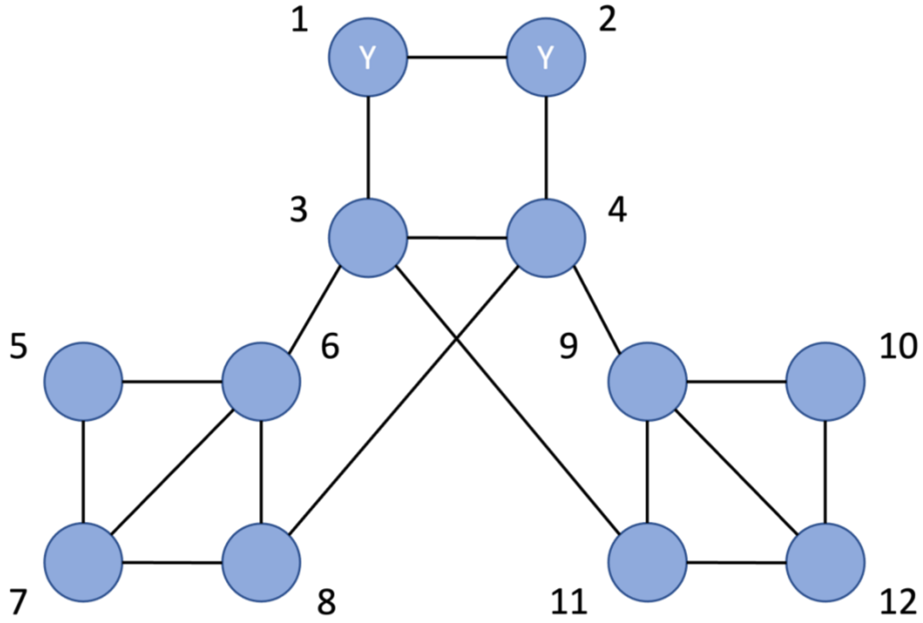
Solution: Big Data Analysis (Practice Final Exam)

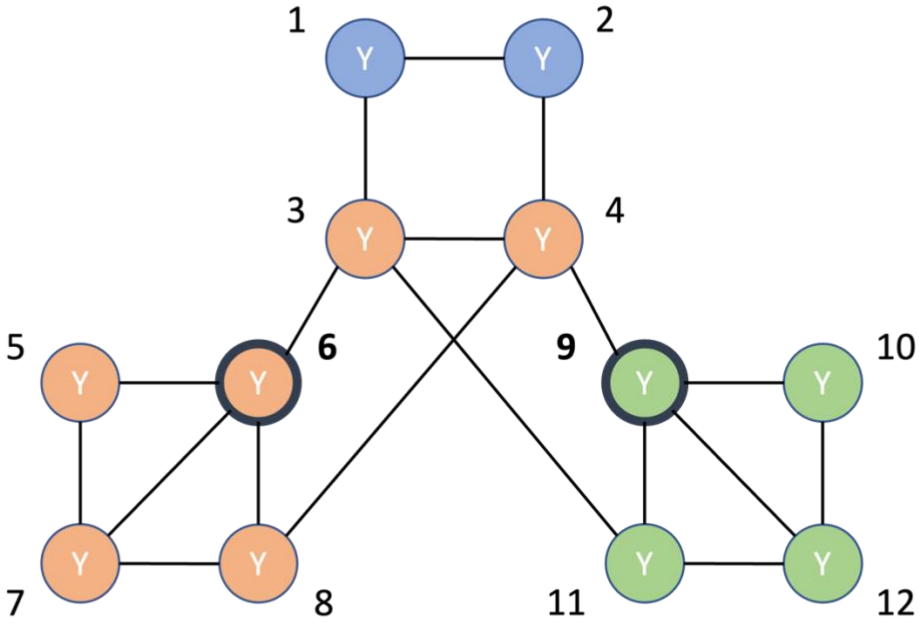
Question 1:



a)

In this problem, there are two outcomes, either the majority voting “Yes” (Y) or the majority voting “No”. For the first day of voting, we iterate through each note by the increasing number of ID, node one already voted (Y); therefore, we start from node 2. Node 2 has two neighbors (node 1 and node 4). Since node 1 already voted Y, >=50% of the neighbors of node 2 have voted yes, therefore, node 2 will vote Y. Then, we come to node 3, node 3 has four neighbors (node 1, 4, 11, 6), but only node 1 voted Y and other nodes have not voted. Therefore, node 3 remains not voted. Similar to the other following nodes. Up to the final day, the process will happen exactly the same (with only node 1 and 2 voted in the first day, and new note voted Y), therefore, other nodes will vote N. The result we have is that the majority voting N (10 nodes – the remaining nodes).

b)



The minimum number of voters to invite is 2. We can invite node 6, and node 9.

There are many possibilities, but we can focus on the reason why the outcome of **a)** have the majority voting N. As seen on the answer above, the vote cannot be spread to node 3, since the majority of node 3's neighbors are undecided. Intuitively, we want to spread the vote Y to node 3 as well to continue the process of spreading the news. Therefore, we can choose any of node 3's neighbors to spread. Here, we need to choose wisely, so that it can spread the news as much as possible. The most intuitive way to choose a node is to pick the one with most connections (high degree centrality). Here, we choose node 6 as it has most connection. From there, if we try to spread the vote Y to other nodes, we can see that it will stop at node 9, which is symmetrical to node 6. Here, we choose node 9 as our guest to the dinner table, so that node 9 votes Y. From there, other nodes will vote Y too, due to the influence of node 9.

c)

There is no existing solution.

In this question, we have the option to create any connection between any nodes to make all the voters in the graph vote Y. However, although it sounds possible to make all nodes vote Y by connecting nodes, it also heavily depends on the topology of the graph as well. Intuitively, we only have three nodes to consider connecting to other nodes, which are node 1, 2 and 3. The reason is that node 1 and 2 surely voted Y, so we may be able to make something out of it. Node 3 is the node that stops the spread of influence voting Y, as we know from **a)**. Other nodes are undecided, so they need the influence from the three nodes above. Let say we connect either node 1 or 2 to any node, we can notice that we still cannot surpass the 50% of that node's neighbors voting Y **(1)**. For example, connecting node 2 to 3, now node 3 has five neighbors (originally four), two nodes (2/5=40%) voting Y do not pass the condition to vote Y for node 3. Similarly to other nodes. Therefore, logically, we need to choose the node that has two neighbors to balance the vote, so that we can connect two nodes already voting Y (which are node 1 and 2). In the network, there are two nodes meeting that condition, which are node 5 and node 10. However, we cannot reach those nodes, since the voting processed in an increasing order of node ID **(2)**.

From **(1)** and **(2)**, there is no solution existed.

Question 2 (Lecture 5-6):

a) How do you model the problem (input, output, etc.)? Justify your model.

Input: A schema repository D={S1,S2,...} and a query schema Q

Output: a ranking score f: D-->[0,1]

Model: TF-IDF. We consider each attribute as a term. As some attributes are common among schemas, we need to not consider those attributes and focus more on the other distinct attributes among schemas.

b) What steps should be involved? Provide a quantitative measure for each step if needed. Justify the design choice for each step.

Step 1. Build a term vocabulary of the current schema repository.

--> Establish the table of terms, so that we can perform calculations in the following steps.

Step 2. Compute term frequency and normalize term frequency by tf-idf weights.

--> Calculate the term frequency (tf), then inverse document frequency (idf), so that we can calculate the importance of the terms among schemas by using tfidf.

Step 3. Transform all schemas and the query into a vector of term frequencies.

--> Schemas and query are now illustrated as vectors, which is to use for the next step.

Step 4. Compute the similarity between the query and each schema by cosine similarity.

--> Now we compute the similarity between the query and each schema.

c) Apply your approach to the above example and calculate the quantitative results.

Term vocabulary: {a1, a2, a3, a4, a5, a6, a7, a8, a9, a10}

Document-term matrix:

	S1	S2	S3	S4
a1	1	1	0	1

a2	0	0	1	0
a3	1	0	0	0
a4	0	1	0	0
a5	0	0	0	1
a6	0	0	1	0
a7	1	0	0	0
a8	0	1	0	0
a9	0	0	1	0
a10	0	0	0	1

TF-IDF normalization:

idf(a1,D) = ln(4/3) ≈ 0.29

idf(a2,D) = idf(a3,D) = idf(a4,D) = idf(a5,D) = idf(a6,D) = idf(a7,D) = idf(a8,D) = idf(a9,D) = idf(a10,D) = ln(4/1) ≈ 1.39

S1 = [0.29, 0, 1.39, 0, 0, 0, 1.39, 0, 0, 0]

S2 = [0.29, 0, 0, 1.39, 0, 0, 0, 1.39, 0, 0]

S3 = [0, 1.39, 0, 0, 0, 1.39, 0, 0, 1.39, 0]

S4 = [0.29, 0, 0, 0, 1.39, 0, 0, 0, 0, 1.39]

Q = [0.29, 1.39, 0, 0, 0, 0, 0, 0, 0, 0]

Cosine similarity:

Sim(Q,S1) = (0.29*0.29)/(sqrt(0.29^2 + 1.39^2 + 1.39^2) * sqrt(0.29^2 + 1.39^2)) ≈ 0.029

Sim(Q,S2) = (0.29*0.29)/(sqrt(0.29^2 + 1.39^2 + 1.39^2) * sqrt(0.29^2 + 1.39^2)) ≈ 0.029

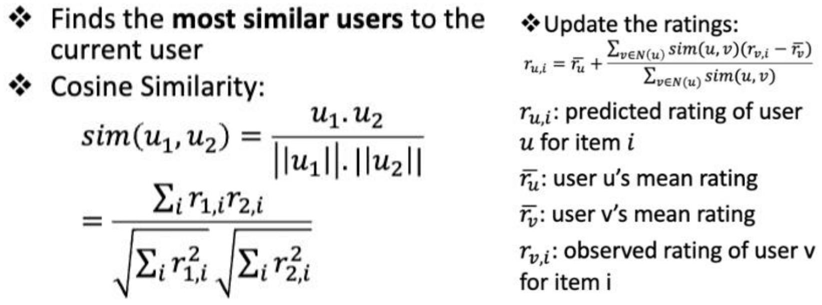
Sim(Q,S3) = (1.39*1.39)/(sqrt(1.39^2 + 1.39^2 + 1.39^2) * sqrt(0.29^2 + 1.39^2)) ≈ **0.565**

Sim(Q,S4) = (0.29*0.29)/(sqrt(0.29^2 + 1.39^2 + 1.39^2) * sqrt(0.29^2 + 1.39^2)) ≈ 0.029

Question 3:

Instruction (lec10, slide 28-29):

We use these formulas for the calculation of finding the missing rating for **user-based**.



- ❖ Finds the **most similar users** to the current user
- ❖ Cosine Similarity:

$$sim(u_1, u_2) = \frac{u_1 . u_2}{||u_1||.||u_2||}$$
$$= \frac{\sum_i r_{1,i} r_{2,i}}{\sqrt{\sum_i r_{1,i}^2}\sqrt{\sum_i r_{2,i}^2}}$$

❖Update the ratings:

$$r_{u,i} = \overline{r_u} + \frac{\sum_{v\in N(u)} sim(u,v)(r_{v,i} - \overline{r_v})}{\sum_{v\in N(u)} sim(u,v)}$$

$r_{u,i}$: predicted rating of user u for item i

$\overline{r_u}$: user u's mean rating

$\overline{r_v}$: user v's mean rating

$r_{v,i}$: observed rating of user v for item i

There are three main steps, which are:

1. Calculate the user-user similarity
2. Calculate the average rating of **<u>the user</u>**.
3. Calculate the missing rating of an item from that person.

Solution:

Calculate the rating of "Love at Last" from Alice

1. *<u>Calculate the user-user similarity</u>*

sim(Alice, Bob) = $\frac{2\times5\ +\ 3\times5}{\sqrt{2^2+3^2}\times\sqrt{5^2+5^2}} = 0.9805$

sim(A, C) = $\frac{2\times3+4\times5+3\times4}{\sqrt{29}\times\sqrt{50}} = 0.9979$

sim(A, D) = $\frac{4\times4+3\times5}{\sqrt{25}\times\sqrt{41}} = 0.968$

2. *<u>Calculate the average rating</u>*

$$\overline{r_B} = \frac{5+5+5}{3} = 5$$

$$\overline{r_C} = \frac{1+3+5+4}{4} = 3.25$$

$$\overline{r_A} = \frac{2+4+3}{3} = 3$$

$$\overline{r_D} = \frac{4+4+5}{3} = 4.33$$

3. *<u>Calculate the missing rating of "Love at last" from Alice.</u>*

Assume that neighborhood size = 2, we have:

$$r_{Alice,\ LAL} = \overline{r_A} + \frac{sim(A,B)\left(r_{B,LAL} - \overline{r_B}\right) + sim(A,C)\left(r_{C,LAL}\ - \overline{r_C}\right)}{sim(A,B) + sim(A,C)} = 1.865$$

Calculate the rating of “Nonstop Car Chases” from Bob

1. *Calculate the user-user similarity (only calculate the ones that we have not calculated above)*

sim(B, C) = $\frac{5\times1+5\times3+5\times4}{5\sqrt{3}\times\sqrt{26}} = 0.905$

sim(B, D) = $\frac{5\times4+5\times5}{\sqrt{50}\times\sqrt{41}} = 0.993$

2. *Calculate the average rating*

<Done in the above calculation>

3. *Calculate the missing rating of “Nonstop Car Chase” from Bob.*

$$r_{Bob,\,NCC} = \overline{r_B} + \frac{sim(A,B)(r_{A,NCC} - \overline{r_A}) + sim(D,B)(r_{D,NCC} - \overline{r_D})}{sim(A,B)+sim(D,B)} = 5.33$$

Calculate the rating of “Romance Forever” from Dave

sim(D, C) = $\frac{4\times1+4\times5+5\times4}{\sqrt{57}\times\sqrt{42}} = 0.899$

$$r_{D,\,RF} = \overline{r_D} + \frac{sim(A,D)(r_{A,RF} - \overline{r_A}) + sim(B,D)(r_{B,RF} - \overline{r_B})}{sim(A,D) + sim(D,B)} = 3.836$$

After the calculation, we have:

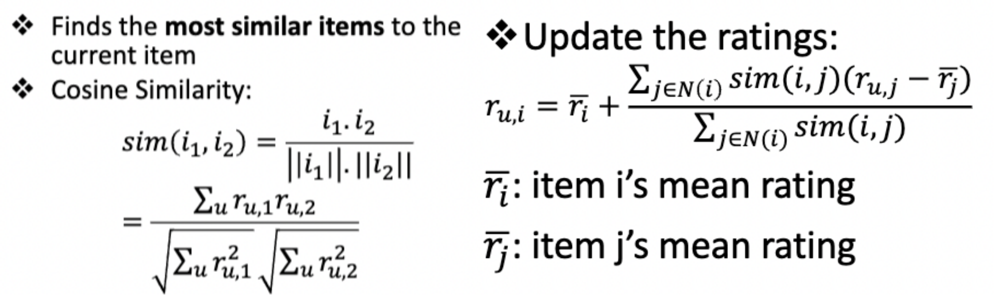
	Alice	Bob	Carol	Dave
Love at last	**1.865 | 4**	5	1	4
Romance forever	2	5	3	**3.836 | 3**
Nonstop car chases	4	**5.33 | 4**	5	4
Swords vs. karate	3	5	4	5

Calculate the predictive accuracy of the recommendation (User-based)

$$MAE = \frac{|1.865 - 4| + |5 - 4| + |3.836 - 3|}{3} = 1.323$$

$$RMSE = \sqrt{\frac{(1.865 - 4)^2 + (5 - 4)^2 + (3.836 - 3)^2}{3}} = 1.444$$

Item-based CF: With similar steps, we can calculate the missing rating using the item-based CF with these formulas (lec10, slide 28-29):



❖ Finds the **most similar items** to the current item

❖ Cosine Similarity:

$$sim(i_1, i_2) = \frac{i_1 . i_2}{||i_1||.||i_2||} = \frac{\sum_u r_{u,1} r_{u,2}}{\sqrt{\sum_u r_{u,1}^2}\sqrt{\sum_u r_{u,2}^2}}$$

❖ Update the ratings:

$$r_{u,i} = \overline{r_i} + \frac{\sum_{j \in N(i)} sim(i,j)(r_{u,j} - \overline{r_j})}{\sum_{j \in N(i)} sim(i,j)}$$

$\overline{r_i}$: item i’s mean rating

$\overline{r_j}$: item j’s mean rating

The steps are:

1. Calculate the item-item similarity
2. Calculate the average rating of **the items**
3. Calculate the missing rating of an item from a person.

Question 4:

a) What are the differences between feature selection and feature reduction?

One difference is (but not limited to) that feature selection uses existing features while feature reduction can create a new feature space.

Feature selection aims to improve model performance, reduce overfitting, enhance interpretability, and reduce computational complexity by eliminating irrelevant or redundant features.

Feature reduction methods are particularly useful when dealing with high-dimensional datasets, reducing computational complexity, and visualizing data in lower-dimensional spaces. However, feature reduction techniques may sacrifice some interpretability compared to feature selection since the transformed features may not directly correspond to the original ones.

b) Given the following image, which technique is used? Justify your choice.

The output seems to change into a different feature space (pc1, pc2). So, it is feature reduction, and the technique we use here could be Principal Component Analysis (PCA).

Question 5:

a. Who is correct? Why?

Adam is correct. The reason is that when looking at the figure above, there are negative number. However, in our dataset, there is none of the features having negative number.

b. Compared with the table's data, do you think that the graph makes sense? Explain why.

Ireland has some values quite different from the others such as Alcoholic drinks, Cheese, Fish, etc. So Ireland is an outlier compared to the others, which is show n in the image.

Question 6:

Index	Time	Sales ($)	Index	Time	Sales ($)
1	Spring 2017	4836	5	Spring 2018	5412
2	Summer 2017	5890	6	Summer 2018	6138
3	Fall 2017	6510	7	Fall 2018	6666
4	Winter 2017	7564	8	Winter 2018	8184
Total		**24800**			26400

a)

$$F_{2019} = \frac{A_1 + A_2}{2} = \frac{(4836 + 5890 + 6510 + 7564) + (5412 + 6138 + 6666 + 8184)}{2} = 25600$$

b)

From the result above, we can see that the prediction for total sell in 2019 has been decreased, which is not reasonable, since we can see the total sell is increasing.

This is because, in question **a)**, we only use two closest actual value (t-1 and t-2, assuming we want to calculate t); therefore, it seems like that we calculate the average of those two years (and we actually do). It leads to the conclusion that we need more actual values over a set time period, so that the calculation of moving average is more accurate. In our case, the smoothing effect happens in the calculation. This is one of the limitations of moving average, which cannot capture the trend well.

c)

$$AVG_{2017} = \frac{4836 + 5890 + 6510 + 7564}{4} = 6200$$

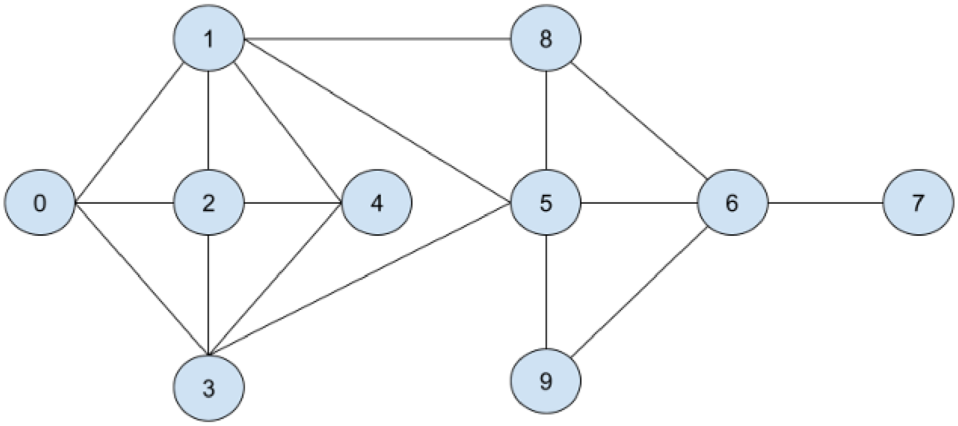
$$AVG_{2018} = \frac{5412 + 6138 + 6666 + 8184}{4} = 6600$$

d) Lec04, Page 37

Quarter	2017	Seasonal Index	2018	Seasonal Index	Average Index	2019
Spring	4836	**0.78**	5412	**0.82**	**0.80**	**5520**

Summer	5890	0.95	6138	0.93	0.94	6486
Fall	6510	1.05	6666	1.01	1.03	7107
Winter	7564	1.22	8184	1.24	1.23	8487
Average	**6200**		**6600**			**6900**

Question 7:



$$Alice\colon v_0 = v_1 = v_2 = v_3 = v_4 = v_5 = v_6 = v_7 = v_8 = v_9 = 0.5$$

$$Bob\colon v_0 = 3,\ v_1 = 5,\ v_2 = 4,\ v_3 = 4,\ v_4 = 3,\ v_5 = 5,\ v_6 = 4,\ v_7 = 1,\ v_8 = 3,\ v_9 = 2$$

$$Charles\colon v_0 = 3,\ v_1 = 1,\ v_2 = 2,\ v_3 = 2,\ v_4 = 3,\ v_5 = 1,\ v_6 = 2,\ v_7 = 5,\ v_8 = 3,\ v_9 = 4$$

a) Compute the adjacency matrix A of this network, where each cell A_{ij} is the number of edges from node i to node j .

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
0	0	1	1	1	0	0	0	0	0	0
1	1	0	1	0	1	1	0	0	1	0
2	1	1	0	1	1	0	0	0	0	0
3	1	0	1	0	1	1	0	0	0	0
4	0	1	1	1	0	0	0	0	0	0
5	0	1	0	1	0	0	1	0	1	1
6	0	0	0	0	0	1	0	1	1	1
7	0	0	0	0	0	0	1	0	0	0

8	0	1	0	0	0	1	1	0	0	0
9	0	0	0	0	0	1	1	0	0	0

b) Do they achieve the same ranking of nodes in terms of centrality values after the algorithm converges?

We can compute a few iterations and see that they will achieve the same ranking.

c) Between Alice and Bob, who reaches the final ranking faster (less iterations)? Explain your answer. (Note: we count the number of iterations excluding initialization and including the final iteration that reaches convergence)

We can compute a few iterations to see who faster.

d) Among Alice, Bob, and Charles, whose strategy has the least number of iterations? Whose strategy has the greatest number of iterations? Explain your answer.

We can compute a few iterations to see who faster.

Question 8:

Movies	**Tags**
M_1	Adventure, Action
M_2	Action, Comedy
M_3	Comedy, Adventure
M_4	Action, Fantasy, Adventure

a)

	M_1	**M_2**	**M_3**	**M_4**
t_1	1	0	1	1
t_3	0	1	1	0
t_5	0	0	0	0

b)

$$idf(t_1, C) = \ln \quad \frac{4}{3} = 0.29$$

$$idf(t_3, C) = \ln \quad \frac{4}{2} = 0.69$$

c)

$$tfidf(t_1, M_3, C) = 0.29 \times 1 = 0.29$$

$$tfidf(t_3, M_3, C) = 0.69 \times 1 = 0.69$$

d)

	M₁	**M₂**	**M₃**	**M₄**	**Q**
t₁	1	0	1	1	0
t₂	1	1	0	1	1
t₃	0	1	1	0	0
t₄	0	0	0	1	0
t₅	0	0	0	0	0

IDF Calculation:

$$idf(t_1, C) = \ln \frac{4+1}{3+1} = 0.22$$

$$idf(t_2, C) = \ln \frac{4+1}{3+1} = 0.22$$

$$idf(t_3, C) = \ln \frac{4+1}{2+1} = 0.51$$

$$idf(t_4, C) = \ln \frac{4+1}{1+1} = 0.91$$

$$idf(t_5, C) = \ln \frac{4+1}{0+1} = 1.60$$

Vectors:

M_1 = [0.22, 0.22, 0, 0, 0] M_3 = [0.22, 0, 0.51, 0, 0]
M_2 = [0, 0.22, 0.51, 0, 0] M_4 = [0.22, 0.22, 0, 0.91, 0]

e) The vector for the query "Action"

Q = [0, 0.22, 0, 0, 0]

f)

$$sim(M_1, Q) = \frac{0.22 \times 0.22}{\sqrt{0.22^2 + 0.22^2} \times \sqrt{0.22^2}} = 0.707$$

$$sim(M_2, Q) = \frac{0.22 \times 0.22}{\sqrt{0.22^2 + 0.51^2} \times \sqrt{0.22^2}} = 0.396$$

$$sim(M_3, Q) = \frac{0 \times 0.22}{\sqrt{0.22^2 + 0.51^2} \times \sqrt{0.22^2}} = 0$$

$$sim(M_4, Q) = \frac{0.22 \times 0.22}{\sqrt{0.22^2 + 0.22^2 + 0.91^2} \times \sqrt{0.22^2}} = 0.228$$

The ranking is: M_1 (0.707), M_2 (0.396), M_4 (0.228), M_3 (0)

g) If each movie has an additional tag "Hollywood", what is the result of the query "Action"? What is the result of the query "Hollywood"? And what is the result of the query "Hollywood Action"?

	M_1	**M_2**	**M_3**	**M_4**	**Hollywood**	**Hollywood Action**
t_1	1	0	1	1	0	0
t_2	1	1	0	1	0	1
t_3	0	1	1	0	0	0
t_4	0	0	0	1	0	0
t_5	0	0	0	0	0	0
Hollywood	1	1	1	1	1	1

If we add tag "Hollywood" to all movies, it does not have discriminate power. Therefore, the idf will be 0. The result of query "Action" will stay the same. The query "Hollywood" will have all movies as the same rank. The query "Hollywood Action" has the same result as "Action".

h) If the appearance order of tags in each movie matters, how do you propose to solve the retrieval problem? What is the result of the query "Action" now?

We can add extra point for the rank of the tag that has higher order in the description of a movie. The result of the query now will be: M2, M4, M1, M3 (since M2 and M4 has Action at the beginning).

The result of the query now will be: M2, M4, M1, M3 (since M2 and M4 has Action at the beginning)