopsis, etc. I'm using IMDB dataset as the corpus. And since the IMDB

dataset is already classified by its genre, I can compare the result of classification to the original genre and get evaluate the accuracy of the program.

UNI : yz2373

ID : 19

TITLE : Email Classification System

TYPE : SOFTWARE

Even though automatic document categorization has been extensively investigated, email classification is a unique challenge because email folders do not necessarily correspond to a semantic topic and topic associated with a certain email folder may shift as well. This project will implement two methods to do automatic classification of email folders. One is Naive Bayes, which is a classic method in document classification. And another method is wide-margin Winnow algorithm which is similar to perceptron. I partition the dataset into training and testing according to date of emails and apply incremental timeline splits. Then, for each test case, I evaluate the effectiveness of a method in terms of accuracy. My experiments suggests that, Naive Bayes is very efficient while training models for different classes, but in some occasions, cannot achieve a satisfactory accuracy. On the contrary, Wide-Margin Winnow algorithm can outperform Naive Bayes for enron dataset, but its training is pretty slow.

UNI : ab2929

ID : 21

TITLE : Bwog.com blog post categorization (unsupervised learning)

TYPE : SOFTWARE

Problem statement:

KMeans and GAA clustering algorithms were compared in their attempts to to try categorize subsets of the posts in the corpus. This involved several substeps, including XML parsing, feature extraction, part-of-speech tagging, and word lemmatization using NLTK, BeautifulSoup, lxml, and other tools.

Clustering was only useful for posts for which there was a regular, consistent format among posts in the same category (e.g. morning news roundups, sports articles, movie recommendations). Much of the writing is humorous and relies on puns and imagery, leading to sparse or misleading text in many cases. GAA clustering gave better results than Kmeans.

Query Expansion

UNI : xg2158
ID : 23
TITLE : Query Expansion Methods
TYPE : PAPER

Various query expansion methods have been widely used in information retrieval to enhance the search performance. It's especially useful in Web retrieval systems because the queries submitted by users tend to be very short and query terms differ a lot from document terms used by the author. Traditional query expansion methods fall into two categories: global analysis such as term clustering, latent semantic indexing, similarity thesaur, ect. , and local analysis such as relevance feedback. Query logs have also been used in various ways to expand queries. Recently, query expansion methods using each user's personal information repository to personalize the search output has also been developed. This paper examines the different query expansion approaches including global analysis, local analysis, query log analysis and personalized query expansion. A comparison between these methods will be provided.

UNI : is2378
ID : 24
TITLE : Query Expansion : Improving IR Relevance and Performance
TYPE : Software

Query Expansion is the process of making user query less ambiguous and more informative, so as to increase the relevance of the documents returned by the search engine. It involves evaluating the input query and to either add words to it or to give user some options of what can be added to the query to match more relevant documents. Query expansion can be achieved in many ways, this system uses relevance feedback and pseudo relevance feedback for expansion.

For implementation of this system results returned by YAHOO! BOSS (Yahoo!'s open search and data services platform) is used. The returned documents are then analysed for important words which can be added to the seed query to make it less ambiguous and more focussed. As an additional feature, the system also gives option for query reformulation using part of AOL query logs which is available on web.

UNI : aev2114
ID : 27
Title: Multiple Method Query Expansion and Automatic Thesaurus Generation
TYPE : SOFTWARE

My project is a query expansion system capable of performing expansion using an automatically constructed thesaurus (based on syntactic context), a hand-crafted thesaurus (WordNet), and by using the Rocchio Algorithm with pseudo relevance feedback. Both a web interface and command line utilities have been created for interacting with the system. Querying, viewing expanded queries, looking up related words, and evaluation can easily be performed from within a browser. Query-able data sets include Cranfield, Medline, and a subset of Wikipedia. Evaluation is performed on the Cranfield and Medline data sets using standard test queries and trec_eval methods. Results thus far have been promising with some methods showing significant increases in effectiveness.

UNI : rf2415
ID : 10
TITLE : Image Query Expansion
TYPE : SOFTWARE

My project is a web application software. I implemented the "IMAGLE" system, which is an images retrieval system with relevance feedback and automatic query expansion. I use Yahoo Boss API to help me get basic data from the Internet. The IMAGLE firstly shows image results base on user's original query and settings through web browser. If user does not satisfy the result, after user marking the relevant, it can returns new results based on automatically expanded query, which derived by information of all relevant source web pages. Specifically, I used and also modified Rocchio Algorithm to do the query expansion. Evaluation of this system is also based on use's target precision.

UNI : ww2267
ID : 2
TITLE : Question Answering on Social Search
TYPE : PAPER

Compared to traditional web search, which analyzes the text of documents and the relationship between documents, the social search takes into account the social graph of the person initiating the search query. Thus, the result from Social Search Engine is generated from the "friends" in the initializer's social network.

Apart from the difference on forms of dataset, the usability of social search is different from traditional web search. In contrast to web search engine, from which users ask for the facets or navigational information, social search lets users to query experiential information from friends of social network who he/she trusts, e.g. a person who can query with "Can anyone give me suggestion on finding a baby-sitter who's task is not to allow the child watching TV at bay area?", whose answers cannot be found from traditional web search engine.

In this survey paper, we will cover: 1. Introduction to state of art of social search. 2. Analysis of several models of question answering system based on social search. 3. Analysis of methods applied in a typical question answering system.

Miscellaneous

UNI : dlb2155

ID : 3

TITLE: Real Time Indexing

TYPE : SOFTWARE

About:

This is a software project where I deal with indexing information in real time. Once the query engine is started, any changes to the data it indexes should not require re-indexing and restarting query engine. This has significant advantages because indexing is time consuming. Even a small data set like Cranfield docs takes around 100 seconds on CLIC lab machines. Dynamic indices will be created instead of re-indexing which will be merged later to the main index. A GUI will be implemented.

Implementation is still going on. I have written a module to remove a document from an existing index and build a dynamic index. It can be clearly seen removing a document from an index and creating another index for the document is much faster than re-indexing.

UNI: wc2372

ID : 8

TITLE : Geography QA System

TYPE : SOFTWARE

I've broken the CIA Factbook into its essential components and extracted the core of its data. I've also been able to build a basic question classifier module that distinguishes between two types of questions, general factoid questions and ranking questions (ex. "Which country has the fastest population growth rate?"). I've also built an information retrieval module that takes this determination and the components of a question and attempts to zero in on the snippet of text that would most likely contain the answer. It uses the $wf * idf$ formula to rank documents on how likely they are to have the answer. With ranking questions, it tries to return the name of the country that is ranked at the level the user asks for and in the category that the user wants. The user is required to phrase this type of question in a specific way that isn't completely natural, but is still a fair approximation to natural language. With factoid questions, it looks at the most likely country description document and tries to return the text that hopefully would answer the user's question. At this point, the answers returned are whole sections of text. I will continue to try to narrow down these snippets further, by using some more complex NLP algorithms. I will also try to increase of number of question types that the system can recognize in the hopes that this will lead to more accurate answers.

UNI: r12553

ID : 13

TITLE: A Stock Return Prediction System

TYPE : SOFTWARE

Evaluation: Used the following metrics:

1. Compare the return of the system to that of S&P 500 index
 2. Compare the return of the system to that of the mutual fund in specific industries
 3. Compare the return of the system to the actual average returns of the companies predicted
 4. Show the predicting trend accuracy
-

UNI : yy2342
ID : 14
Title: Spam Recognition
TYPE: SOFTWARE

There are many methods to do spam recognition, and also many commercial or open-source anti-spam systems. In this system, we mainly focus on the performance of different Bayes classifiers. We use TREC spam track's dataset, and 5 different Bayes classifiers, evaluate the performance using SPAM recall and HAM(non-spam) recall, and obtain the learning curve (recall as larger datasets) of the system. Finally, the system shows that Bayes classifier could achieve remarkably good result on spam recognition.

UNI : rs3078
ID : 18
TITLE : How Do They Feel About It: Opinion Mining The Blogosphere
TYPE : SOFTWARE

It is a natural tendency of people to look for reviews of a product on the Internet before deciding whether to buy it. Opinions expressed by people on their blogs help to define opinion trends that could be used in multiple ways. The goal of this project is to build an information retrieval system that retrieves blogs relevant to and opinionated about a user's query. The software uses an opinion lexicon and is based upon a proximity based probabilistic model for assigning subjectivity scores to documents. The search results include snippets from blogs that best capture the sentiments of the bloggers. I am using a corpus of blog posts on American politics during 2007-2008. I am using Mean Average Precision to evaluate the performance of the retrieval system.

UNI : pg2354
ID : 5
TITLE : Web Graph Analysis
TYPE : SOFTWARE

The main motivation for the Web Crawler is that all the search engines are based upon it. The web crawler helps determining the number of hits and PageRank. There are many other uses as well like specialized search engines (e.g. news, shopping, reviews, etc.), business intelligence (keep track of potential competitors, partners), monitor websites of interest and wrong use as well like harvesting emails for spamming or phishing.

The goal of this project is to implement a Web Crawler addressing the all following implementation issues:

- * Spider Traps - infinite number of pages dynamically generated
 - * Parsing - pages may not be strictly syntactic
 - * Fetching - broken links, redirections
 - * URL Canonicalization - all URLs must be in canonical form
 - * Relative URLs - must be converted to Absolute URLs
 - * Crawler etiquette # - must be ethical and should not crawl restricted pages (robot exclusion)
-

UNI : lmv29
ID : 16
TITLE : Stocks & stilettos: The effects of brand-specific online text streams on abnormal prices.
TYPE : PAPER

This paper summarizes past contributions to social text stream analysis and examines potential applications of pricing models that make use of event-driven trends observed an online forum dedicated to one luxury shoe brand. I chose to examine the effects of a brand-specific online forum on an easily observable and highly unregulated market: eBay. Initial analysis of approximately 6 months of daily data from eBay & the online forum showed several short-term trends based on volume & content of posts in the forum. Several abnormal end prices are observed following events on the forum.