RECOMMENDATION ENGINES

MASTER IN BUSINESS ANALYTICS AND BIG DATA

RECOMMENDATION ENGINES

LAB 2

- Building a Content-based Filtering Engine
- Building an Hybrid Recommendation Engine

RECOMMENDATION ENGINE LAB (PART 2)

Overview
Provide a recommendation engine for a question-and-answer website, Quora-like. The prerequisites for the first part of the assignment are that the engine must be based on users binary feedback and the question topics. For the second part of assignment, a set of hybrid methods have to be used to fine-tune the recommendations and more actions can be used.

Instructions
The **resolution can be done directly in the Excel file**. R,Python solutions are allowed. Each assignment has to be delivered in a separate tab within the excel file. If you choose the Excel file resolution, you can use the *SUMPRODUCT* and *CORREL* (Pearson Correlation) functions available among others.

Apart from the Excel file, or the R/Python code. The proposed solution for the Hybrid Challenge has to be explained in a **pdf document** to understand the rationale behind.

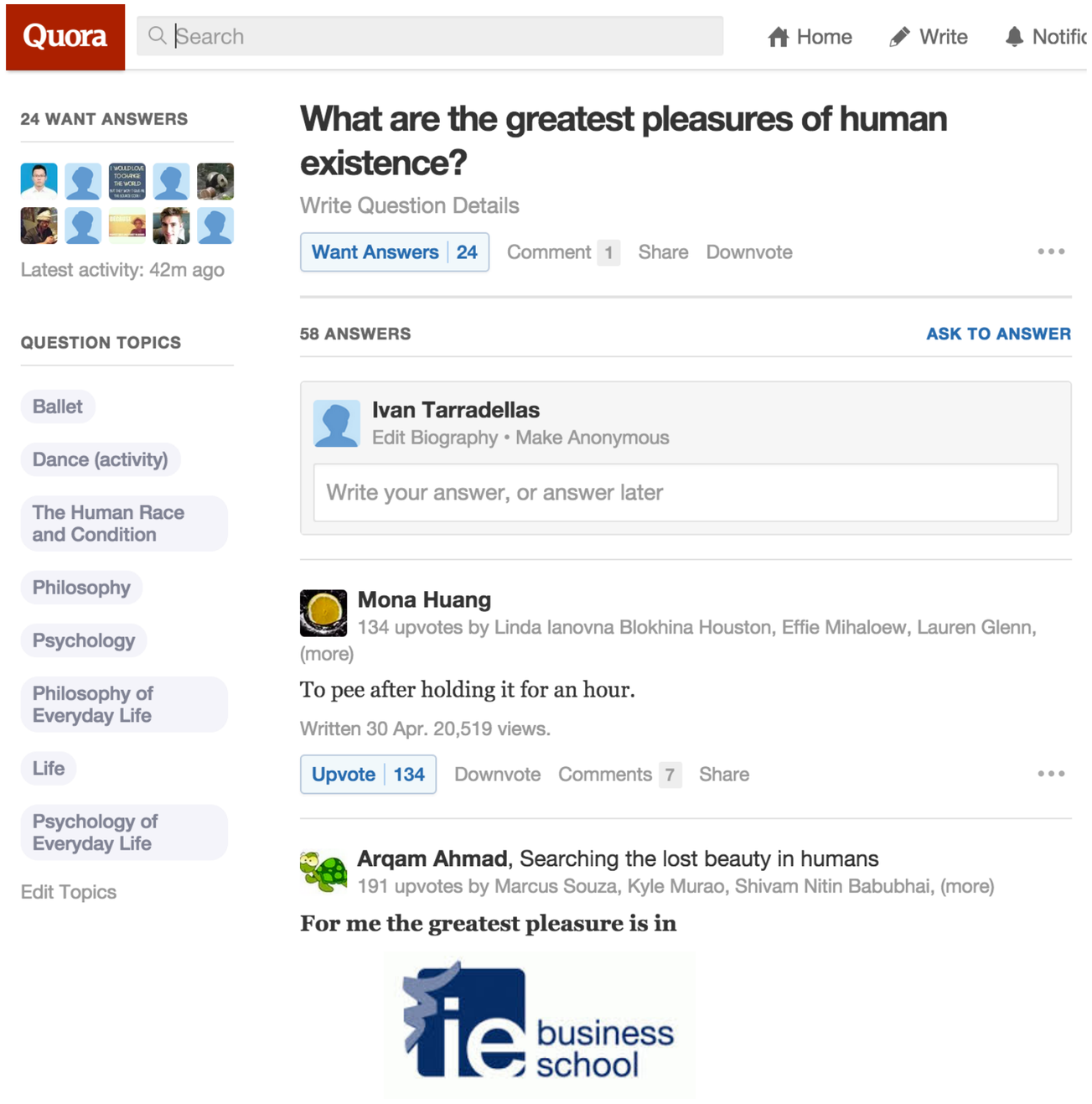
This assignment is **in group**.

RECOMMENDATION ENGINE LAB (PART 2)

1. Collected Data

- **Category**: questions and answers
- **Actions types**: explicit (binary ratings), implicit (popularity)
- **Format**: excel
- **Size**: 20x10 matrix

RECOMMENDATION ENGINE LAB (PART 2)



RECOMMENDATION ENGINE LAB (PART 2)

RECOMMENDATION ENGINE LAB (PART 2)

KUHORA	Topics (t)											User Feedback (f): Want answer / Down vote					User Answers (a): Up/Downs					Predictions			
	Sports	Books	Leadership	Philosophy	Society	Fiction	Security	Love	VideoGames	Superheroes		User 1	User 2	User 3	User 4		User 1	User 2	User 3	User 4		Pred1	Pred2	Pred3	Pred4
question1	1	0	1	0	1	1	0	0	0	1		1	-1				15								
question2	0	1	1	1	0	0	0	1	0	0		-1	1						40						
question3	0	0	0	1	1	1	0	0	0	0															
question4	0	0	1	1	0	0	1	1	0	0			1												
question5	0	1	0	0	0	0	0	0	1	1				1				2							
question6	1	0	0	1	0	0	0	0	0	0		1					25								
question7	0	0	0	0	0	0	0	1	0	1				-1											
question8	0	0	1	1	0	0	1	0	0	1				1				-4							
question9	0	0	0	0	0	1	0	0	1	0															
question10	0	1	0	0	1	0	1	0	0	0															
question11	0	0	1	0	1	0	0	0	1	0															
question12	1	0	0	0	0	1	1	0	0	0			-1	-1											
question13	0	0	1	1	1	0	0	1	0	0								-3	20						
question14	0	1	1	1	0	0	0	0	1	0															
question15	0	0	0	1	0	1	1	1	0	0				-1											
question16	1	0	0	0	0	1	0	0	1	0		1		-1			26								
question17	0	1	1	1	0	0	0	1	0	0			1					-4	82						
question18	0	0	0	1	0	0	0	0	1	0															
question19	0	1	1	0	1	0	1	0	0	1		-1						-2	110						
question20	0	0	1	1	0	0	1	0	1	0								3	50						
																					TOTAL				
DF																					Likes				
																					Dislikes				
																					Neutral				

User Profile	Sports	Books	Leadership	Philosophy	Society	Fiction	Security	Love	Videogames	Superheroes
User1										
User2										
User3										
User4										

RECOMMENDATION ENGINE LAB (PART 2)

2. Building a Content-based Filtering Engine

- **Simply Unary**. Given a set of users and questions, infer the users profile considering how many questions with its associated topics the user likes / dislikes. Use a dot product. Each user would end with a numeric value for each topic. With the user profiles, predict each user probability to like / dislike each question and count the total number of likes, dislikes and neutral predictions. To finalise, with the predictions provide the top-5 questions recommended per each user.
- **Unit Weight**. Some questions have more influence in the result as contain more topics. Normalise the topics frequency for each question and calculate the predictions again. Divide the keywords (topics) appearance by the total number of keywords that the question has. With the new predictions, provide the top-5 questions recommended per each user.
- **IDF**. With the unit weight applied, now evaluate the topics relevance using IDF. The higher the number of questions a topic has, the lower its relevance is. Rare topics would have more weight applying IDF now, thus being more relevant for the final prediction. With the new predictions, provide the top-5 questions recommended per each user.

RECOMMENDATION ENGINE LAB (PART 2)

Expected results:

- **Basic Profile.** Some expected results for this exercise:

User Profile	Sports
User1	3
User2	-2
User3	-2
User4	0

Predictions			
Pred1	Pred2	Pred3	Pred4
0,390	-0,298	-0,293	0

TOTAL				
Likes	7	15	5	0
Dislikes	11	4	10	0
Neutral	2	1	5	20

- **Unit weight**:

User Profile	Sports
User1	1,0333
User2	-0,5333
User3	-0,6667
User4	0,0000

Predictions			
Pred1	Pred2	Pred3	Pred4
0,428	-0,268	-0,382	0,0000

TOTAL				
Likes	10	16	4	0
Dislikes	10	4	13	0
Neutral	0	0	3	20

- **IDF:**

DF	4
IDF	0,6990

User Profile	Sports
User1	0,7223
User2	-0,3728
User3	-0,4660
User4	0,0000

Predictions			
Pred1	Pred2	Pred3	Pred4
0,490	-0,436	-0,451	0,0000

TOTAL				
Likes	10	14	5	0
Dislikes	10	6	14	0
Neutral	0	0	1	20

RECOMMENDATION ENGINE LAB (PART 2)

3. **Building an Hybrid Recommendation Engine** (30m)

   - **Switched Hybrid.** Consider the case of User4. User4 is new in the webpage and is not having a defined profile. Solve the User4 cold-start problem switching the content-based to non-personalise for users without actions collected. Provide the top-5 questions recommended per each user.
   - **Hybrid Challenge.** Define your own Quora-like RecSystem. Choose a feature-weighted linear stacking, a trust-aware CF, content-based similarity or build your own. It is key in this exercise to explain in detail your solutions with good argumentations. The “best argumented" solution will have the best note. Remember that **a RecSystem is not just an algorithm**. i.e. A good way to show it is to use a mockup of the site, or app, that you envision, pointing out the motivations and/or algorithm behind of each UI component (To collect implicit feedback, to fill the profile explicitly, to recommend non-personalised questions to solve cold-start, etc.)

RECOMMENDATION ENGINE LAB (PART 2)

EVALUATION			EXAMPLE
PARTIAL SCORE	Correct Predictions + Correct Hybrid Switch Solution	50 %	10
	Hybrid challenge	50 %	3
Homework Delivery		FINAL SCORE:	6,5