MOSIG – MSIAM – 2017-2018 Information Access and Retrieval – GBX9MO23
Georges Quénot – Philippe Mulhem – Jean-Pierre Chevallet
28 January 2018 – 13h00-15h00 (1:00pm-3:00pm) – 2 hours

Course materials, the two papers related to the examinations, personal notes, and calculators (without network capabilities) are allowed.

The examination consists in questions related to three scientific papers and/or to the contents of the course:

[1] “BitFunnel: Revisiting Signatures for Search” from Bob Goodwin, Michael Hopcroft, Dan Luu, Alex Clemmer, Mihaela Curmei, Sameh Elnikety and Yuxiong He, in the Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval.

[2] “Deep Image Retrieval: Learning global representations for image search” from Albert Gordo, Jon Almazan, Jerome Revaud, Diane Larlus, ECCV 2016.

Please use separate examination sheets for questions related to paper [1] and questions related to paper [2].

You should spend about 5 minutes per question and we expect concise answers.

Questions related to paper 1.

Q1.1: Data Base vs Information Retrieval

List and explain the main differences between an Information Retrieval System and a Database. Explain the main goal of both systems; the way user interacts with them using queries. Detail the typical data structures that are used in both systems. Explain the role of index for both systems, and give examples of data structure for indexes in both systems.

Q1.2: Vector Space Model

Recall the basic notion of the Vector Space Model in IR domain. Explain the meaning of the dimension value (i.e.: weight) when they are positive, null, or negative.

Q1.3: General Matching Problem

The paper mentions in part 2 (Background and prior work) the notion of "Matching problem". Explain how this "Matching problem" can be related (or not) to the Vector Space Model. NB: to simplify, we suppose the matching function to be simply the dot product, that there is no negative weight, and that a non- null weight for a term in document or query index means that it belongs to the document or query set.

Q1.4: Inverted Files vs Signature

Compare the notion of **inverted files** (with posting list) and the notion of **signature** (the simple notion of signature without the notion of Bloom filter) for a textual Information Retrieval System. Compare mainly the role of these two data structures, the algorithms used to query these structures, and the data stored in the structure.

Q1.5: Signature usage in IRS

Explain why, in most IR system, signatures are not used.

Q1.6: Definition of a Bloom Filter

Recall the main characteristics of a Bloom filter. Give a small example on constructing such a filter with 5 terms {a,b,c,d,e,f}, a signature length of 4 bits, 2 documents d1= {a,b), and d2 = {b,c}, and 2 hashing functions (i.e.: term signatures have only 2 bits at value 1).

Q1.7: Bloom Filter to index documents

Given a Bloom filter on 32 bits to index documents with an average size of m terms, from a corpus of n documents: propose an algorithm to solve a query having only one term {t} and give the complexity of this algorithm. Compare with the complexity when using an inverted file structure. Which structure is more efficient in this case?

NB: we do not take into account the term frequency or document frequency.

Q1.8: Using a Bloom Filter to query an index

Consider some hash function, a signature length of 8 bits, and a set of 8 documents whom signature are:

d1: 0100 1101
d2: 1110 0110
d3: 1001 0101
d4: 0001 0101
d5: 11 01 1011
d6: 11 10 0111
d7: 0100 0101
d8: 1010 1010

Computes the list of documents that answer the query q = {t1} where the signature of term t1 is = 1000 0001.

Q1.9: Bit-sliced signature

Explain the notion of bit-sliced signature and the interest against normal signature. Explain the bit-sliced algorithm given in the paper. How large is the bit-sliced document signature in the example of the previous question (Q8)?

Q1.10: Bit-sliced signature efficiency

Suppose we have a corpus of 'n' documents, a signature of size 's', and 'h' hashing functions. When using a computer with 64 bits long integer, then Boolean operations are done with all 64 bits at once, and a binary string (i.e. binary vector) computation is done using 64 bits slices of the string. We consider that all basic Boolean operations on 64 bits to take the same time 't' to compute. Propose a formula expressing the time taken to solve a query with 1) simple signature and with2) bit-sliced signature. Justify your formulas. Note: we do not care about all other factors like data structure to access the data, cost of loops, etc.

Finally, using these formulas, justify why and when Bit-sliced signature is more time efficient, and estimate the ratio of theoretical speed gain.

Q1.11:

Consider again the 8 documents in question (Q8).

The dot product considers each signature as a binary vector. The following tables is the dot product of all 8 documents signatures:

	1	2	3	4	5	6	7	8
1	4	2	2	2	3	3	3	1
2	2	5	2	1	3	5	2	3
3	2	2	4	3	3	3	2	1
4	2	1	3	3	2	2	2	0
5	3	3	3	2	6	4	2	3
6	3	5	3	2	4	6	3	3
7	3	2	2	2	2	3	3	0
8	1	3	1	0	3	3	0	4

The Hamming distance is the number of bits that are different. The following table is all the Hamming distances between document signatures:

	1	2	3	4	5	6	7	8
1	0	5	4	3	4	4	1	6
2	5	0	5	6	5	1	4	3
3	4	5	0	1	4	4	3	6
4	3	6	1	0	5	5	2	7
5	4	5	4	5	0	4	5	4
6	4	1	4	5	4	0	3	4
7	1	4	3	2	5	3	0	7
8	6	3	6	7	4	4	7	0

Use theses information to propose a blocked signature with a blocking factor of 2 that improve performance but minimize the noise.

Q1.12: Bit-slices vs signature

Explain the differences in the algorithm enhancement of signature based search using bit-sliced Signatures and the algorithm of inverted file matching when considering that, query Q could be viewed as composed 3 "sub-terms" instead of 3 bits positions by the hashing function. See the example of figure 1, where bit 2, 5 and 9 can be viewed as 3 terms composing the query Q. Give an example how to represent the 3 slices to run the algorithm.

Q1.13: Evaluation

Recall what a MAP measure is, then explain briefly the way this work was evaluated.

Questions related to paper 2.

Q2.1: Conventional Retrieval

List the classical image retrieval techniques described in the paper.

Q2.2: Transfer Learning

What is transfer learning and how is it used in this work?

Q2.3: Siamese Network

What is the main goal of Siamese networks?

Q2.4: Three-branch networks

Why are three-branch Siamese networks better than two-branch ones?

Q2.5: Loss Function

Explain the choice of the loss function given in equation (1). What is the role of the m parameter?

Q2.6: Revisited R-MAC

On which aspects has the original R-MAC method been modified? Why is it important that all operations in the revisited R-MAC method are differentiable?

Q2.6: Region Proposal Network

What is the role of the Region Proposal Network? Does it bring a significant improvement? What is necessary for its training?

Q2.7: PCA

What are PCA and whitening? Is it an actual PCA which is implemented in the revisited version?

Q2.8: Oxford 5k versus Oxford 105k tasks

What is the difference between the Oxford 5k and Oxford 105k tasks? Is the latter significantly harder than the former?

Q2.9: VGG-16 network

How many learnable parameters are there in:

a) the sixth convolutional layer (56×56×256 → 56×56×256)?
b) the second pooling layer (112×112×128 → 56×56×128)?
c) the last fully connected layer (4096 → 1000)?
d) the softmax layer (1000 → 1000)?

Which of these depend upon the input image size? Reminder: all convolutions are 3×3.

Q2.10: Query Expansion

What is query expansion and how is it implemented? Does it bring significant performance improvements? Can it be used with all methods?

Q2.11: Progress over time on the Oxford 105k task

According to the results displayed in table 3 and using the dates of the reference, build a table indicating year by year the progress on the Oxford 105k instance retrieval tasks, separately with and without Query Expansion. Indicate the main innovations that enabled the performance improvements.