DATA ANALYTICS

Unit 4: Rule Generation (Apriori Algorithm) + Evaluation of Recommender Systems

Gowri Srinivasa
Department of Computer Science and Engineering

DATA ANALYTICS

Rule Generation

How to efficiently generate rules from frequent itemsets?

In general, confidence does not have an anti-monotone property

$c(ABC \rightarrow D)$ can be larger or smaller than $c(AB \rightarrow D)$

But confidence of rules generated from the same itemset has an anti-monotone property e.g., L = {A,B,C,D}:

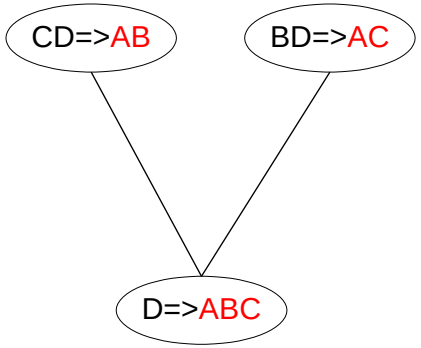
$$c(ABC \rightarrow D) \geq c(AB \rightarrow CD) \geq c(A \rightarrow BCD)$$

Confidence is anti-monotone w.r.t. number of items on the RHS of the rule

Candidate rule is generated by merging two rules that share the same prefix in the rule consequent

join(CD=>AB,BD=>AC)
would produce the candidate rule D => ABC

Prune rule D=>ABC if its
subset AD=>BC does not have high confidence



Slide courtesy of Tan,Steinbach, Kumar, Introduction to Data Mining

Rule Generation

Given a frequent itemset L, find all non-empty subsets f $\subset$ L such that f $\rightarrow$ L – f satisfies the minimum confidence requirement

If {A,B,C,D} is a frequent itemset, candidate rules:

ABC $\rightarrow$D,	ABD $\rightarrow$C,	ACD $\rightarrow$B,	BCD $\rightarrow$A,
A $\rightarrow$BCD,	B $\rightarrow$ACD,	C $\rightarrow$ABD,	D $\rightarrow$ABC
AB $\rightarrow$CD,	AC $\rightarrow$ BD,	AD $\rightarrow$ BC,	BC $\rightarrow$AD,
BD $\rightarrow$AC,	CD $\rightarrow$AB		

If |L| = k, then there are $2^k - 2$ candidate association rules (ignoring L $\rightarrow \varnothing$ and $\varnothing \rightarrow$ L)

Support and Confidence

$$support(A \Rightarrow B) = P(A \cup B)$$

$$confidence(A \Rightarrow B) = P(B|A).$$

$$confidence(A \Rightarrow B) = P(B|A) = \frac{support(A \cup B)}{support(A)} = \frac{support\_count(A \cup B)}{support\_count(A)}.$$

Computing Confidence

- In confidence of rule equation A => B can be easily derived from the support counts of A and A∪B.
- That is, once the support counts of A, B, and A ∪B are found, it is straightforward to derive the corresponding association rules A =>B and B =>A and check whether they are strong.
- Thus, the problem of mining association rules can be reduced to that of mining frequent itemsets.

DATA ANALYTICS

Evaluation of an association rule

Contingency table for $X \rightarrow Y$

	Y	Y	
X	f_{11}	f_{10}	f_{1+}
X	f_{01}	f_{00}	f_{o+}
	f_{+1}	f_{+0}	|T|

f_{11}: support of X and Y
f_{10}: support of $\overline{X}$ and $\overline{Y}$
f_{01}: support of $\overline{X}$ and $\overline{Y}$
f_{00}: support of X and Y

$$Lift = \frac{P(Y \mid X)}{P(Y)}$$

$$Interest = \frac{P(X,Y)}{P(X)P(Y)}$$

$$PS = P(X,Y) - P(X)P(Y)$$

$$\phi - coefficient = \frac{P(X,Y) - P(X)P(Y)}{\sqrt{P(X)[1-P(X)]P(Y)[1-P(Y)]}}$$

Slide courtesy of Tan,Steinbach, Kumar, Introduction to Data Mining

Limitation of Confidence

	Coffee	Not Coffee	
Tea	15	5	20
Not Tea	75	5	80
	90	10	100

Association Rule: Tea → Coffee

Confidence= P(Coffee|Tea) = 0.75 (75% of those who drink tea also drink coffee)

but P(Coffee) = 0.9 (90% of the people in our sample drink coffee (most of them do!))

⇒ Although confidence is high, rule is misleading

⇒ P(Coffee|NotTea) = 0.9375 (more interesting/ meaningful that nearly 94% of those who do not drink tea, drink coffee)

⇒ One is more likely to drink coffee if they do not drink tea (than if they do drink tea)

DATA ANALYTICS

Continuous and Categorical Attributes

How to apply association analysis formulation to non-asymmetric binary variables?

Session Id	Country	Session Length (sec)	Number of Web Pages viewed	Gender	Browser Type	Buy
1	USA	982	8	Male	IE	No
2	China	811	10	Female	Netscape	No
3	USA	2125	45	Female	Mozilla	Yes
4	Germany	596	4	Male	IE	Yes
5	Australia	123	9	Male	Mozilla	No
…	…	…	…	…	…	…

Example of Association Rule:

$\{\text{Number of Pages} \in [5,10) \wedge (\text{Browser=Mozilla})\} \rightarrow \{\text{Buy = No}\}$

Transform categorical attribute into asymmetric binary variables

Introduce a new “item” for each distinct attribute-value pair

Example: replace Browser Type attribute with

Browser Type = Internet Explorer

Browser Type = Mozilla

Browser Type = Mozilla

Handling of Categorical Attributes

Potential Issues

What if an attribute has many possible values?

Example: attribute country has more than 200 possible values

Many of the attribute values may have very low support

Potential solution: Aggregate the low-support attribute values

What if distribution of attribute values is highly skewed?

Example: 95% of the visitors have Buy = No

Most of the items will be associated with (Buy=No) item

Potential solution: drop the highly frequent items

Multiple minimum support also comes in handy in both cases

DATA ANALYTICS

Handling of Continuous Attributes

Different kinds of rules:

Age$\in$[21,35) $\wedge$ Salary$\in$[70k,120k) $\rightarrow$ Buy

Salary$\in$[70k,120k) $\wedge$ Buy $\rightarrow$ Age: μ=28, σ=4

Different methods:

Discretization-based

Statistics-based (mean, median, standard deviation, etc.)

Non-discretization based minApriori (concept hierarchy)

Discretization-based

Unsupervised:

Equal-width binning

Equal-depth binning

Clustering

Supervised:

Attribute values, v

Class	V_1	V_2	V_3	V_4	V_5	V_6	V_7	V_8	V_9
Anomalous	0	0	20	10	20	0	0	0	0
Normal	150	100	0	0	0	100	100	150	100

bin_1 bin_2 bin_3

Evaluation – objective measures

#	Measure	Formula
1	ϕ-coefficient	$\frac{P(A,B)-P(A)P(B)}{\sqrt{P(A)P(B)(1-P(A))(1-P(B))}}$
2	Goodman-Kruskal's (λ)	$\frac{\sum_j \max_k P(A_j,B_k)+\sum_k \max_j P(A_j,B_k)-\max_j P(A_j)-\max_k P(B_k)}{2-\max_j P(A_j)-\max_k P(B_k)}$
3	Odds ratio (α)	$\frac{P(A,B)P(\overline{A},\overline{B})}{P(A,\overline{B})P(\overline{A},B)}$
4	Yule's Q	$\frac{P(A,B)P(\overline{AB})-P(A,\overline{B})P(\overline{A},B)}{P(A,B)P(\overline{AB})+P(A,\overline{B})P(\overline{A},B)} = \frac{\alpha-1}{\alpha+1}$
5	Yule's Y	$\frac{\sqrt{P(A,B)P(\overline{AB})}-\sqrt{P(A,\overline{B})P(\overline{A},B)}}{\sqrt{P(A,B)P(\overline{AB})}+\sqrt{P(A,\overline{B})P(\overline{A},B)}} = \frac{\sqrt{\alpha}-1}{\sqrt{\alpha}+1}$
6	Kappa (κ)	$\frac{P(A,B)+P(\overline{A},\overline{B})-P(A)P(B)-P(\overline{A})P(\overline{B})}{1-P(A)P(B)-P(\overline{A})P(\overline{B})}$
7	Mutual Information (M)	$\frac{\sum_i \sum_j P(A_i,B_j)\log \frac{P(A_i,B_j)}{P(A_i)P(B_j)}}{\min(-\sum_i P(A_i)\log P(A_i), -\sum_j P(B_j)\log P(B_j))}$
8	J-Measure (J)	$\max\left(P(A,B)\log(\frac{P(B\|A)}{P(B)}) + P(A\overline{B})\log(\frac{P(\overline{B}\|A)}{P(\overline{B})}),\right.$ $\left. P(A,B)\log(\frac{P(A\|B)}{P(A)}) + P(\overline{A}B)\log(\frac{P(\overline{A}\|B)}{P(\overline{A})})\right)$
9	Gini index (G)	$\max\Big(P(A)[P(B\|A)^2+P(\overline{B}\|A)^2]+P(\overline{A})[P(B\|\overline{A})^2+P(\overline{B}\|\overline{A})^2]$ $-P(B)^2-P(\overline{B})^2,$ $P(B)[P(A\|B)^2+P(\overline{A}\|B)^2]+P(\overline{B})[P(A\|\overline{B})^2+P(\overline{A}\|\overline{B})^2]$ $-P(A)^2-P(\overline{A})^2\Big)$
10	Support (s)	$P(A,B)$
11	Confidence (c)	$\max(P(B\|A), P(A\|B))$
12	Laplace (L)	$\max\left(\frac{NP(A,B)+1}{NP(A)+2}, \frac{NP(A,B)+1}{NP(B)+2}\right)$
13	Conviction (V)	$\max\left(\frac{P(A)P(\overline{B})}{P(A\overline{B})}, \frac{P(B)P(\overline{A})}{P(B\overline{A})}\right)$
14	Interest (I)	$\frac{P(A,B)}{P(A)P(B)}$
15	cosine (IS)	$\frac{P(A,B)}{\sqrt{P(A)P(B)}}$
16	Piatetsky-Shapiro's (PS)	$P(A,B)-P(A)P(B)$
17	Certainty factor (F)	$\max\left(\frac{P(B\|A)-P(B)}{1-P(B)}, \frac{P(A\|B)-P(A)}{1-P(A)}\right)$
18	Added Value (AV)	$\max(P(B\|A)-P(B), P(A\|B)-P(A))$
19	Collective strength (S)	$\frac{P(A,B)+P(\overline{A}\overline{B})}{P(A)P(B)+P(\overline{A})P(\overline{B})} \times \frac{1-P(A)P(B)-P(\overline{A})P(\overline{B})}{1-P(A,B)-P(\overline{A}\overline{B})}$
20	Jaccard (ζ)	$\frac{P(A,B)}{P(A)+P(B)-P(A,B)}$
21	Klosgen (K)	$\sqrt{P(A,B)}\max(P(B\|A)-P(B), P(A\|B)-P(A))$

It is sufficient if we understand the idea behind the measures and are able to use some of these, such as, support, confidence, lift (or interest), phi-coefficient to evaluate a confidence rule or test for independence of (or correlation) between itemsets

Slide courtesy of Tan,Steinbach, Kumar, Introduction to Data Mining

Evaluation – subjective measures

Objective measure:

Rank patterns based on statistics computed from data e.g., 21 measures of association (support, confidence, Laplace, Gini, mutual information, Jaccard, etc).

Subjective measure:

Rank patterns according to user's interpretation

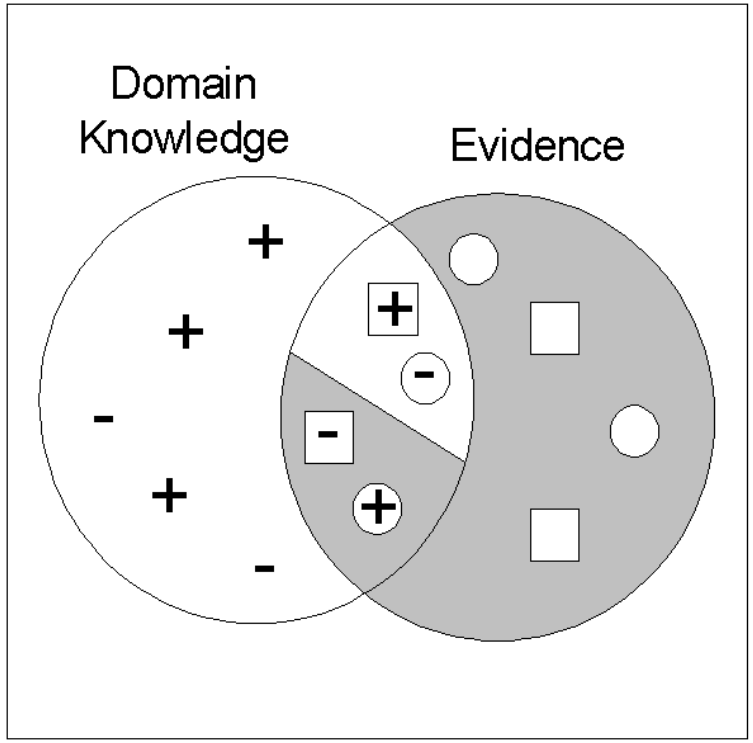
A pattern is subjectively interesting if it contradicts the expectation of a user (Silberschatz & Tuzhilin)

A pattern is subjectively interesting if it is actionable (Silberschatz & Tuzhilin)

Slide courtesy of Tan,Steinbach, Kumar, Introduction to Data Mining

Interestingness via unexpectedness

Need to model expectation of users (domain knowledge)



+ Pattern expected to be frequent

- Pattern expected to be infrequent

Pattern found to be frequent

Pattern found to be infrequent

Expected Patterns

Unexpected Patterns

Need to combine expectation of users with evidence from data (i.e., extracted patterns)

Slide courtesy of Tan,Steinbach, Kumar, Introduction to Data Mining

DATA ANALYTICS

Evaluation of Recommender Systems

Paradigms

1. User Studies: test subjects are actively recruited, and they are asked to interact with the recommender system to perform specific tasks (likes and dislikes inferred). Feedback can be collected from the user before and after the interaction.
   Results from user evaluations cannot be fully trusted.

2. Online Evaluation: A/B Testing (coming up in Unit 5)

3. Offline Evaluation: With historical datasets

Evaluation of Recommender Systems

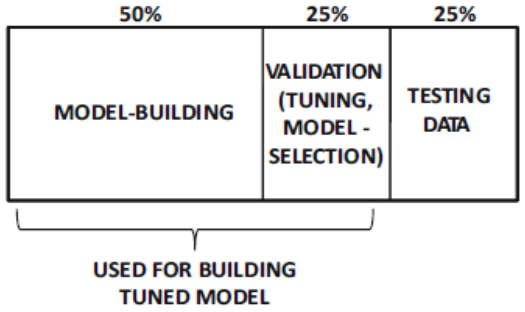
Measures

1. Accuracy: MSE, RMSE, etc.
2. Coverage: The fraction of users for which at least *k* ratings may be predicted (user-space coverage); the fraction of items for which the ratings of at least *k* users can be predicted (item-space coverage); the fraction of items that are recommended to at least one user (catalog coverage)
3. Confidence and Trust: confidence measures the system's faith in the recommendation, trust measures the user's faith in the evaluation
4. Novelty: Likelihood of a system to recommend an item the user was not aware of (A *differential* accuracy between future and past predictions can be used to quantify this.)
5. Serendipity – 'lucky discovery' or surprise factor (Ethiopian restaurant recommended to someone who likes Indian food is serendipitous; all that is novel is not serendipitous!)
6. Diversity: If three movies are recommended, they must not all be of the same genre; the changes a user will select one of them will then be higher (measured using content-centric similarity between pairs of items)
7. Robustness and stability: not significantly affected in the presence of attacks such as fake ratings or when the patterns in the data evolve significantly over time
8. Scalability: perform effectively and efficiently in the presence of large amounts of data (quantified based on training time, prediction time and memory requirements)

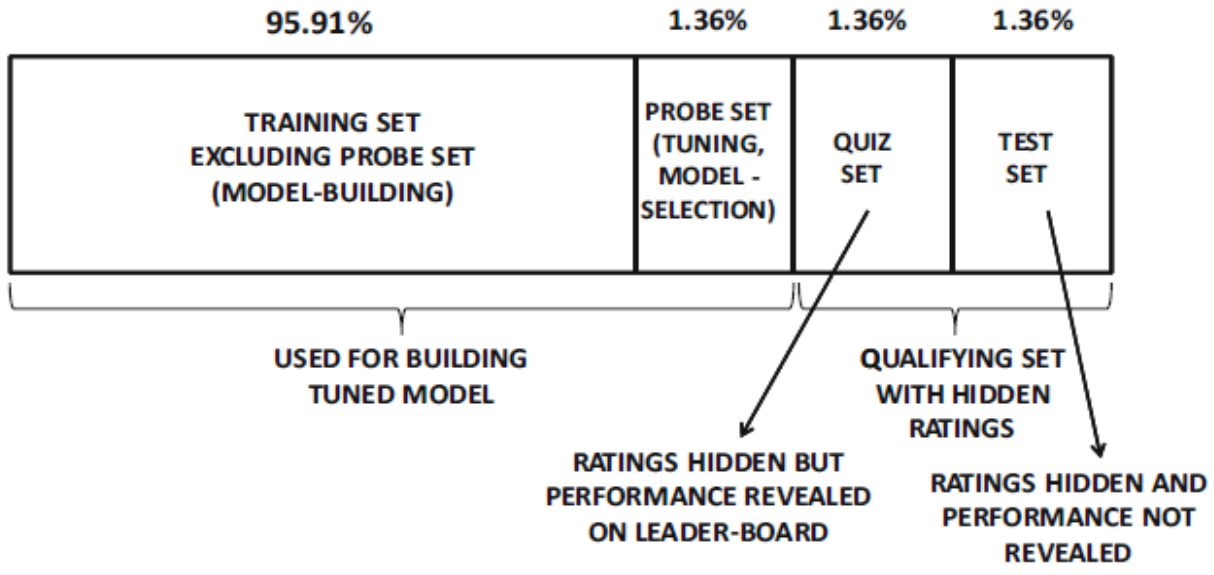
Case Study: The Netflix Prize - 1

The competition began on October 2, 2006: Prize awarded to the best collaborative filtering algorithm to predict user ratings based on training data <user, movie, date of grade, grade>

Usual partitioning of data in problems



Partitioning of Data for the Netflix Prize Challenge to penalize overfitting



Case Study: The Netflix Prize - 2

On September 21, 2009 "BellKor's Pragmatic Chaos" won the $1 million for their algorithm

This team used an ensemble of models; specifically, they used Gradient Boosted Decision Trees to combine 500 models! (Previous solutions had used linear regression for the combination).

Briefly, gradient boosted decision trees work by sequentially fitting a series of decision trees to the data; each tree is asked to predict the error made by the previous trees, and is often trained on slightly perturbed versions of the data.

Since GBDTs have a built-in ability to apply different methods to different slices of the data, we can add in some predictors:

- Number of movies each user rated
- Number of users that rated each movie
- Factor vectors of users and movies
- Hidden units of a restricted Boltzmann Machine

that help the trees make useful clusterings (For example, one thing that Bell and Koren found (when using an earlier ensemble method) was that RBMs are more useful when the movie or the user has a low number of ratings, and that matrix factorization methods are more useful when the movie or user has a high number of ratings.)

For the interested student: read more about this here (a summary) and here (links to the top papers). Crowd sourcing problem solving led to founding Kaggle and other similar organizations!

A few other application domains

1. *Query recommendation:* How can web logs can be used to recommend queries to users? Typically *session-specific* (i.e., dependent on the history of user behavior in a short session) and do not use *long-term* user behavior. This is because queries are often issued in scenarios in which user re-identification mechanisms are not available over multiple sessions.

2. *Portal content and news personalization:* Many online portals have strong user identification mechanisms by which returning users can be identified. In such cases, the content served to the user can be personalized. This approach is also used by news personalization engines, such as Google News, in which Gmail accounts are used for user identification. News personalization is usually based on implicit feedback containing user behavior (clicks), rather than explicit ratings.

3. *Computational advertising:* A form of recommendation, because it is desirable for companies to be able to identify advertisements for users based on a relevant context (Web page or search query). Therefore, many ideas from recommendation systems are directly used in the area of computational advertising.

4. *Reciprocal recommender systems:* In these cases, both the users and items have preferences (and not just the users). For example, in an online dating application, both parties have preferences, and a successful recommendation can be created only by satisfying both parties.

DATA ANALYTICS

Additional References

R1 Data Mining: Concepts and Techniques by Han, Kamber and Pei (Morgan Kaufman)

Introduction to Data Mining by Tan, Steinbach and Kumar (Pearson – First Edition) Chapters 6 and 7

Recommender Systems – The Textbook by Charu C. Agarwal (Chapter 7)

THANK YOU

Gowri Srinivasa
Professor,
Department of Computer Science
gsrinivasa@pes.edu