Week 3: User Profiles and Content Based RSs

CS3448: Recommender Systems /

CSX4207/ITX4207: Decision Support and Recommender Systems

Asst. Prof. Dr. Rachsuda Setthawong

Objectives

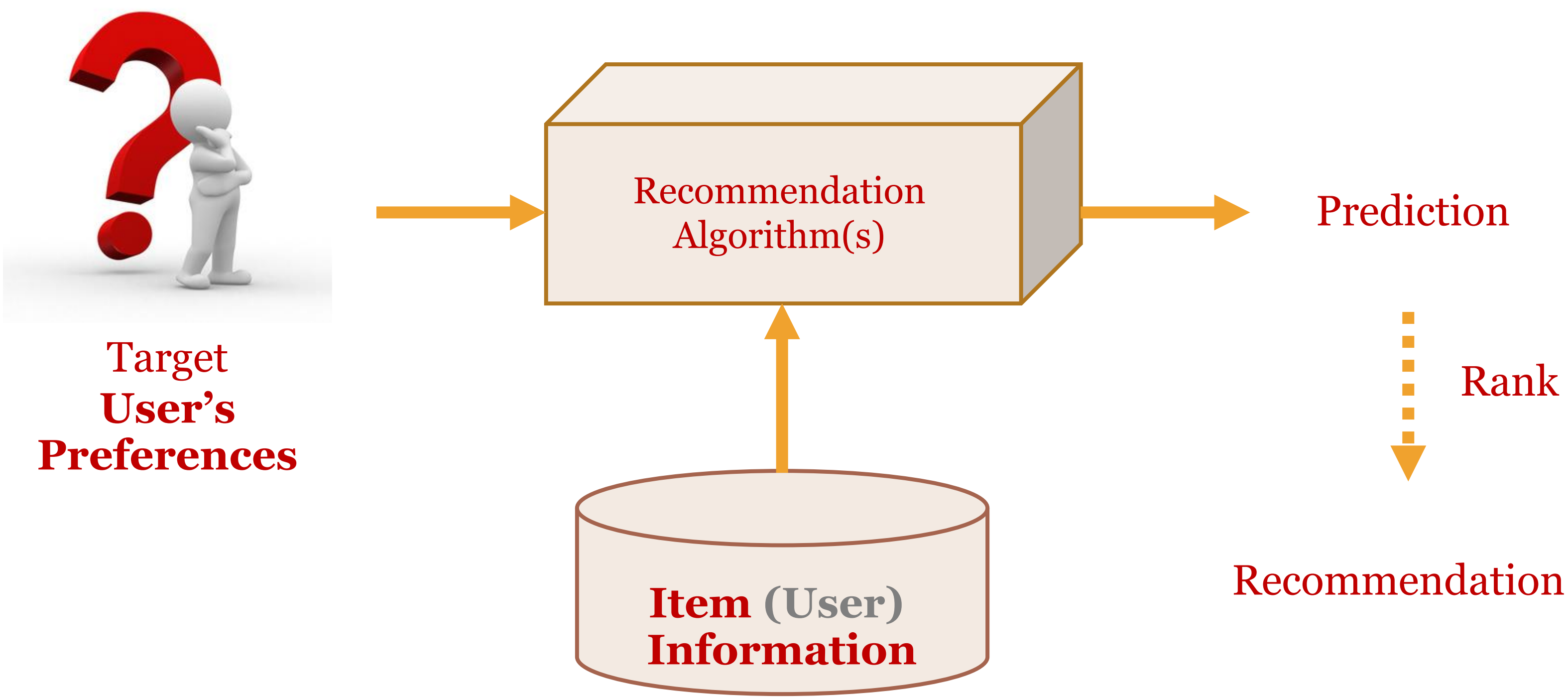
- To understand concepts of user profile and be able to construct it
- To understand concepts of content based Recommender Systems and be familiar with some algorithms in this approach
- To understand strong points and weak points of content based RSs

Outlines

- User Profiles and User Profiling
- Term Frequency and Invert Document Frequency (TF-IDF)
- How to Generate Recommendation Using Content Based Approach
  - Additional Similarity Measures
- A Technique for User Preference Profiling based on user behaviors on Facebook page categories
- Pros and Cons of Content-based RSs
- Vector Space Model and Recommending Items Using Nearest Neighbors
- Case Study
- Available Tools

Asst. Prof. Dr. Rachsuda Setthawong

How to Generate Prediction/Recommendation?



Content-based:

"Display more items similar to what I like."

Typical User Profile

- The description of what information is of interest to a user
  - An approximation of the real user's interests
- Compact representation in terms of memory and complexity
- Same representation as information filtered

Asst. Prof. Dr. Rachsuda Setthawong

Examples of Items and Descriptors

Item	**Descriptors**
Movie	Genre, Main Actor (Actress), Producer, Production, Release Year
Book	Genre, Title, Author(s), Publisher, Abstract, Year
Clothes	Fabric Type, Make, Color
Restaurant	Type, Rating, Opening Hours, Wifi Availability, Range of Price, Location, Service, Parking Available
Vehicle	Type, Make, Model, Color, Horse Power, Number of Doors, Year
Music	??
Attraction (tourism)	??

Asst. Prof. Dr. Rachsuda Setthawong

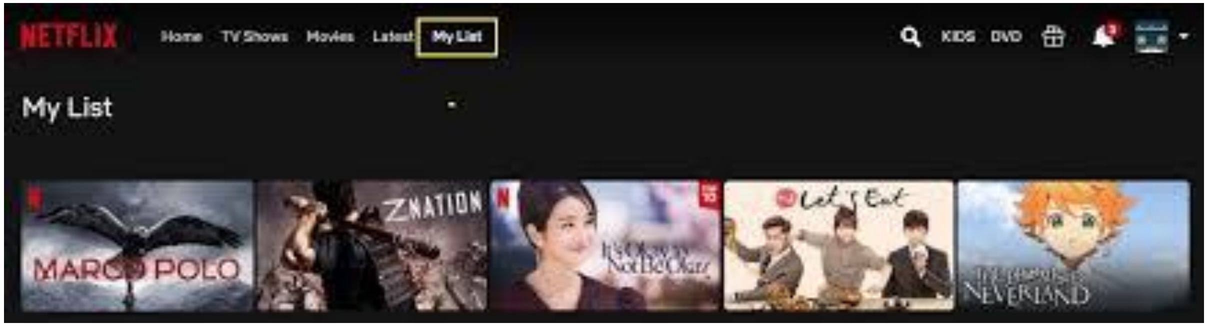
User Profiling (User Modeling) in RSs

- A process of user profile gathering, construction and representation
- Approaches:
  - Explicit Model (ask users straightforwardly)
    - Enabling building and editing profile by user are useful.
  - Implicit Model (observe their behaviors, e.g., click, view, buy)
    - May also infer profile from explicit user ratings
    - May require mapping of item preference and attribute preference

Asst. Prof. Dr. Rachsuda Setthawong

A Simple Approach of User Profiling for New Users

- Register for an account and answer couple questions before starting using the websites.
- Prompt users to give ratings on an initial items' set.



Asst. Prof. Dr. Rachsuda Setthawong

User (Item) Representation

- A vector of (features) attributes representing users (items)
  - $v = <a_1, a_2, \ldots, a_n>$, where v_i, $1 \leq i \leq n$, is an element of $dom(A_i)$
- Attribute value: $v[A_i]$ or $v.A_i$
  - The i^{th} value in vector v corresponding to attribute A_i.
  - The values can be in form of **weighted** keywords/terms, topics, ratings, etc.
    - Frequency
    - TF-IDF

Asst. Prof. Dr. Rachsuda Setthawong

Example of Vector Construction -- 1

Vehicle ID	**Type**	**Make**	**Model**	**Color**	**Year**
v_1	Sedan	Toyota	Altis	White	2015
v_2	Sedan	Mazda	2	Red	2014
v_3	Sedan	Mazda	2	White	2014
v_4	Wagon	Toyota	Fortuner	Black	2014
v_5	Pickup Truck	Toyota	Hilux	Green	2015
…					

v_1 = <'Sedan', 'Toyota', 'Altis', 'White', 2015>
v_2 = <'Sedan', 'Mazda', '2', 'Red', '2014>
v_3 = <'Sedan', 'Mazda', '2', 'White', '2014>
…

Example of Vector Construction -- 2

Book ID	**Title**	**Genre**	**Author(s)**	**Year**
v_1	The Great Gatsby	Novel	F. Scott Fitzgerald	1925
v_2	Lolita	Novel	Vladimir Nabokov	1955
v_3	Android Programming: The Big Nerd Ranch Guide	Computer	Brian Hardy, Bill Phillips	2013
v_4	Introduction to Data Mining	Computer	Pang-Ning Tan, Michigan State University, Michael Steinbach	2005
…				

v_1 = <‘The Great Gatsby’, ‘Novel’, ‘F. Scott Fitzgerald’, 1925>
v_2 = <‘Lolita’, ‘Novel’, ‘Vladimir Nabokov’, 1955>
v_3 = <‘Android Programming: The Big Nerd Ranch Guide’, ‘Computer’, ‘Brian Hardy, Bill Phillips’, 2013>
…

Asst. Prof. Dr. Rachsuda Setthawong

Note on the Vector of Attributes

- Select only subset attributes that contributes to the recommendation/are applicable to the algorithm used.
- May need to preprocess the data *(e.g., TF-IDF)* to come up with more suitable representation of attribute used.

Cannot be used directly!

Book ID	Title	Genre	Author(s)	Year
v_1	The Great Gatsby	Novel	F. Scott Fitzgerald	1925
v_2	Lolita	Novel	Vladimir Nabokov	1955
…				

Outlines

- User Profiles and User Profiling
- Term Frequency and Invert Document Frequency (TF-IDF)
- How to Generate Recommendation Using Content Based Approach
  - Additional Similarity Measures
- A Technique for User Preference Profiling based on user behaviors on Facebook page categories
- Pros and Cons of Content-based RSs
- Vector Space Model and Recommending Items Using Nearest Neighbors
- Case study
- Available Tools

Asst. Prof. Dr. Rachsuda Setthawong

Frequency

- The number of occurrences of a repeating event

(per unit time)

Book ID	Yale	World War I	Mining	Hotel	Algorithm	Formula	New York
V_1	50	30	3	12	0	1	15
V_2	0	0	0	24	0	0	13
V_3	0	0	25	0	30	55	1
V_4	0	0	0	0	15	5	0

$v_1 = <50, 30, 3, 12, 0, 1, 15>$
$v_2 = <0, 0, 0, 24, 0, 0, 13>$
$v_3 = <0, 0, 25, 0, 30, 55, 1>$
$v_4 = <0, 0, 0, 0, 15, 5, 0>$

Term-Frequency – Inverse Document Frequency (TF-IDF) – 1

Asst. Prof. Dr. Rachsuda Setthawong

- Overcome the problems of traditional keyword representation with the following assumption *that every word is equally important.*
- Prevent domination of the longer length documents over shorter ones when matching with the user profile.

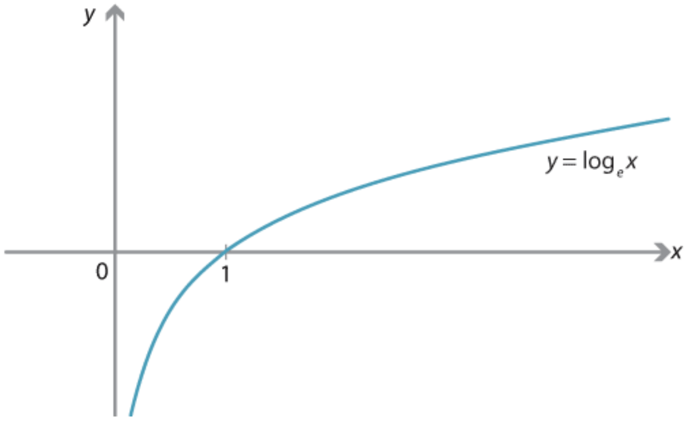
Term-Frequency – Inverse Document Frequency (TF-IDF) – 2

Asst. Prof. Dr. Rachsuda Setthawong

- TF: Frequency of a term appeared in a document (Normalization is required.)
- IDF: Decrease the significance of common terms (appeared in all documents.)

Asst. Prof. Dr. Rachsuda Setthawong

TF-IDF



$$TFIDF(t) = TF(t) \cdot IDF(t) \quad (1)$$

A measure calculated from an individual document:

$$TF(t) = \frac{n_{td}}{n_t} \quad (2)$$

A measure calculated from the whole dataset:

$$IDF(t) = log_e(\frac{N_d}{n_{dt}}) \quad (3)$$

where,

n_{td} : number of times that a term t appeared in a document d,
n_t : total number of terms in that document (for normalization),

consider an individual document

N_d : total number of documents, and
n_{dt} : number of documents containing the term t.
(if n_{dt} = 0, then IDF(t) = 0)

Consider all documents

Asst. Prof. Dr. Rachsuda Setthawong

$$TFIDF(t) = TF(t) \cdot IDF(t) \quad (1)$$

TF-IDF Examples

$$TF(t) = \frac{n_{td}}{n_t} \quad (2)$$

$$IDF(t) = log_e\left(\frac{N_d}{n_{dt}}\right) \quad (3)$$

Frequency

Book ID	Yale	World War I	Mining	Hotel	Algori-thm	Formula	New York	The
v_1	50	30	3	12	0	1	15	100
v_2	0	0	0	24	0	0	13	150
v_3	0	0	25	0	30	55	1	120
v_4	0	0	0	0	15	5	0	130

TF-IDF

TFIDF	Yale	World War I	Mining	Hotel	Algorithm	Formula	New York	The
v_1	0.328506	0.197103	0.009855	0.039421	0	0.001363	0.020451	0
v_2	0	0	0	0.08896	0	0	0.019999	0
v_3	0	0	0.075016	0	0.090019	0.068496	0.001245	0
v_4	0	0	0	0	0.069315	0.009589	0	0

Asst. Prof. Dr. Rachsuda Setthawong

Pros and Cons of TF-IDF

Pros	Cons
• Reduce weight of stop words, e.g., ‘a’, ‘an’, ‘the’, ‘is’, ‘am’, ‘are’ • Increase weight of key terms (not incidental ones) • Widely used to create a profile of a document • Feasibly used together with ratings in user profiles	• Insufficient frequency of key terms could result in low TF-IDF • Require preprocessing to handle phrases, e.g., ‘World War I’ ▫ N-gram

Outlines

- User Profiles and User Profiling
- Term Frequency and Invert Document Frequency (TF-IDF)
- How to Generate Recommendation Using Content Based Approach
  - Additional Similarity Measures
- A Technique for User Preference Profiling based on user behaviors on Facebook page categories
- Pros and Cons of Content-based RSs
- Vector Space Model and Recommending Items Using Nearest Neighbors
- Case study
- Available Tools

Asst. Prof. Dr. Rachsuda Setthawong

How to Generate Recommendation Using Content Based Approach – 1

Target
User's Preferences (keyword vectors)
- Present/Absent (0/1)
- Frequency
- TF-IDF
- etc.

Recommendation Algorithm(s)

Item **(User)** **Information (keyword matrix)**
- Present/Absent (0/1)
- Frequency
- TF-IDF
- etc.

Prediction

Rank

Recommendation

Content-based:
"Display more items similar to what I like."

How to Generate Recommendation Using Content Based Approach – 2

1. Constructing user profiles
2. Predicting items
3. Updating user profiles (for system maintenance)

Step 1: Constructing User Profiles – 1

- Observation: a user has experienced several items.
- Commonly used techniques:
  - Aggregation
  - Normalization
  - Weighting

Asst. Prof. Dr. Rachsuda Setthawong

Constructing User Profiles – 2

A. Simple unary (adding all items)
B. Unary with threshold (adding items whose rating above threshold)
C. Weighting with positive rated items (adding all items but with different weights wrt positive rating)
D. Weighting with positive and negative rated items (adding all items but with different weights wrt positive and negative ratings)

Asst. Prof. Dr. Rachsuda Setthawong

Constructing User Profiles – 3 Example 1

- $p_{Tom} = \{v_1, v_2\}$

  v_1 = <50, 30, 3, 12, 0, 1, 15, 100> (Frequency)

  v_2 = <0, 0, 0, 24, 0, 0, 13, 150> (Frequency)

- A. Simple unary

  v_{Tom} = <50, 30, 3, 36, 0, 1, 28, 250>

Asst. Prof. Dr. Rachsuda Setthawong

Constructing User Profiles – 4 Example 2

- $p_{Tom} = \{<v_1, 5.0>, <v_2, 2.0>, <v_3, 4.0>\}$

  $v_1 = <50, 30, 3, 12, 0, 1, 15, 100>$ (Frequency)

  $v_2 = <0, 0, 0, 24, 0, 0, 13, 150>$ (Frequency)

  $v_3 = <20, 0, 2, 14, 0, 0, 10, 120>$ (Frequency)

- B. Unary with threshold (>**3.0**)

- $v_{Tom} = <70, 30, 5, 26, 0, 1, 25, 220>$

Asst. Prof. Dr. Rachsuda Setthawong

Note: Classical Information Retrieval (IR) vs RS

- Classical IR – based methods are based on keywords.
  - Retrieve relevant items based on explicitly searching keywords as user's input
- RS – user profile (preferences) are rather learned than explicitly elicited.
  - Recommend ranked items (without a query) that a target user should like.

Step 2: Predicting Items

- IDEA: Calculating similarity between a target user's vector (user profile) and existing items.
- Similarity measures for numeric attributes:
  - Distance based similarity
    - e.g., converting Euclidean distance to similarity
  - Cosine similarity
  - Pearson correlation coefficient
  - Etc.

Asst. Prof. Dr. Rachsuda Setthawong

Prediction Example – Step 1

Suppose that v_{Tom} = <50, 30, 3, 36, 0, 1, 28, 250> (refer to slide no. 25)

What book to suggest for Tom?

First! Need to calculate TF-IDF for Tom.

TF-IDF_{Tom}

	Yale	World War I	Mining	Hotel	Algorithm	Formula	New York	The
v_Tom	0.174158	0.104495	0.005225	0.062697	0	0.000723	0.020239	0

TF-IDF *of unreaded book*

TFIDF	Yale	World War I	Mining	Hotel	Algorithm	Formula	New York	The
v_3	0	0	0.075016	0	0.090019	0.068496	0.001245	0
v_4	0	0	0	0	0.069315	0.009589	0	0

Asst. Prof. Dr. Rachsuda Setthawong

Prediction Example – Step 2

- Distance based similarity (Euclidean)
  - Euclidean distance: $distance(v_1, v_2) = \sqrt{\sum_{i \in Item} (v_{1i} - v_{2i})^2}$
  - $similarity(v_1, v_2) = \frac{1}{1+(distance(v_1,v_2))}$

Interpretation:
The closer to 1 the more similar.

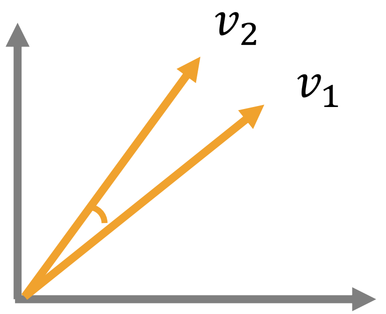
- Examples:
  - similarity (v_{Tom}, v_3) = 0.7992
  - **similarity** (v_{Tom}, $\mathbf{v_4}$) = **0.8165**

Suggest the book that more similar to Tom's!

Note: Not suggest the items that are already experienced by the target user

Asst. Prof. Dr. Rachsuda Setthawong

Prediction Example – *Other similarity measure*



- Cosine similarity
  - A measure of similarity between two non-zero vectors of an inner product space (dot product) that measures the cosine of the angle between them.
  - $cosine\_sim(v_1, v_2) = \frac{v_1 \cdot v_2}{\|v_1\| \llbracket v_2 \rrbracket}$

Interpretation:
- Range of possible value is -1 and 1.
- The closer to 1 the more similar.

The dot product of two vectors:

$$v_1 . v_2 = (v_{11} \times v_{21}) + (v_{12} \times v_{22}) + \cdots (v_{1n} \times v_{2n})$$

The norm of a vector:

$$\|v\| = \sqrt{a_1^2 + a_2^2 + \cdots + a_n^2}$$

Source: https://en.wikipedia.org/wiki/Cosine_similarity

Asst. Prof. Dr. Rachsuda Setthawong

Prediction Example – *Other similarity measure (Cont.)*

- Examples:
  - **cosine_sim** (v_{TFIDF_Tom}, $\mathbf{v_{TFIDF3}}$) = **0.0161**
  - cosine_sim (v_{TFIDF_Tom}, v_{TFIDF4}) = 0.0005

Suggest the book that more similar to Tom's!

Note: Check the complete calculation in calculation_w3.xlsx

Asst. Prof. Dr. Rachsuda Setthawong

Dice Coefficient

(Categorical Attributes)

$$Dice\_coefficient(v_1, v_2) = \frac{2 \times |keywords(v_1) \cap keywords(v_2)|}{|keywords(v_1)| + |keywords(v_2)|}$$

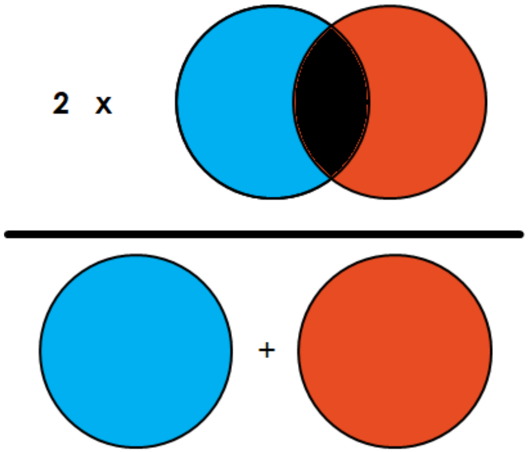


Image resource: https://en.wikipedia.org/wiki/Dice-S%C3%B8rensen_coefficient

Asst. Prof. Dr. Rachsuda Setthawong

Dice Coefficient - Example

Word Occurrence in 2 books

Book ID	Yale	World War I	Mining	Hotel	Algorithm	Formula	New York	The
v_1	yes	yes	yes	yes		yes	yes	yes
v_2				yes			yes	yes

- $Dice\_coefficient(v_1, v_2) = \frac{2 \times |keywords(v_1) \cap keywords(v_2)|}{|keywords(v_1)| + |keywords(v_2)|}$
- $= \frac{2 \times 3}{7+3} = 0.6$

Asst. Prof. Dr. Rachsuda Setthawong

Step 3: Updating User Profiles - 1

- Observation: a user buy/use new items from time to time.
- Consequence:
  - Existing user profiles may be outdate.
- Remedy:
  - Allow incremental update of user profiles.
  - Use time decay reflecting item aging to improve accuracy of prediction over time.

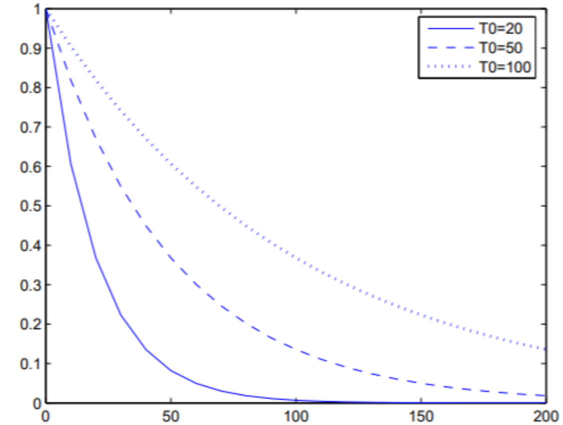
Asst. Prof. Dr. Rachsuda Setthawong

Step 3: Updating User Profiles – 2
Time Decay Strategies

- Time Window
  - consider only data in a window (sliding window) that contains either the latest N instances
    - E.g., the latest 1000 ratings
  - consider only the instances contained in the latest time interval
    - E.g., all ratings given in the last 24 hours.
- Time Decay Function
  - Applied as a weight (when calculating similarity (*sim*) between v_1 and v_2, multiplying *sim* with an exponential decay functions)

$$f(t) = e^{-\lambda \cdot t} \quad \text{where } \lambda = 1/T_0$$

Figure 2: the curves of time function using different T_0



Ref: https://dl.acm.org/doi/pdf/10.1145/1099554.1099689

Asst. Prof. Dr. Rachsuda Setthawong

Outlines

- User Profiles and User Profiling
- Term Frequency and Invert Document Frequency (TF-IDF)
- How to Generate Recommendation Using Content Based Approach
  - Additional Similarity Measures
- A Technique for User Preference Profiling based on user behaviors on Facebook page categories
- Pros and Cons of Content-based RSs
- Vector Space Model and Recommending Items Using Nearest Neighbors
- Case study
- Available Tools

A Technique for User Preference Profiling based on user behaviors on Facebook page categories – 1

- User preferences profiling from social networking data is essential in both social networking mining and recommender systems.
- Facebook provides many useful, both implicit and explicit social data.
- It is a challenge to explore Facebook user behaviors in creating customized user preferences.

Ref: https://ieeexplore.ieee.org/document/7886077

Asst. Prof. Dr. Rachsuda Setthawong

A Technique for User Preference Profiling based on User Behaviors on Facebook Page Categories – 2

Defining a set of desirable user preferences

- Feasible Technique: Latent Dirichlet allocation (LDA), pre-defined attributes wrt application's objectives

Gathering and preprocessing Data

- Create user's feature vectors of aggregated frequencies of 'like' pages and/or of 'share' page posts
- Obtains also their user preferences (16 class attributes)

Constructing classifiers

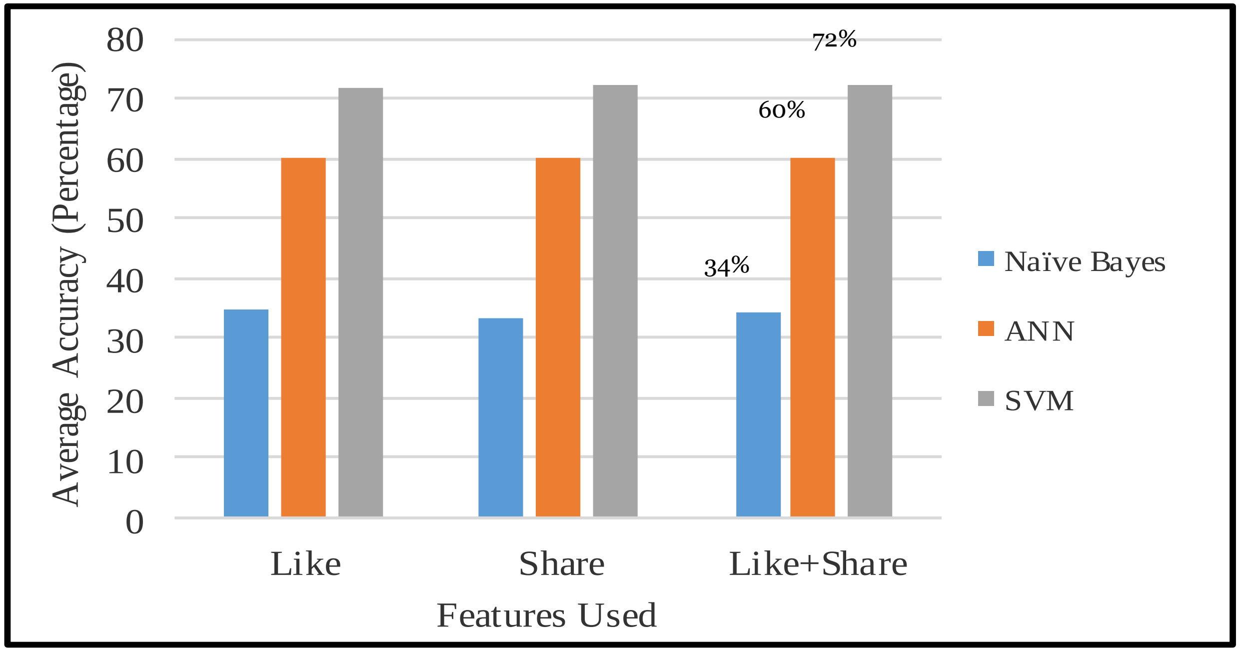
- Naïve Bayes Classifier (NB)
- Artificial Neural Network (ANN)
- Support Vector Machine (SVM)

Constructing user preferences

Use the classifiers constructed from the third step to classify new users' preferences given a vector of their behavior attributes.

Asst. Prof. Dr. Rachsuda Setthawong

Percentage of Average Accuracy of 16 User Preferences Using 3 Feature Sets with 3 Algorithms



Outlines

- User Profiles and User Profiling
- Term Frequency and Invert Document Frequency (TF-IDF)
- How to Generate Recommendation Using Content Based Approach
  - Additional Similarity Measures
- A Technique for User Preference Profiling based on user behaviors on Facebook page categories
- Pros and Cons of Content-based RSs
- Vector Space Model and Recommending Items Using Nearest Neighbors
- Case study
- Available Tools

Asst. Prof. Dr. Rachsuda Setthawong

Content Based RSs

Pros	Cons
• Easy to compute • Has structured and available information in some domains, e.g., Book and Movie. • Easily address new item/user problem (cold start problem)	• Cannot handle independencies ▫ E.g., • I like ***Leonardo Dicaprio*** in 'Catch Me If You Can' but not when he plays 'The Revenant'. • I love romantic and comedy movie but not scary, comedy movie.

Reference: https://en.wikipedia.org/wiki/Cold_start

Outlines

- User Profiles and User Profiling
- Term Frequency and Invert Document Frequency (TF-IDF)
- How to Generate Recommendation Using Content Based Approach
  - Additional Similarity Measures
- A Technique for User Preference Profiling based on user behaviors on Facebook page categories
- Pros and Cons of Content-based RSs
- Vector Space Model and Recommending Items Using Nearest Neighbors
- Case study
- Available Tools

Asst. Prof. Dr. Rachsuda
Setthawong

The Vector Space Model

- Enhancements
  - Remove stop words , article, preposition, etc ("a", "the", ..)
  - Stemming
    - Use with caution. (stemming of “university” and “universal” is “univers”)
  - User-specified thresholds (e.g., top 100 frequent words or words with TF-IDF greater than its global average)
  - Detection of phrases as terms (such as United Nations)
- Limitations
  - Address neither semantic nor sentiment in texts.
    - Example: My parent do not eat vegetarian since I was very young. So am I.

Asst. Prof. Dr. Rachsuda Setthawong

Recommending Items: Nearest Neighbors – 1

Document Vector + Rating

- Given
  - A set of documents *D* **already rated** by the user (like/dislike)
  - Similarity measure for two document vectors
  - Input: vectors of terms in each document,
    - The ratings of each doc can be obtained implicitly or explicitly.
- Find the *k* nearest neighbors (k-NN) of a not-yet-seen item *i* in *D*
  - Take these **ratings** to predict a rating/vote for *i*
    - Majority voting (If the majority items were liked by the user, so do *i*.)
  - Variations:
    - neighborhood size
    - lower/upper similarity thresholds
    - Weighting of the votes based on the degree of similarity

Asst. Prof. Dr. Rachsuda Setthawong

An Illustration of Recommending Items: Nearest Neighbors

A Set of Document *D*

ItemID	attr1	attr2	att3
16			
17			
18			
…			
100			

A Similarity Matrix

ItemID	16	17	18	…	100
1	sim(16, 1)				
2					
3					
…					
15					

A Target User

ItemID	attr1	attr2	att3	rating
1				3
2				5
3				4
…				…
15				5

Assumption: rating > 3 implies 'like'
Otherwise, it is 'dislike'

Find 3-nn of the item 16, (e.g., the item 1, 2, 15)

Take these ratings to predict a rating/vote for *the item 16*
= Majority Vote of (2 likes, 1 dislike) → like

Asst. Prof. Dr. Rachsuda Setthawong

An Illustration of Recommending Items: Nearest Neighbors
Variation: neighborhood size

A Set of Document *D*

ItemID	attr1	attr2	att3
16			
17			
18			
…			
100			

A Similarity Matrix

ItemID	16	17	18	…	100
1	sim(16, 1)				
2					
3					
…					
15					

A Target User

ItemID	attr1	attr2	att3	rating
1				3
2				5
3				4
…				…
15				5

Assumption: rating > 3 implies 'like'
Otherwise, it is 'dislike'

Find 5-nn of the item 16, (e.g., the item 1, 2, 7, 10, 15)

Take these ratings to predict a rating/vote for *the item 16*
= *Majority Vote of (2 likes, 3 dislikes) → dislike*

Asst. Prof. Dr. Rachsuda Setthawong

An Illustration of Recommending Items: Nearest Neighbors

Variation: upper similarity thresholds

(e.g., news domain to select not too similar news)

A Set of Document *D*

ItemID	attr1	attr2	att3
16			
17			
18			
…			
100			

A Target User

ItemID	attr1	attr2	att3	rating
1				3
2				5
3				4
…				…
15				5

Assumption: rating > 3 implies 'like'
Otherwise, it is 'dislike'

A Similarity Matrix

ItemID	16	17	18	…	100
1	sim(16, 1)				
2					
3					
…					
15					

Find 3-nn of the item 16, that
Similarity must also be less than 0.9 → (e.g., the item 1, 2, 3)
Sim(16.3) =0.6, rating 2
Sim(16,1) = 0.6
Sim(16,2) = 0.8
~~Sim(16,15) = 0.9~~

Take these ratings to predict a rating/vote for *the item 16*
= *Majority Vote of (1 like, 2 dislikes) → dislike*

Asst. Prof. Dr. Rachsuda Setthawong

An Illustration of Recommending Items: Nearest Neighbors

Variation: Weighting of the votes based on the degree of similarity

A Set of Document *D*

ItemID	attr1	attr2	att3
16			
17			
18			
…			
100			

A Target User

ItemID	attr1	attr2	att3	rating
1				3
2				5
3				4
…				…
15				5

Assumption: rating > 3 implies 'like'
Otherwise, it is 'dislike'

A Similarity Matrix

ItemID	16	17	18	…	100
1	sim(16, 1)				
2					
3					
…					
15					

Find 3-nn of the item 16, (e.g., the item 1, 2, 15), where
Sim(16,1) = 0.6
Sim(16,2) = 0.8
Sim(16,15) = 0.9

Take these ratings to predict a rating/vote for *the item 16*
*= **Majority Vote** of (2 likes (0.8+0.9) > 1 dislike (0.6))*
(1.7 > 0.6) → like

Recommending Items: Nearest Neighbors – 2

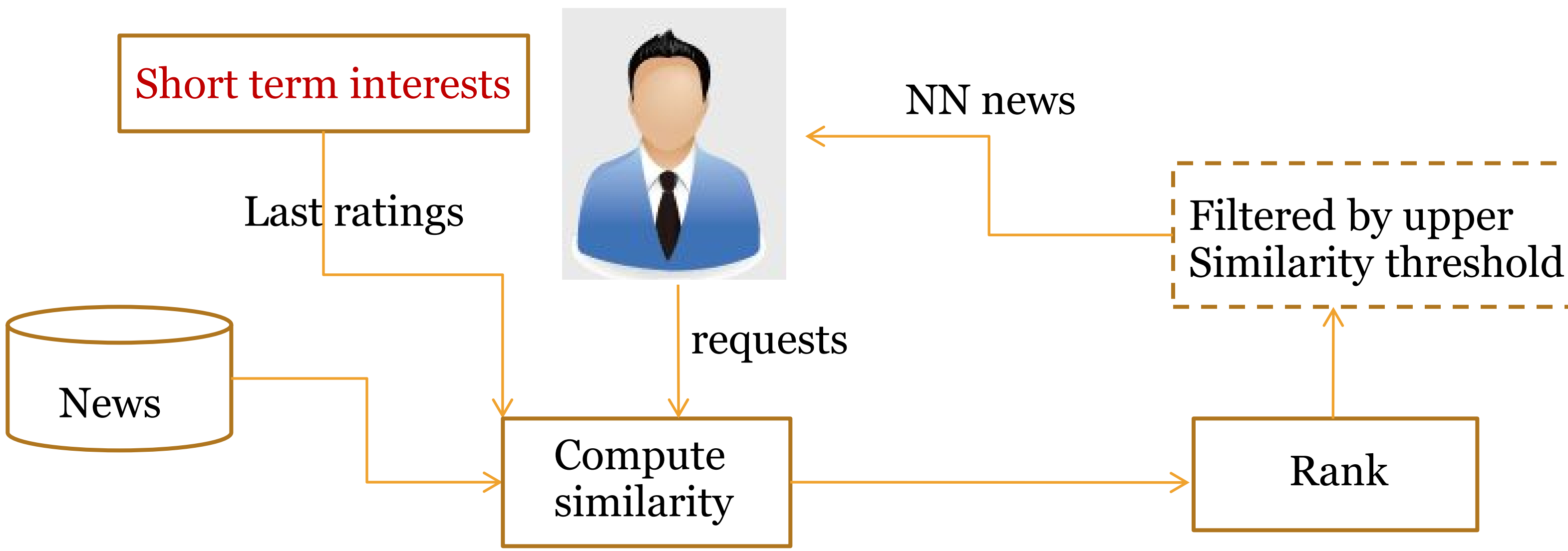
- Good to model short-term interests / follow-up stories.
- Used in combination with method to model long-term preferences.

Outlines

- User Profiles and User Profiling
- Term Frequency and Invert Document Frequency (TF-IDF)
- How to Generate Recommendation Using Content Based Approach
  - Additional Similarity Measures
- A Technique for User Preference Profiling based on user behaviors on Facebook page categories
- Pros and Cons of Content-based RSs
- Vector Space Model and Recommending Items Using Nearest Neighbors
- Case study
- Available Tools

Asst. Prof. Dr. Rachsuda Setthawong

Case Study: the Personalized and Mobile News Access System -- 1/3

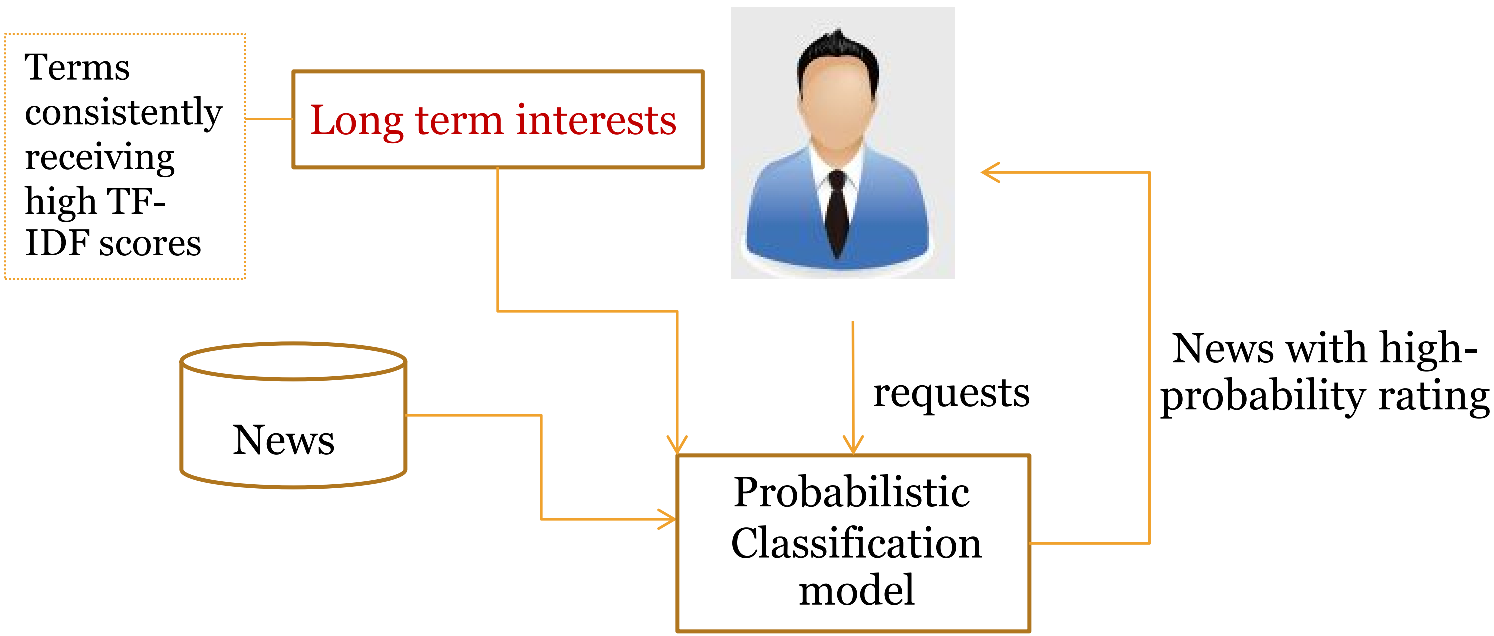


"Short term interests' model"

Note: Set an ***upper threshold*** for *item similarity* to prevent the system from recommending items that the user most probably has already seen.

Asst. Prof. Dr. Rachsuda Setthawong

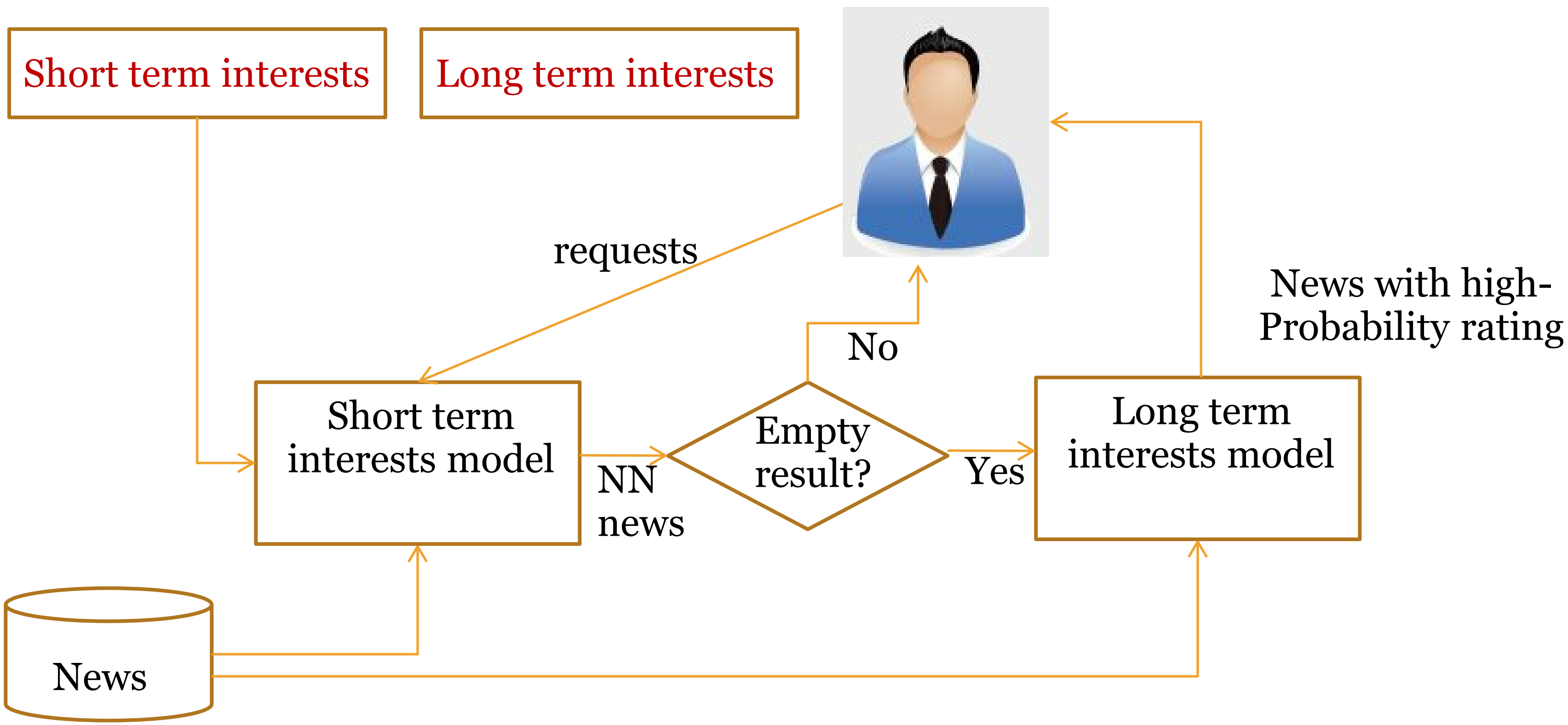
Case Study: the Personalized and Mobile News Access System -- 2/3



"Long term interests model"

Asst. Prof. Dr. Rachsuda Setthawong

Case Study: the Personalized and Mobile News Access System -- 3/3



"Combined short term and long term interests model"

Asst. Prof. Dr. Rachsuda Setthawong

Advantages of kNN-based Method

- Simple to implement.
- Adapt quickly to recent changes.
- Relatively small number of ratings is sufficient to make a prediction of reasonable quality.

Outlines

- User Profiles and User Profiling
- Term Frequency and Invert Document Frequency (TF-IDF)
- How to Generate Recommendation Using Content Based Approach
  - Additional Similarity Measures
- A Technique for User Preference Profiling based on user behaviors on Facebook page categories
- Pros and Cons of Content-based RSs
- Vector Space Model and Recommending Items Using Nearest Neighbors
- Case study
- Available Tools

Asst. Prof. Dr. Rachsuda Setthawong

Available Tools

- LensKit[1]
- Machine Learning Toolkits:
  - Microsoft Machine Learning Studio[2]
  - Rapidminer Studio[3]
  - Surprise (scikit) in Python[4]
  - Matlab
  - R
  - Etc.

[1]Source: http://lenskit.org
[2]Source: https://azure.microsoft.com/en-us/blog/building-recommender-systems-with-azure-machine-learning-service/
[3]Source:: https://academy.rapidminer.com/learn/article/recommendation-engine-youre-going-to-love-this-movie
[4]Source:: https://surpriselib.com/

Asst. Prof. Dr. Rachsuda Setthawong

Practice 3-1

TFIDF	Yale	World War I	Mining	Hotel	Algorithm	Formula	New York	The
v_1	0.328506	0.197103	0.009855	0.039421	0	0.001363	0.020451	0
v_2	0	0	0	0.08896	0	0	0.019999	0
v_3	0	0	0.075016	0	0.090019	0.068496	0.001245	0
v_4	0	0	0	0	0.069315	0.009589	0	0

- Given $p_{Pete} = \{v_a, v_b\}$

  v_a = <10, 20, 0, 0, 0, 10, 30, 150> (Frequency)

  v_b = <0, 5, 20, 5, 0, 0, 10, 100> (Frequency)

- Build user profile vector for Pete (simple unary)
- Predict the book from the example that Pete will like using
  - Cosine similarity
  - Note: use the value of IDF that is already created by the dataset.

Asst. Prof. Dr. Rachsuda Setthawong

Practice 3-2

- Given the dataset below, design and explain user profiles (how feature vectors look like).
- Recommend 3 vehicles to the following user:
  - Mandy is a big fan of Toyota and drove Altis whose model was of year 2010.
- Also explain how you generate the results and show the computation.
  - Similarity measures, item aggregation techniques, etc.

Vehicle ID	Type	Make	Model	Color	Year
v_1	Sedan	Toyota	Altis	White	2015
v_2	Sedan	Mazda	2	Red	2014
v_3	Sedan	Mazda	2	White	2014
v_4	Wagon	Toyota	Fortuner	Black	2014
v_5	Pickup Truck	Toyota	Hilux	Green	2015
v_6	Pickup Truck	Ford	Ranger	Black	2014
v_7	Wagon	Mitsubishi	Space Wagon	Black	2013
v_8	Sedan	Toyota	Camry	Red	2015
v_9	Sedan	Nissan	Almera	White	2012