Information Retrieval and Data Mining (IRDM) SS 2015
Prof. Dr.-Ing. Sebastian Michel
MSc. Koninika Pal
TU Kaiserslautern, FB Informatik – Lehrgebiet Informationssysteme
Sheet 2: Handout 06.05.2015, Presentation 19.05.2015
TECHNISCHE UNIVERSITÄT KAISERSLAUTERN
http://dbis.informatik.uni-kl.de

Assignment 1: Retrieval Effectiveness and Ranking Model (1 P.)

(a) Suppose we have a set of 10,000 documents. 64 of the documents are relevant for our query. We evaluate three different IR methods with this query, obtaining the following results:
Method I retrieves 0 documents.
Method II retrieves 183 documents of which 61 are relevant.
Method III retrieves 84 documents of which 32 are relevant.

(i) Compute precision, recall, accuracy, and F_1 for the methods.

(ii) What can you say about the three methods? Is one of them better or worse than the others as an IR method?

(b) Consider a document collection given by following table. The numbers in the table denote the term frequencies for each document.

	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
d_1	0	1	2	1	1	3	1	1	4	0
d_2	0	0	1	0	1	0	1	1	1	0
d_3	0	0	1	1	0	0	1	0	1	0
d_4	0	2	3	3	1	2	1	0	1	0
d_5	0	0	0	0	1	1	1	0	1	3
d_6	0	0	0	0	1	1	0	0	1	0
d_7	0	0	2	0	1	1	0	0	1	0
d_8	0	0	1	0	1	1	0	0	0	0
d_9	0	3	1	1	3	0	0	1	1	0
d_{10}	1	0	0	0	0	0	0	0	1	1

(i) Rank the documents according to the query $q = \langle t_1, t_3 t_5, t_6 \rangle$ using the following scoring models.

- $score(q,d) = \sum_{t \in q} tf.idf(t,d)$, where $tf.idf(t,d) = tf(t,d) \times \frac{|D|}{df(t)}$
- $score(q,d) = \sum_{t \in q} tf.idf(t,d)$, where $tf.idf(t,d) = tf(t,d) \times log\frac{|D|}{df_t}$

(ii) Compare the rankings returned by these two scoring models and explain why the rankings differ?

Assignment 2: Probability Theory Warm-Up (1 P.)

(a) Calculate the answers to the following questions. Hint: You should restrict yourself to the specific probability distributions mentioned in the lecture slides and investigate their applicability in terms of sufficient parameters given in the statements.

(i) A dice is rolled 10 times. What is the probability that the face "4" turns up for 3 times?

(ii) How many times do we need to roll a dice until a "4" or a "2" turns up?

(iii) Suppose we have a document collection where the term "computer" occurs 5 times per document on average. What is the probability that the term "computer" occurs exactly 8 times in a document?

(b) Consider the following joint distribution of Boolean random variables X_1, X_2, and X_3:

X_1	X_2	X_3	Pr
0	0	0	0.04
0	0	1	0.08
0	1	0	0.08
0	1	1	0.20
1	0	0	0.06
1	0	1	0.12
1	1	0	0.12
1	1	1	0.30

(i) What are the marginal distributions of X_1, and X_2?

(ii) What is the conditional distribution $\Pr(X_1 \mid X_2)$?

(iii) What is the variance and the expectation of X_2?

Assignment 3: BIM & Tree Dependence model (1 P.)

(a) Consider the query { *q:=Michael Jordan computer science"*} with the four terms $t_1 = Michael, t_2 = Jordan, t_3 = computer, t_4 = science$. An initial query evaluation returns the documents $d_1, \ldots, d_{10}$ that are intellectually evaluated by a human user. The occurrences of the terms $t_1, \ldots, t_4$ in the documents as well as the relevance feedback of the user are depicted in the following table, where "1" points out a relevant document and "0" points out a non-relevant document.

	t_1	t_2	t_3	t_4	**Relevant**
d_1	**1**	0	**1**	0	0
d_2	**1**	**1**	0	0	0
d_3	**1**	0	0	0	0
d_4	0	**1**	**1**	**1**	**1**
d_5	**1**	**1**	**1**	**1**	**1**
d_6	0	**1**	0	**1**	**1**
d_7	0	**1**	**1**	0	0
d_8	**1**	0	**1**	**1**	**1**
d_9	**1**	**1**	0	0	0
d_{10}	**1**	**1**	0	0	0

Compute the similarities of the documents $d_{11} and d_{12}$ to the given query using the probabilistic retrieval model with relevance feedback according to the formula by *Robertson & Spärck-Jones* with Lidstone smoothing ($\lambda = 0.5$).

	t_1	t_2	t_3	t_4
d_{11}	**1**	0	**1**	0
d_{12}	0	**1**	**1**	**1**

(b) Suppose we have $n = 4$ documents $d_1 = (001)$, $d_2 = (110)$, $d_3 = (101)$, $d_4 = (111)$ composed of terms $T = \{t_1, t_2, t_3\}$ which are denoted by binary RVs $X_1, ...X_3$. Compute a tree dependence model for binary term correlations following the algorithm provided in the lecture slides, i.e., using the following steps:

(i) Construct the joint probability density function f_{X_1,X_2,X_3} and use this joint distribution to compute the (true) pairwise marginals f_{X_1,X_2}, f_{X_1,X_3}, and f_{X_2,X_3}.

(ii) Compute the marginal probability density functions $g_{X_1}, ...g_{X_3}$ from the joint distribution f_{X_1,X_2,X_3}

(iii) Reconstruct the pairwise joint probability distributions g_{X_1,X_2}, g_{X_1,X_3},... from $g_{X_1}, .., g_{X_3}$ by assuming independence among the g_{X_i}.

(iv) Measure the Kullback-Leibler divergence between matching pairs of f_{X_i,X_j} and f_{X_i,X_j} and assign the corresponding error weights to the all three edges in the complete term graph $G = (T, T \times T)$.

(v) Construct the maximum spanning tree (i.e., the term dependence tree) over T.