Quiz 1

Q1: Write one idea to improve user experience in Google search, without changing the ranking.

S1:

1. Clustering of Results (Click one of the clusters and get results without query)

    Cluster 1: Lincoln president

    Cluster 2: Lincoln cars

2. Vertical Search(Detect the topic of the query and based on it shows videos, weather sports, text, FAQ's etc and use It on top of returned results to give personalized exp)

3. Popup on mouseover giving more information

Quiz 2

Q1: Describe an example - 2 documents and a query - where vector space model ranking fails. Only consider text and no formatting.

S1:

Examples portraying the following points

1. Assumption of term independence: Vector space model completely ignores context and just matches weights for ranking documents. Here, D1 matches more with the query regarding but according to context D2 is the right answer.

   D1:The temple is a peaceful place to relax there is also a bus stop to sit.

   D2:There is bus stop near temple within a mile.

   Q: What's the distance from bus stop to temple to peacefully catch a bus.

2. No synonyms/stemming handling: Even though D2 should clearly be more relevant, this method does not place more weight on the word "pool"

   D1: Thomas plays soccer

   D2: Thomas is playing pool

   Q: plays pool

3. Ignores order and distance between terms: Vector space model uses the cos formula to measure the distance. Length of the vector is not good because if the doc has more words it can increase the size of vector and angle increases too.

   D1: UCR baseball

   D2: UCR Riverside

Q: UCR Riverside

Q2: Give an example of 2 documents and a query to show that unigram LM does not account for dependencies between query keywords.

S2:

1. Dependence(Considering Riverside and Soccer equally popular in Unigram Model (LM) D1 and D2 will be assigned same score but actually D2 should have been ranked higher)

   Q: UCR Riverside Soccer

   D1: UCR Riverside

   D2: UCR Soccer

2. Synonyms (Both D1 and D2 again will be ranked similar in Unigram but since play and show are similar verbs D1 should have been ranked higher)

   Q: cats play

   D1: cats show

   D2: cats Riverside

Quiz 3

Q1:How would you crawl Twitter, that is, retrieve as many tweets as possible?

Assume Twitter offers the following APIs:

- getTweets(userid) -> tweets list

- getFollowers(userid) -> users list

S1:

```
def crawlTwitter():
    q=deque()
    seed=Seed(users)
    q.append(seed)
    tweets=[]
    While q:
        user=q.dequeue() ## popping from head
        q.enqueue(getFollowers(user))
        tweets.append(getTweets(user))
```

Quiz 4

Q1: Write high level MapReduce pseudocode to compute the document frequencies (df) of all words in a collection. What is the format of the {key,value} pairs output by the mappers?For example, if collection contains documents:D1: UCR school, D2: UCR play UCR the output should be, UCR, 2, school,1, play, 1

S1:

```
def MapWord(docid, doc):
  while doc:
     words = document.split(“ ”)
     tokens = Parse(words)
     for w in tokens:
       emit(word,1)
 def ReduceWord(key,values):
  sum = 0
  For all v in values:
          sum+=v
  emit(key, sum)
```

Quiz 5:

Q2: Consider Twitter users u1, u2, u3.

u1 follows u2

u1 follows u3

u2 follows u3

u3 follows u1

Use PageRank to compute the influencer score of each user. Show the calculations of the first iteration.

S2:

Assuming d=0.85

Iteration 0 ->PR(u1)=PR(u2)=PR(u3)=1

Iteration 1->

PR(u1) = (0.15 / 3) + 0.85 * (PR(u3)/1) = 0.15/3 + 0.85 * 1/1 = 0.9

PR(u2) = 0.15 / 3 + 0.85 * (PR(u1) / 2) = 0.475

PR(u3) = 0.15 / 3 + 0.85 * (PR (u1) / 2 + PR(u2) / 1) = 1.325

Assuming d = 0.5

Iteration 0 ->PR(u1)=PR(u2)=PR(u3)=1

Iteration 1->

PR(u1)=0.5 + 0.167 = 0.667

PR(u2)=0.25 + 0.167 = 0.417

PR(u3)=0.75 + 0.167 = 0.917

Q3: Which user will have the highest influencer score (highest PageRank)?

S3: Highest PR user is u3

Q4: Now imagine you want to compute the influencer scores personalized for u2. How would you modify the PageRank formulas? Show the first iteration's equations.

S4:

Base Set={u2}

Assuming d=0.5.

Iteration 1:

pr(u1) = .5+0 = .5

pr(u2) = .25 + .5/(1) = .75

pr(u3) = .75 +0 = .75

Quiz 6

Q1: Consider query q, which has 2 relevant documents in the collection.
Let a ranking algorithm generate return 4 results:
rxxx, where r means "relevant" and x means "non-relevant".
What is the precision of this result?
S1:

- ¼

Q2: What is the recall?

S2:

- ½

Q3: What is the Average Precision of the ranking?
S3:

- ½