Exam, November 15th, 2016. Erilliskoe 15.11.2016
582634 Data Mining

Hannu Toivonen
Department of Computer Science, University of Helsinki
Tietojenkäsittelytieteen laitos, Helsingin yliopisto

Answer all questions, on both sides of the sheet. Be concise in your answers. Write your name and student number on each sheet that you return.
Finnish translations are written in italics.

Vastaa kaikkiin kysymyksiin tämän paperin molemmilla puolilla. Vastaa tiiviisti. Kirjoita nimesi ja opiskelijanumerosi jokaiseen vastauspaperiin. Voit vastata vapaasti suomeksi tai englanniksi. Jos vastaat suomeksi, käytä samoja käännöksiä kuin kysymyksissä tai anna väärinkäsitysten välttämiseksi myös englanninkielinen termi. Opintorekisteriin suorituksen kieleksi merkitään vastausten enemmistön kieli.

(If you did not yet register for the exam, do it ASAP after the exam so that your participation can be registered!) *(Jos et ole vielä ilmoittautunut tenttiin, tee se välittömästi jotta suorituksesi voidaan rekisteröidä!)*

1. (9 points) Consider the following six transactions:

   `ABDEFI, BCDEG, ABEFH, ACDH, ABDEF, BCEGI`

   For convenience, letters represent items and strings represent sets of items. I.e., `ABD` stands for the set `{A, B, D}`.

   Let the support threshold be 0.4. Simulate Apriori algorithm (in the course book, the "$F(k-1) \times F(k-1)$" variant). For each iteration of the main algorithm, list the generated candidates, indicate which of the candidates are pruned by the Apriori principle, and list the frequent itemsets and their support counts.

   Tarkastellaan seuraavia kuutta tapahtumaa:

   `ABDEFI, BCDEG, ABEFH, ACDH, ABDEF, BCEGI`

   Yksinkertaisuuden vuoksi alkioita merkitään kirjaimilla ja joukkoja merkkijonoina. Esim. `ABD` *tarkoittaa joukkoa* `{A, B, D}`.

   Olkoon tuen kynnysarvo 0,4. Simuloi Apriori-algoritmia (tekstikirjassa versio "$F(k-1) \times F(k-1)$"): anna pääohjelman kullakin iteraatiolla generoidut kandidaattijoukot, merkitse niistä Apriori-periaatteella karsittavat joukot, ja luettele kaikki toistuvat joukot sekä niiden tuet.

2. (6 points) Consider a transaction data set which is not shown here, but all frequent closed itemsets are listed below with their support counts. The support count threshold is 100.

   `AB`: 231
   `B`: 283
   `BC`: 144
   `BDE`: 167

   (For convenience, letters represent items and strings represent sets of items. I.e., `ABD` stands for the set `{A, B, D}`.)

   What are the support counts of the following itemsets:

   `A`, `B`, `AD`, `BD`

   (In Finnish on the other side of the sheet.)

(Question 2 in Finnish:)

Tarkastellaan ostoskoriaineistoa jota ei anneta tässä, mutta jonka kaikki kattavat suljetut joukot ja niiden tuet annetaan ohessa. Tuen kynnysarvo on 100.

`AB`*: 231*
`B`*: 283*
`BC`*: 144*
`BDE`*: 167*

(Yksinkertaisuuden vuoksi alkioita merkitään kirjaimilla ja joukkoja merkkijonoilla. Esimerkki: `ABD` *tarkoittaa joukkoa* `{A, B, D}`*.)*

Mitkä ovat seuraavien joukkojen tuet:

`A`, `B`, `AD`, `BD`

3. (6 points) Define the concepts of confidence and lift, and illustrate them and their properties with examples.

   Määrittele luottamuksen ja nosteen käsitteet ja havainnollista niitä ja niiden ominaisuuksia esimerkein.

4. (9 points) Consider the following frequent pattern mining task. The data consists of a number of strings, and the task is to discover frequent (contiguous) substrings. For instance, in strings `computer` and `camp`, strings `c` and `mp` are maximal patterns with frequency two.

   (a) What are the essential concepts to be specified in a frequent pattern discovery task? Formulate them for this task in an exact form.

   (b) Draw the complete search space for the case where the data consists of strings `computer`, `camp`, `compare`. Mark patterns that have a frequency of at least two.

   (c) Outline in pseudocode an efficient candidate generation algorithm for an Apriori type of method. (No need to give Apriori here, just the candidate generation step.)

   Tarkastellaan seuraavaa toistuvien hahmojen etsimisongelmaa. Aineisto koostuu useista merkkijonoista, ja tehtävänä on tuottaa kattavat (yhtenäiset) osamerkkijonot. Esimerkki: merkkijonoissa `computer` *ja* `camp` *jonot* `c` *ja* `mp` *ovat maksimaaliset hahmot joiden tuki on kaksi.*

   (a) Millaiset käsitteet pitää määritellä toistuvien hahmojen etsintää varten? Muotoile ne täsmällisesti tätä ongelmaa varten.

   (b) Piirrä hakuavaruus, kun aineisto muodostuu merkkijonoista `computer`*,* `camp`*,* `compare`*. Merkitse piirrokseen hahmot, joiden tuki on vähintään kaksi.*

   (c) Hahmottele tehokas Apriori-tyyppinen kandidaattien tuottamisalgoritmi pseudokoodina. (Apriori-algoritmia ei tarvitse antaa, vain kandidaattien generointivaihe.)