Department of Computer and Information Science and Engineering

University of Florida

CAP4770/CAP5771 Fall 2015

Midterm I

Instructor: Dr. Daisy Zhe Wang

This is a in-class, closed-book exam.

This exam contains 7 single-sided sheets of paper (excluding this one).

Write all answers on these pages, preferably on the white space in the problem statement. Continue on the draft pages if running out of space but **clearly number** your answers if doing so.

Make sure you attack every problem; partial credit will be awarded for incomplete or partially correct results.

THIS IS A CLOSED BOOK, CLOSED NOTES EXAM.

Name:

UFID:

For grading use only:

Question:	**I**	**II**	**III**	**Total**
Points:	**30**	**30**	**40**	**100**
Score:				

I. [**30 points**] **Information Extraction and Natural Language Processing.**
Given the following text from the reference section of a research paper:

C. Bizer, ”Web of linked data - a global public data space on the Web,” in Proc. 13th Int. Workshop on the World Wide Web and Databases, 2010.
K. Wilkinson, ”Jena property table implementation,” HP-Labs, Tech. Rep. HPL-2006-140, 2006.
Y. Halevy, ”Answering queries using views: A survey,” VLDB J., vol. 10, no. 4, pp. 270294, 2001.
R. Ullmann, ”An algorithm for subgraph isomorphism,” J. ACM, vol. 23, no. 1, pp. 3142, 1976.

(**1**) [**15 points**] Write regular expressions to extract author, paper title and year of publication fields. Also write out the extractions in CSV format, by applying those regular expressions over the reference section above.

Solution: Answer must include something similar to this regular explression

```
$ sed 's/\([A-Z]\.* [A-Z][a-z]*\), "\(.*\)," .*,
    \([0-9][0-9][0-9][0-9]\)\./\1,\2,\3/'
```

(**2**) [**15 points**] Consider the task of extracting, from blogs, informal reviews of live performances by music bands. One of the subtasks is the extraction of musicians and their instruments to populate a relation: *MusicianPlaysInstrument (Musician string, Instrument string)*. Example text:

John Pipe plays the guitar.
Marco Benevento on the Hammond organ.

Please describe the possible techniques that can be used to develop such an extractor. What optimizations can you think of if the corpus is really large (> 1TB)?

Solution: A good solution inclueds the use of: regular expressions, POS tagging and a dictionary of music instruments. The pipeline: create a regular expression to capture band names, and keep a dictionary that matches instruments (guitar, piano, etc) then parse sentences using parsing trees and POS taggs to determine sentences where a music group tagged with its POS and match the RE, and an instrument can be found in the same sentense. Then from sentence extract elements in the relation.

A large scale slution to the problem uses System T for this information retrieval task.

II. [30 points] Data Modeling and Similarity Metrics.

In this problem, we will explore the use of different scoring and weighting schemes with vector space model on information retrieval tasks. Consider a simple collection with the following two documents, query and stop words list:

Document 1: the way to the school is long and hard when walking in the rain
Document 2: the rain has not stopped in days and the school has closed
Query: school closed rain
Stop Words: the, to, is, and, in, has, not

(1) [**15 points**] For each document and query, please write their *set* representation in a bag-of-words model. What are the similarity scores of the query with each document given above using Jaccard coefficient with stop words included? What are the similarity scores excluding stop words? What are the pros and cons of using stop words?

Solution:

Set Representation
Doc1 = the, way, to, school, is, long, and, hard, when, walking, in, rain
Doc2 = the, rain, has, not, stopped, in, days, and, school, closed
Query = school, closed, rain

	Doc1	Doc2
using stopwords list	2/13 = 0.1538	3/10 = 0.3
without using stopwords	2/8 = 0.25	3/5 = 0.6

(2) [**15 points**] For each document and query, please write their *vector* representation using bag-of-words model with term frequency (TF). What are the similarity scores of the query with each document given above using cosine similarity over the vector model? What is TF-IDF and why it is usually used in search engines to compute ranking instead of TF?

Solution:

vector representation (Vector can be in any order as long as both are in same order) I took off stop words, answers that did not remove stop words also get full marks.

id	word	doc1	doc2	query
0	walking	1	0	0
1	closed	0	1	1
2	hard	1	0	0
3	when	1	0	0
4	school	1	1	1
5	long	1	0	0
6	days	0	1	0
7	stopped	0	1	0
8	rain	1	1	1
9	way	1	0	0

	doc1	doc2
without stop words	$\frac{2}{\sqrt{7}\sqrt{3}}$	$\frac{3}{\sqrt{5}\sqrt{3}}$

tfidf, short for term frequency inverse document frequency, is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus. It is often used as a weighting factor in information retrieval and text mining. The tf-idf value increases proportionally to the number of times a word appears in the document, but is offset by the frequency of the word in the corpus.

III. [40 points] Map Reduce.

The difference of two sets $(B - A)$ that share the same schema is defined as all the elements that are in B but not in A. Formally,

$$B - A = \{x : x \in B \land x \notin A\}$$

Examples

$$\{1, 2, 3\} - \{2, 3, 4\} = \{1\}$$
$$\{(1, 2), (1, 3), (2, 4)\} - \{(1, 2), (2, 4), (3, 3)\} = \{(1, 3)\}$$

If we have two sets/tables A and B that share the same schema (a,b,c). Please define Map and Reduce functions to calculate $B - A$ that can be executed in parallel. Assume that the two input tables are concatenated in a single file that will be processed.

(1) **[20 points] Map Function:** Each input line is a list of 4 elements. The first element has two possible values A or B. The remaining 3 elements corresponds to the 3 attributes in the identical schema of A and B. Please write the output of the mapper given the example input and pseudocode for the map function.

Input example:
A, 1,2,3
A, 1,2,5
B, 1,2,3
B, 2,2,2

Solution:
The Map Function: For a tuple t in A, produce key-value pair (t,A), and for a tuple t in B, produce key-value pair (t,B). Note that the intent is that the value is the name of A or B (or better, a single bit indicating whether the relation is A or B), not the entire relation.

```
map(tuple)
  #Tuple have 4 elements
  emit_intermediate(tuple[1:], tuple[0])
```

Sample Output:
$< 1, 2, 3 >$, "A"
$< 1, 2, 5 >$, "A"
$< 1, 2, 3 >$, "B"
$< 2, 2, 2 >$, "B"

(**2**) [**20 points**] **Reduce Function:** The output of reduce function should be lines with 3 comma separated elements as shown below. Write the input to the reduce function according to the given the example and the pseudocode of the reduce function to compute set difference.

Output Example:
2,2,2

Solution:

The Reduce Function: For each key t, if the associated value list is [B], then produce (t, t). Otherwise, produce nothing.

Sample input:
$< 1, 2, 3 >$, [”A”,”B”]
$< 1, 2, 5 >$, [”A”]
$< 2, 2, 2 >$, [”B”]

```
reduce(key, list_of_values)
  if list_of_values == ["B"]:
    emit(t,t)
```

(*This page is intentionally left blank*)