MP3: CS 412 Introduction to Data Mining

Frequent Pattern Mining Programming

Sittichok Thanomkulrat (thanomk2)

October, 24 2016

Introduction

In this assignment, we performed the frequent pattern mining on the titles of conference paper from five different domains namely, Data mining(DM), Information retrieval(IR), Machine learning(ML), Database(DB) and Theory(TH). The process began by preprocessing the data for the LDA and then extracting the result into meaningful phrases according to each different domain. We decided to use the Apriori algorithm since it was simple to implement and the dataset size was not enormous. In addition, python3 was a main language for this assignment due to its readability and maintainability.

The Apriori algorithm (Step 4)

To extract the pattern, Apriori algorithm was implemented based on the pseudo code in the lecture 6 slide. The general idea was as follow.

1) Import the data from "title-i.txt" to be used as the main dataset.
2) Find all the frequent pattern of length 1 and their support value. This is the candidate of frequent pattern of length 1 list.
3) Prune the pattern from candidate list unless its support value is at least equal to threshold. This is the frequent pattern of length 1 list.
4) Then, create a candidate frequent pattern of length 2 list from frequent pattern of length 1 list
5) Like step 3, prune the pattern again from candidate list of length 2. The result is the frequent pattern of length 2 list.
6) Keep doing step 4 and 5 until the next candidate list cannot be generated or the frequent pattern list is empty.
7) Print all patterns in the frequent pattern list of each length and save pattern to file

***"Question to ponder A**: How you choose min_sup for this task? Note that we prefer min_sup to be the consistent percentage (e.g. 0.05 / 5%) w.r.t. number of lines in topic files to cope with various-length topic files."*

The minimum support is an important parameter for Apriori algorithm since it is the threshold to retain or reject the pattern. Usually, the higher the value the more extracted pattern will be returned and vice versa. Indeed, there is no general rule to determine this value since it is based on the dataset and target of the mining.

For this assignment, the dataset is not huge and our goal is to extract the pattern that helps determine the topic domain. Hence, the support value must be low enough to retain longer phrase and high enough to not generate unrelated noise that will unnecessarily increase workload. After trial and error, the appropriate value of relative support is approximately between 0.0015 – 0.0005 since in this range the frequent pattern of length 3-4 will be retained and those pattern

will be extremely useful in determining topic domain, especially when combining the ranking method in step 7 that will be discussed later.

Max-pattern mining (Step 5)

For max-pattern mining, we decided to mine the pattern that was generated from step 4 by using the definition of max-pattern as a guideline since the amount patterns that were generated from step 4 was not tremendous. The algorithm was as follow.

1) Import patterns that were generated from step 4 to be used as dataset
2) For each pattern in each topic, search through the pattern list for a super-pattern
3) If the super-pattern is found, discard current pattern and move to next pattern
4) Else, this is the max pattern; store it and move to next pattern
5) Keep doing until the end of list is reached
6) Save all the pattern to file

Closed-pattern Mining (Step 5)

For closed-pattern mining, we utilized the same methodology in max pattern mining. The general idea was as follow.

1) Import patterns that were generated from step 4 to be used as dataset
2) For each pattern in each topic, search through the pattern list for a super-pattern
3) If the super-pattern is found and its support value equals to that of current pattern, discard current pattern and move to next pattern
4) Else, this is the closed-pattern; store it and move to next pattern
5) Keep doing until the end of list is reached
6) Save all the pattern to file

***"Question to ponder B:** Can you figure out which topic corresponds to which domain based on patterns you mine? Write your observations in the report."*

Based on the information above, we can only determine that topic-2 could be Database domain since the mined patterns are closely related to the terms that appear a lot in Database paper. For other topics, we cannot reach a conclusive decision yet.

Purity Mining (Step 6)

Based on the paper, purity was defined as the ratio between probability of seeing pattern in title-t document and that of same pattern in title-t and title-t` (t ≠ t`) combined document.

$$purity = \log\left(\frac{prob(pattern)\ in\ document_t}{\max t`\ prob(pattern)\ in\ document_{t+t`}}\right)$$

Hence, we implemented this constraint as follow.

1) Import patterns that were generated from step 4 to be used as dataset
2) Generate a set of each topic patterns
3) For each pattern in topic t, calculate the logarithm of probability of seeing this pattern in topic t document
4) Calculate the logarithm of probability of seeing this pattern in topic t`+ t document for every t` and find the maximum value among them
5) Calculate the purity based on the formula in paper and assign it to current pattern
6) Keep doing these step for all patterns of every topics
7) Sort all pattern based on purity value and output them into file

Combined ranking function (Step 7)

Since the extracted patterns from step 4-6 were not satisfied enough to clearly distinguished the topics, we decided to implement extra filtering measures based on the paper to enhance the quality of mined patterns. Hence, Coverage, Phraseness and Completeness were implemented based on the formula on the paper. The implementation idea was as follow.

Coverage

Coverage was defined as the probability of seeing this pattern in the document of relevant topic document. Hence, it can be directly calculated in each loop.

$$coverage = \frac{support(pattern)}{|document|}$$

Phraseness

Phraseness was defined as the ratio between probability of seeing this pattern in the document of relevant topic and that of each words in pattern independently.

$$phraseness = log\left(\frac{prob(pattern)\ in\ document_t}{\Pi\ prob(w_i)\ in\ document_t}\right),\ \ \forall w_i \in pattern$$

Completeness

Completeness was defined as 1 minus ratio between probability of seeing super-pattern that had the maximum value and of that of current pattern in topic-t document. Hence, for every pattern, we had to find whether its super-pattern existed. If they did exist, we had to calculate the probability of them all and find the maximum value among them. Then completeness value can be calculated from the following formula.

$$completeness = 1 - \frac{\max_{\text{super\_pattern}} \text{prob(super\_pattern)}}{prob(pattern)}$$

After we calculated all of the filtering, the combined ranking formula, based on the paper, was as follow.

$$rank = 0, completeness \leq threshold_{com}$$

$$rank = coverage\big((1 - weight)purity + (weight)phraseness\big), completeness > threshold_{com}$$

Result analysis

Here are top 5 sample patterns that were generated from step 4.

Topic-0	Topic-1	Topic-2	Topic-3	Topic-4
2452 system	1345 mining	3278 data	2517 based	2158 using
1168 approach	1337 information	2440 database	2046 learning	1800 model
903 knowledge	1114 retrieval	1973 query	1658 algorithm	887 analysis
757 language	865 tree	637 processing	1226 web	489 semantic
527 distributed	855 time	589 relational	1110 search	488 feature

It was not clear to distinguish the five domains from the generated patterns in this step since all of them were single word and word cohesion was low. However, we could guess that Topic-2 should be Database domain since the entire sample keywords were related to database. This prediction would be confirmed in the later step below.

Here are top 5 sample max-patterns that were generated from step 5.

Topic-0	Topic-1	Topic-2	Topic-3	Topic-4
35 classification approach	104 robust	30 database technology	39 method efficient	21 using structure
34 system interactive	91 retrieval image	28 data driven	33 learning online	21 model computational
31 logic programming inductive	90 computation	27 data managing	31 web search engine	19 using linear
26 automatic system	86 local	27 data incomplete	29 algorithm optimal	19 using cluster
25 system monitoring	78 heuristic	26 data transaction	29 efficient document	19 using modeling

It was clear that the result pattern for each topic was not relevant enough to distinguish topic domain from the extracted pattern. Again, except topic-2 that it was clearly Database domain. This might stem from the fact that most of paper title in the database domain that consisted of small set of keywords, resulting in high support value of related word.

For closed-pattern, this was the top-5 patterns.

Topic-0	Topic-1	Topic-2	Topic-3	Topic-4
2452 system	1345 mining	3278 data	2517 based	2158 using
1168 approach	1337 information	2440 database	2046 learning	1800 model
903 knowledge	1114 retrieval	1973 query	1658 algorithm	887 analysis
757 language	865 tree	637 processing	1226 web	489 semantic
527 distributed	855 time	589 relational	1110 search	488 feature

It is obvious that the sample closed-pattern was identical to that of steps 4 pattern since the close pattern retained all patterns that had no super-pattern with same support value. Also, it was noteworthy that the size of generated pattern > closed pattern > max-pattern as expected.

Here are sample patterns from purity measure from step 6.

Topic-0	Topic-1	Topic-2	Topic-3	Topic-4
2454.133554826074 086 system	1346.268018832293 0326 mining	3279.744160116989 0557 data	2518.208040207435 2024 based	2159.248986938332 549 using
1169.517136199093 7693 approach	1338.264758848592 9508 information	2441.465754801374 4373 database	2047.083106520432 5498 learning	1801.142185029024 6733 model
904.3899158438369 283 knowledge	1115.173886802953 1793 retrieval	1974.310605299713 3082 query	1658.980188748549 2606 algorithm	887.8698103273540 285 analysis
758.3198246292425 926 language	866.0724198102881 426 tree	637.8667514791153 008 processing	1226.865599889132 8147 web	489.7510752438218 086 semantic
507.1993253493578 12 logic	856.0683448306630 405 time	589.8508046352614 806 relational	1110.834830658363 5844 search	488.7507769144662 009 feature

Since the purity measure was biased towards the word that was unique to specific topic, we could notice that the top-ranking patterns were all single word. As a result, purity alone was not helpful enough for our objective. Hence, we decided to implement all measures that were introduced in the paper.

***"Question to ponder C:** Compare the result of frequent patterns, maximal patterns and closed patterns, is the result satisfying? Write down your analysis."*

The extracted patterns are partially satisfied for our goal of determining topic domain since they can help in determining topic-2. Other than that, the mined patterns are not helpful since most of them are general keyword that could appear in many computer science related paper such as data, language, system. Hence, we need more sophisticated filtering measure instead of support value for this task.

Here are sample patterns from combined ranking measure from step 7.

Topic-0	Topic-1	Topic-2	Topic-3	Topic-4
21.11572784297605 knowledge system approach	15.5599461010962 decision mining	26.598095972456388 database query xml	19.709628206706938 learning based algorithm	30.733369538195802 using model analysis
16.46634270941512 classification language	13.850420793870212 rule pattern	23.482444537899124 database object data	16.805008005732407 web clustering search	19.889661521636967 semantic model analysis
20.11052227156417 approach program	12.259937756571453 mining optimization	21.69181349359773 database stream	14.75214927331026 learning algorithm machine	19.051715886999407 model analysis selection
19.68741947659705 system base approach	12.154745930411092 structure information retrieval	21.64844512589802 query data relational	10.598469668633426 method learning machine	15.29268497284076 model analysis probabilistic
18.723586915703002 language distributed	7.21541877666514 mining classifier	18.853329045093826 database xml data relational	8.505861603863913 learning algorithm supervised	11.81706333036124 classification semantic using

This combined ranking measure confirm our assumption that topic 2 is indeed from Database domain and we can guess with more confident that topic 0, 1, 3 and 4 are Information retrieval, Data mining, Machine learning and Theory respectively.

Source Code

The source code for this assignment was as follow.

main.py – the main python file will call helper file in each step and perform the combined ranking measure.

step2.py – this file will import “paper.txt” from current directory and generate “vocab.txt” and “title-i.txt”.

step3.py – this file will separate the result tokens from lda into appropriate topic files.

step4.py – this file will perform the Apriori frequent pattern mining with specified relative support value.

step5.py – this file will generate max and closed pattern based on the results from step4.

step6.py – this file will rank the frequent pattern based on purity measure.