Department of Computer Science

Examination paper for TDT4215 Web Intelligence

Academic contact during examination: Prof. Jon Atle Gulla
Phone: 73591847

Examination date: 6 June 2017
Examination time: 09:00 – 13:00
Permitted examination support material: D (No printed or hand-written support material is allowed. A specific basic calculator is allowed)

Other information:

Language: English
Number of pages (front page excluded): 4
Number of pages enclosed: 5

Informasjon om trykking av eksamensoppgave
Originalen er:

1-sidig ☐ **2-sidig** ☐

sort/hvit ☐ **farger** ☐

skal ha flervalgskjema ☐

Checked by:

Date Signature

Students will find the examination results in Studentweb. Please contact the department if you have questions about your results. The Examinations Office will not be able to answer this.

Question 1. Neighborhood-Based Collaborative Filtering

From Aggarwal: Recommender Systems, Chapter 2.

(a) Explain the "*long tail property*"?
(b) Given the following User-Item table, estimate (using a *user-based approach*) the rating of *Item1* for *User3* and the rating of *Item4* for *User5*.

	Item1	*Item2*	*Item3*	*Item4*	*Item5*
User1	5	4	1	2	2
User2	3	4	2	1	3
User3	**?**	5	3	2	1
User4	1	2	5	3	4
User5	2	3	5	**?**	3

(c) What are the main strengths and weaknesses of the Neighborhood-based methods?

Answers:
(a) The distribution of ratings among items often satisfies a property in real-world settings, which is referred to as the long-tail property. According to this property, only a small fraction of the items are rated frequently. Such items are referred to as popular items. The vast majority of items are rated rarely. This results in a highly skewed distribution of the underlying ratings. An example of a skewed rating distribution is illustrated in Figure. The X-axis shows the index of the item in order of decreasing frequency, and the Y -axis shows the frequency with which the item was rated. It is evident that most of the items are rated only a small number of times. Such a rating distribution has important implications for the recommendation process.

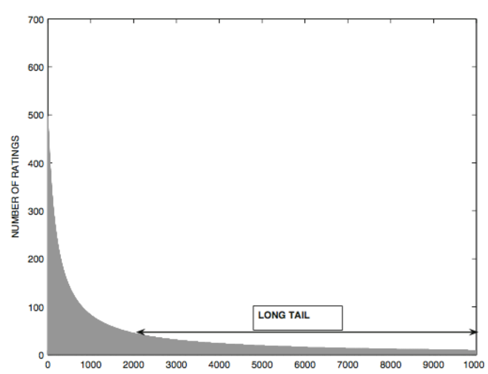

Figure 2.1: The long tail of rating frequencies

(b)

$$u_1 = \frac{5+4+1+2+2}{5} = \frac{14}{5} = 2.8$$

$$u_2 = \frac{3+4+2+1+3}{5} = \frac{13}{5} = 2.6$$

$$u_3 = \frac{5+3+2+1}{4} = \frac{11}{4} = 2.75$$
$$u_4 = \frac{1+2+5+3+4}{5} = \frac{15}{5} = 3$$
$$u_5 = \frac{2+3+5+3}{4} = \frac{13}{4} = 3.25$$

For User3, Item1:

$$Sim(1,3) = Pearson(1,3)$$
$$= \frac{(4-2.8)\times(5-2.75)+(1-2.8)\times(3-2.75)+(2-2.8)\times(2-2.75)+(2-2.8)\times(1-2.75)}{\sqrt{1.2^2+1.8^2+0.8^2+0.8^2}\times\sqrt{2.25^2+0.25^2+0.75^2+1.75^2}}$$
$$= \frac{4.25}{\sqrt{5.96}\times\sqrt{8.75}} \approx 0.5885$$

$$Sim(2,3) = Pearson(2,3)$$
$$= \frac{(4-2.6)\times(5-2.75)+(2-2.6)\times(3-2.75)+(1-2.6)\times(2-2.75)+(3-2.6)\times(1-2.75)}{\sqrt{1.4^2+0.6^2+1.6^2+0.4^2}\times\sqrt{2.25^2+0.25^2+0.75^2+1.75^2}}$$
$$= \frac{3.5}{\sqrt{5.04}\times\sqrt{8.75}} \approx 0.5270$$

$$Sim(3,4) = Pearson(3,4)$$
$$= \frac{(2-3)\times(5-2.75)+(5-3)\times(3-2.75)+(3-3)\times(2-2.75)+(4-3)\times(1-2.75)}{\sqrt{1^2+2^2+0^2+1^2}\times\sqrt{2.25^2+0.25^2+0.75^2+1.75^2}}$$
$$= \frac{-3.5}{\sqrt{6}\times\sqrt{8.75}} \approx -0.4831$$

$$Sim(3,5) = Pearson(3,5)$$
$$= \frac{(3-3.25)\times(5-2.75)+(5-3.25)\times(3-2.75)+(3-3.25)\times(1-2.75)}{\sqrt{1^2+2^2+0^2+1^2}\times\sqrt{2.25^2+0.25^2+0.75^2+1.75^2}}$$
$$= \frac{0.3125}{\sqrt{3.1875}\times\sqrt{8.1875}} \approx 0.0612$$

The top-2 closest users to User3 are User1 and User2 according to Pearson similarities.

$$\hat{r}_{31} = 2.75 + \frac{0.5885\times 2.2+0.5270\times 0.4}{0.5885+0.5270} \approx 4.1$$

Or the top-3 closest users to User3 are User1, User2 and User5.

$$\hat{r}_{31} = 2.75 + \frac{0.5885\times 2.2+0.5270\times 0.4+0.0612\times(-1.25)}{0.5885+0.5270+0.0612} \approx 3.97$$

For User5, Item4:

$$Sim(1,5) = Pearson(1,5)$$
$$= \frac{(5-2.8)(2-3.25)+(4-2.8)(3-3.25)+(1-2.8)(5-3.25)+(2-2.8)(3-3.25)}{\sqrt{2.2^2+1.2^2+1.8^2+0.8^2}\times\sqrt{1.25^2+0.25^2+1.75^2+0.25^2}}$$
$$= \frac{-0.5}{\sqrt{10.16}\times\sqrt{4.75}} \approx -0.0720$$

$$Sim(2,5) = Pearson(2,5)$$
$$= \frac{(3-2.6)(2-3.25)+(4-2.6)(3-3.25)+(2-2.6)(5-3.25)+(3-2.6)(3-3.25)}{\sqrt{0.4^2+1.4^2+0.6^2+0.4^2} \times \sqrt{1.25^2+0.25^2+1.75^2+0.25^2}}$$
$$= \frac{-2}{\sqrt{2.64} \times \sqrt{4.75}} \approx -0.5648$$

$Sim(3,5) = 0.0612$ (See last question)

$$Sim(4,5) = Pearson(4,5)$$
$$= \frac{(1-3)(2-3.25)+(2-3)(3-3.25)+(5-3)(5-3.25)+(4-3)(3-3.25)}{\sqrt{2^2+1^2+2^2+1^2} \times \sqrt{1.25^2+0.25^2+1.75^2+0.25^2}}$$
$$= \frac{-2}{\sqrt{10} \times \sqrt{4.75}} \approx 0.8706$$

The top-2 closest users to User5 are User3 and User4 according to Pearson similarities

$$\hat{r}_{54} = 3.25 + \frac{0.8706 \times 0 + 0.0612 \times (-0.75)}{0.8706 + 0.0612} \approx 3.2$$

(c)

Weakness:

- **Impractical offline** phase in large scale setting
  - Possible solution: clustering
- **Sparsity** of the rating matrix

Strenghts:

- **Simple** and **Intuitive** approach
- **Interpretability** of the provided recommendation (e.g. item-based methods)
- **Online** phase generally **efficient**

Question 2. Model-Based Collaborative Filtering

From Aggarwal: Recommender Systems, Chapter 3.

(a) What are the main advantages of model-based methods in comparison to NN-based methods?

(b) Given the following binary ratings matrix, estimate the ratings of *Item3* for *User4* and the rating of *Item5* for *User5* using the Bayes method.

	Item1	*Item2*	*Item3*	*Item4*	*Item5*
User1	1	1	-1	-1	1
User2	-1	1	1	1	-1
User3	1	-1	-1	1	-1
User4	1	-1	**?**	1	1

User5	1	-1	-1	1	**?**

(c) What is the main intuition behind the Matrix Factorization approach? What are the main methods?

Answers:

(a) Model-based recommender systems often have a number of advantages over neighborhood-based methods:

1. Space-efficiency: Typically, the size of the learned model is much smaller than the original ratings matrix. Thus, the space requirements are often quite low. On the other hand, a user-based neighborhood method might have O(m2) space complexity, where m is the number of users. An item-based method will have O(n2) space complexity.

2. Training speed and prediction speed: One problem with neighborhood-based methods is that the pre-processing stage is quadratic in either the number of users or the number of items. Model-based systems are usually much faster in the preprocessing phase of constructing the trained model. In most cases, the compact and summarized model can be used to make predictions efficiently.

3. Avoiding overfitting: Overfitting is a serious problem in many machine learning algorithms, in which the prediction is overly influenced by random artifacts in the data. This problem is also encountered in classification and regression models. The summarization approach of model-based methods can often help in avoiding overfitting. Furthermore, regularization methods can be used to make these models robust.

(b)

for User4, Item3:

$$P(r_{43} = 1 | r_{41}, r_{42}, r_{44}, r_{45}) \propto P(r_{43} = 1) P(r_{41} = 1 | r_{43} = 1)$$
$$P(r_{42} = -1 | r_{43} = 1) P(r_{44} = 1 | r_{43} = 1) P(r_{45} = 1 | r_{43} = 1)$$

$$P(r_{43} = 1) = \frac{1}{4}$$

$$P(r_{41} = 1 | r_{43} = 1) = \frac{0}{1} = 0$$

$$P(r_{42} = -1 | r_{43} = 1) = \frac{0}{1} = 0$$

$$P(r_{44} = 1 | r_{43} = 1) = \frac{1}{1} = 1$$

$$P(r_{45} = 1 | r_{43} = 1) = \frac{0}{1} = 0$$

$$P(r_{43} = 1 | r_{41}, r_{42}, r_{44}, r_{45}) \propto \left(\frac{1}{4}\right)(0)(0)(1)(0) = 0$$

$$P(r_{43} = -1 | r_{41}, r_{42}, r_{44}, r_{45}) \propto P(r_{43} = -1) P(r_{41} = 1 | r_{43} = -1)$$
$$P(r_{42} = -1 | r_{43} = -1) P(r_{44} = 1 | r_{43} = -1) P(r_{45} = 1 | r_{43} = -1)$$

$$P(r_{43} = -1) = \frac{3}{4}$$

$P(r_{41} = 1 | r_{43} = -1) = \frac{3}{3} = 1$

$P(r_{42} = -1 | r_{43} = -1) = \frac{2}{3}$

$P(r_{44} = 1 | r_{43} = -1) = \frac{2}{3}$

$P(r_{45} = 1 | r_{43} = -1) = \frac{1}{2}$

$P(r_{43} = -1 | r_{41}, r_{42}, r_{44}, r_{45}) \propto \left(\frac{3}{4}\right)(1)\left(\frac{2}{3}\right)\left(\frac{2}{3}\right)\left(\frac{1}{2}\right) = \frac{1}{6} \approx 0.1667$

The rating r_{43} has a higher probability of taking on the value of -1.

for User5, Item5:

$P(r_{55} = 1 | r_{51}, r_{52}, r_{53}, r_{54}) \propto$
$P(r_{55} = 1) P(r_{51} = 1 | r_{55} = 1) P(r_{52} = -1 | r_{55} = 1) P(r_{53} = -1 | r_{55} = 1) P(r_{54} = 1 | r_{55} = 1)$

$P(r_{55} = 1) = \frac{2}{4} = \frac{1}{2}$

$P(r_{51} = 1 | r_{55} = 1) = \frac{2}{2} = 1$

$P(r_{52} = -1 | r_{55} = 1) = \frac{1}{2}$

$P(r_{53} = -1 | r_{55} = 1) = \frac{1}{1} = 1$

$P(r_{54} = 1 | r_{55} = 1) = \frac{1}{2}$

$P(r_{55} = 1 | r_{51}, r_{52}, r_{53}, r_{54}) \propto \left(\frac{1}{2}\right)(1)\left(\frac{1}{2}\right)(1)\left(\frac{1}{2}\right) = \frac{1}{8} = 0.125$

$P(r_{55} = -1 | r_{51}, r_{52}, r_{53}, r_{54}) \propto$
$P(r_{55} = -1) P(r_{51} = 1 | r_{55} = -1) P(r_{52} = -1 | r_{55} = -1) P(r_{53} = -1 | r_{55} = -1) P(r_{54} = 1 | r_{55} = -1)$

$P(r_{55} = -1) = \frac{2}{4} = \frac{1}{2}$

$P(r_{51} = 1 | r_{55} = -1) = \frac{1}{2}$

$P(r_{52} = -1 | r_{55} = -1) = \frac{1}{2}$

$P(r_{53} = -1 | r_{55} = -1) = \frac{1}{2}$

$P(r_{54} = 1 | r_{55} = -1) = \frac{2}{2} = 1$

$P(r_{55} = -1 | r_{51}, r_{52}, r_{53}, r_{54}) \propto \left(\frac{1}{2}\right)\left(\frac{1}{2}\right)\left(\frac{1}{2}\right)\left(\frac{1}{2}\right)(1) = \frac{1}{16} = 0.0625$

The rating r_{55} has a higher probability of taking on the value of 1.

(c)

- **Main idea**: exploiting the fact that **significant portion** of the **rows** (and **columns**) are **highly correlated**
- Use of **dimensionality reduction** for **estimating** the data matrix in **one-shot**
- Data matrix can be **approximated** by a **low-rank** matrix
- **Transformation** in which **pairwise-correlation** between dimensions are **removed**

Question 3. Content-Based Recommender Systems

From Aggarwal: Recommender Systems, Chapter 4.

(a) The Gini Index is used to assess a word's discriminative properties on a rating. Assume the following table that shows the presence of three words (i.e. three football players) in news stories that have been rated as Interesting or not interesting by the user:

News story	*"Messi"*	*"Ronaldo"*	*"Zlatan"*	*Rating*
1	Yes	Yes	Yes	Interesting
2			Yes	Interesting
3	Yes	Yes	Yes	Not interesting
4		Yes		Not interesting
5	Yes		Yes	Interesting
6		Yes	Yes	Not interesting
7	Yes		Yes	Interesting
8		Yes	Yes	Not interesting
9		Yes	Yes	Interesting
10	Yes			Interesting

Compute the *Gini index* for the three words. Which football player seems to the most interesting one to this user?

(b) We use nearest neighbor (k-NN) classification to recommend news stories to users. How do you think stop word removal, stemming or phrase extraction would affect the results from k-NN? Justify your answer. How can you reduce the computational complexity of k-NN?

Answers:

(a) Gini index of the word w is defined as follows:

$$Gini(w) = 1 - \sum_{i=1}^{t} p_i(w)^2$$

Where t is the total number of possible values of the rating (*Interesting* and *Not interesting*). $p_1(w), p_2(w), \ldots, p_t(w)$ is the fraction of the items rated at each of these t possible values.

Probabilities:

Rating	Zlatan	Ronaldo	Messi
Interesting	5/8 = 0.625	2/6 = 0.33	4/5 = 0.8
Not interesting	3/8 = 0.375	4/6 = 0.67	1/5 = 0.2

$Gini("Messi") = 1 - (0.8^2 + 0.2^2) = 1 - (0.64 + 0.04) = 0.32$

$Gini("Ronaldo") = 1 - (0.33^2 + 0.67^2) = 1 - (0.11 + 0.45) = 0.44$
$Gini("Zlatan") = 1 - (0.625^2 + 0.375^2) = 1 - (0.39 + 0.14) = 0.47$

Messi is the most useful person for discriminating between interesting and not interesting stories.

(b) Stop word removal, stemming and phrase extraction clean up the text, and we are left with better words for calculating similarities. We would expect that the similar documents are content-wise similar, and not just accidentally similar. Stop word removal makes documents more different. Lemmatization makes documents more similar. Phrase extraction makes them more different.

In general, stop word removal and lemmatization increase recall and reduce precision.

Stop word removal and lemmatization reduce the size of the dictionary. Hence, there is a reduction of computational complexity, and speed might increase slightly.

Reduce complexity of k-NN? Clustering

Question 4. Ensemble-Based and Hybrid Recommender Systems

From Aggarwal: Recommender Systems, Chapter 6.

Explain the main differences between the *feature combination* and the *feature augmentation* hybrids methods.

Answers:
(Main: Similarity with Stacking in data classification). The feature augmentation hybrid shares a number of intuitive similarities with the stacking ensemble in classification. In stacking, the first level classifier is used to create or augment a set of features for the second level classifier. In many cases, off-the-shelf systems are used like an ensemble. However, in some cases, changes may be required to the component recommender system to work with the modified data, and therefore the hybrid system is not a true ensemble of off-the-shelf systems
Two main cases:

- Content Based engine using item features generated from a collaborative filetering recsys (ex: The Libra System)
- Content Based first, filling the missing entries of a sparse rating matrix (pseudo-ratings). Then collaborative rec on the dense resulting matrix

Question 5. Evaluating Recommender Systems

From Aggarwal: Recommender Systems, Chapter 7.

Given the following dataset with book ratings from users and predicted ratings from our recommender system:

Nr	User ID	Book ID	User rating (r_i)	Predicted rating (p_i)
1	1	2-140	2	2.5
2	1	2-90	1	3
3	2	1-120	5	4
4	2	2-140	3	2.6
5	2	1-55	3	3.2
6	3	3-80	5	5
7	4	1-120	4	3.6
8	4	3-10	2	2.2
9	4	2-140	3	4
10	5	3-10	2	1.8

(a) Compute the *Mean Absolute Error (MAE)* of the predicted ratings.
(b) Make your own assumptions and compute a *precision* value for the predicted ratings. Explain the assumptions you make.
(c) Define *user-space coverage* and *item-space coverage*. Why is *catalog coverage* often a more useful measure than normal item-space coverage?

Answers:

(a)

$$MAE = \frac{|2.5-2|+|3-1|+|4-5|+|2.6-3|+|3.2-3|+|5-5|+|3.6-4|+|2.2-2|+|4-3|+|1.8-2|}{10} = 0.59$$

(b) Assume for example that 3 and above means that the document was relevant. That gives 6 relevant documents. Do the same on the predictions. That gives us 6 relevant documents as well (but somewhat different documents, overlap is 5 documents).

$$Precision = \frac{5}{6} = 0.83$$

(c) Catalog coverage is the fraction of items that are recommended to at least one user. Item-space coverage is the fraction of items for which the ratings of at least k users can be predicted. But we are rarely interested in generating recommended users for items.

Question 6. Time- and Location-Sensitive Recommender Systems

From Aggarwal: Recommender Systems, Chapter 9.

Explain the main idea behind the *recency-based models* in temporal collaborative filtering.

Answers:

(Main: Recent ratings are more important than older ratings) In recency-based models, recent ratings are given greater importance than older ones. The greater importance of recency can be addressed with either **decay-based** methods or with **window-based** methods. In decay-based methods, older ratings are given less importance with the use of a decay function. Window-based methods can be viewed as special cases of decay-based methods, in which binary decay functions are used to completely disregard data points that are older than a specific amount of time. In other words, the binary decay function ensures that older ratings are given a weight of 0, whereas recent ratings are given a weight of 1.

(Important that the students cite **decay-based** methods and **window-based** methods)

Question 7. Querying the Semantic Web

From Antoniou et al., A Semantic Web Primer, Chapter 2 & 3

You want to build and RDF model (RDF triples) of the 100 biggest mountains in Norway. For each mountain the model should specify the mountain's Norwegian name, how high it is, and in which municipality it is located. Some (but not all) mountains also have a Sami name.

(a) Explain how you would build this model with RDF triples and show an example of how a particular mountain will be described in RDF.

(b) Formulate a query in SPARQL that lists the names (Norwegian name and Sami name if available) of all mountains in the database.

(c) Formulate a query in SPARQL that lists all municipalities that have at least two mountains higher than 1,500 m.

Answers:

(a)

<mount1,name,MountainOne>
<mount1,high,8000>
<mount1, municipality, Trondheim>
<mount1, sameName, SameOne>

(b)

```
SELECT ?name ?same
WHERE {?mount, name, ?name.
            OPTIONAL {?mount, sameName, ?same}}
```

(c) Find two mountains in a municipality, filter in only those higher than 1.500m, and make sure that the two mountains are not the same.

Question 8. Personalized Information Access using Semantic Knowledge

From Plumbaum & Lommatzsch, Personalized Information Access using Semantic Knowledge

(a) The User Behaviour Ontology (UBO) describes all events relevant for modeling user behavior such as user clicks or mouse-over events. What are the two main goals that UBO serves?

(b) In the paper link prediction is used for the enrichment of user profiles. It aims to find important related items from a semantic dataset and to infer missing links in an observed graph that are likely to exist. What are the main advantages of enriching user profiles?

Answers:

(a)

- Sharing: Defining a common data model, an ontology, to manage user behavior information. Data can be shared and reused across systems.
- Enriched with external sources. Linking user behavior data with external knowledge following the Linked Open Data process. Collected behavior can be connected to other ontologies, adding extra knowledge.

(b)

- Cold start. To cope with the cold start problem.
- Collaborative filtering. to be able to use collaborative filtering.