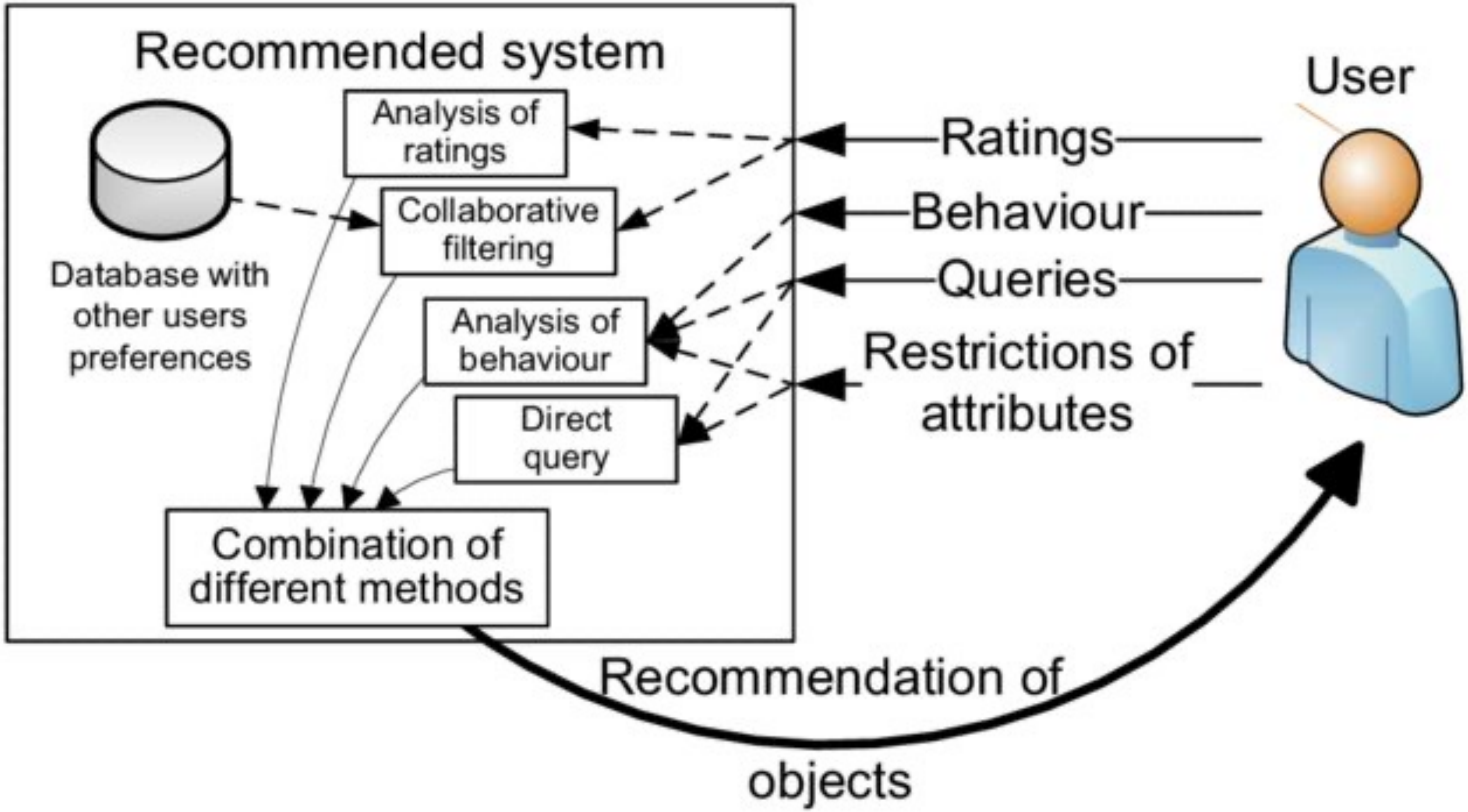
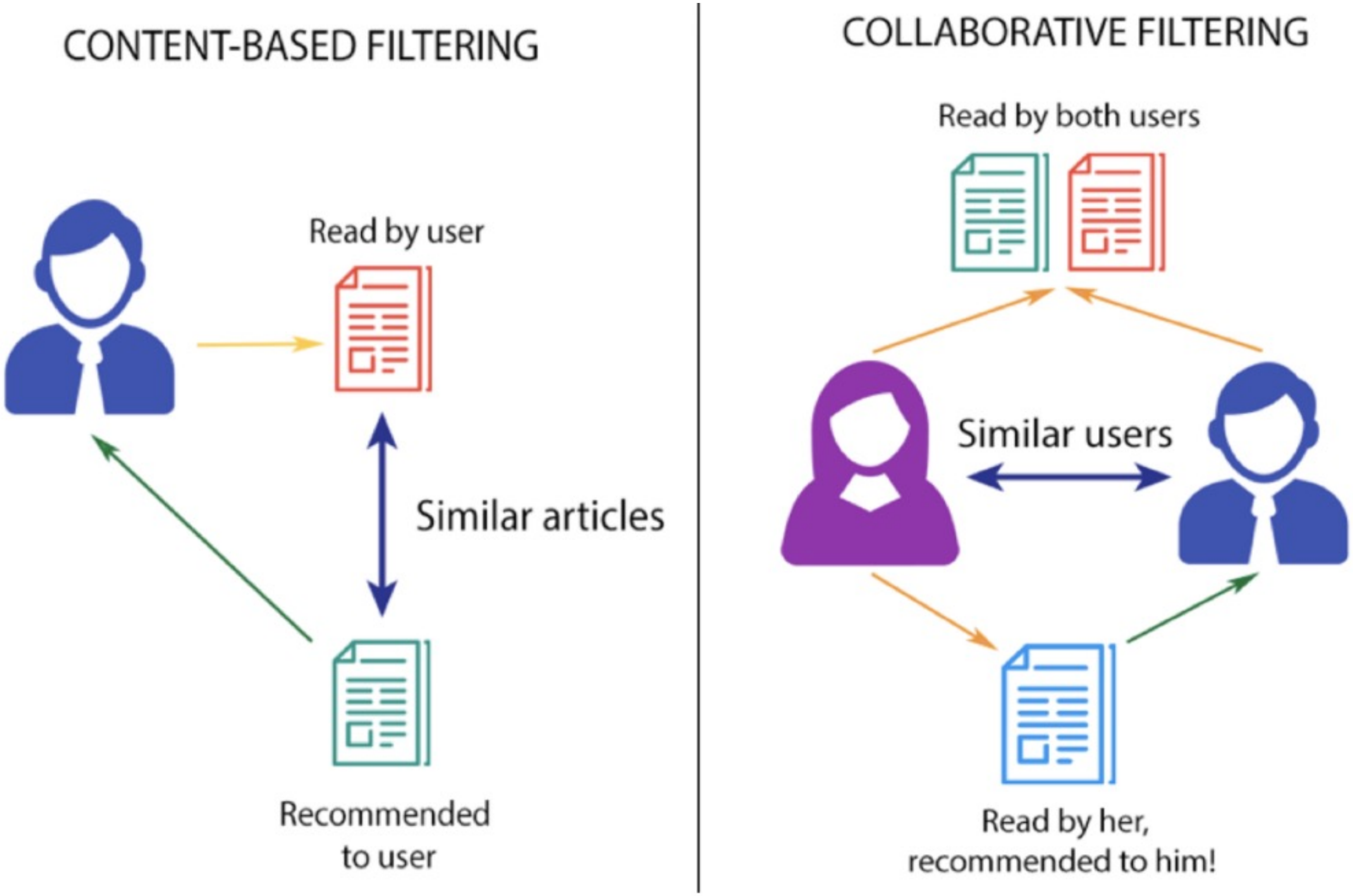
MA5851 Assessments – A2 debrief

- Assessment 2 (40%) – NLP Mapping (recommendation system)
  - Recap Recommendation System
  - Notes on Dataset
  - Dataset extension
  - Tasks / Deliverables
  - Expected solution
    - Approach
    - Input
    - Output
    - Some ideas
  - Marking Rubric

Recommendation System


Recommended system
Analysis of ratings
Collaborative filtering
Database with other users preferences
Analysis of behaviour
Direct query
Combination of different methods
User
Ratings
Behaviour
Queries
Restrictions of attributes
Recommendation of objects

Recommendation System



Recommender System

Content based algorithms

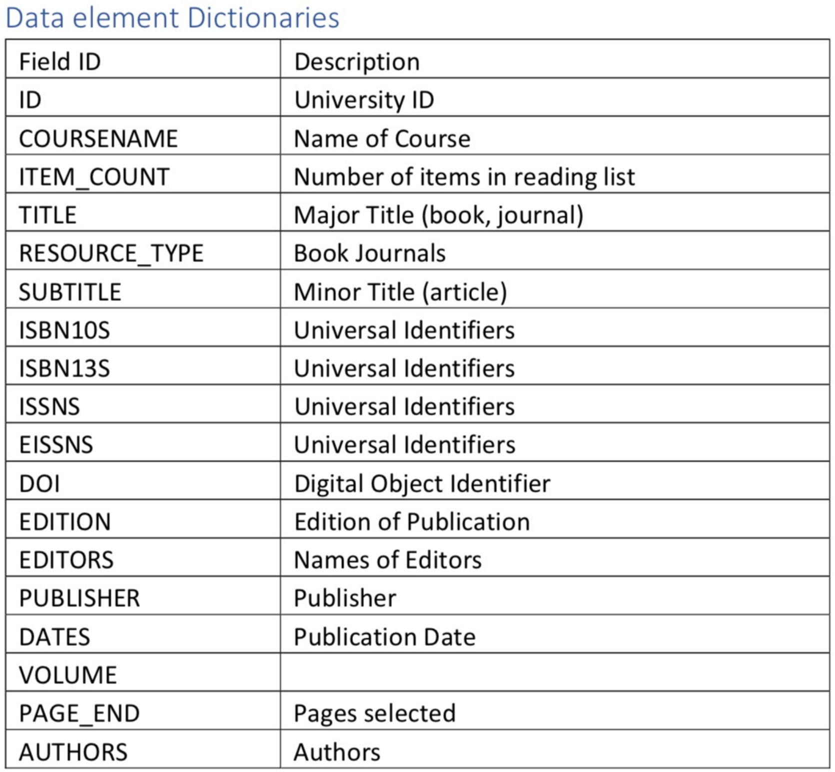
- Based on driving the content of the item. try to find *look alike* items and recommend them.
- Context level information is easier to get when the product/item explained with few dimensions
- TF-IDF score for text. The higher the TF*IDF score (weight), the rarer the term and vice versa.

Collaborative filtering algorithms

- not dependent on any additional information. (only transaction level information)
- User-User Collaborative filtering (User-Based KNN)
  - find look alike customer to every customer and offer products which first customer's look alike has chosen in past
  - This algorithm is very effective but not scalable since it requires to compute every customer pair information
- Item-Item Collaborative filtering (Item-Based KNN)
  - finding item look alike
  - having item look alike matrix, recommend alike items to customer who have purchased any item from the store
  - less resource consuming than user-user collaborative filtering
- Other simpler algorithms: other approaches like market basket analysis, do not have high predictive power.

Dataset

- Input
  - Courses and list of readings
  - 4 institutions/university
  - 3634 unique courses
  - 44003 unique reading items
  - Types of readings:
    - Journal, Book, Website, Proceedings, Webpage, Document, Article, Chapter, AudioVisualDocument, LegalCaseDocument, Legislation, Image, Thesis, AudioDocument, Page, LegalDocument, Report
  - Total 68,530 records
  - Data is misaligned, incomplete, messy
  - Reliable fields: ID, coursename, title, subtitle, resource_type,



Data element Dictionaries

Field ID	Description
ID	University ID
COURSENAME	Name of Course
ITEM_COUNT	Number of items in reading list
TITLE	Major Title (book, journal)
RESOURCE_TYPE	Book Journals
SUBTITLE	Minor Title (article)
ISBN10S	Universal Identifiers
ISBN13S	Universal Identifiers
ISSNS	Universal Identifiers
EISSNS	Universal Identifiers
DOI	Digital Object Identifier
EDITION	Edition of Publication
EDITORS	Names of Editors
PUBLISHER	Publisher
DATES	Publication Date
VOLUME	
PAGE_END	Pages selected
AUTHORS	Authors

Dataset extension

Google Book API Example

- https://www.googleapis.com/books/v1/volumes?q=%22Post-Colonial%20Studies:%20the%20Key%20Concepts%22

Trove API

- https://trove.nla.gov.au/about/create-something/using-api/api-technical-guide#examples
  - https://api.trove.nla.gov.au/v2/result?key=l3bajnd6bukre2p6&zone=book&q=%22essential%20guide%20to%20Rapunzel%27s%20world%22

OCLC WorldCat

- http://classify.oclc.org/classify2/
  - http://classify.oclc.org/classify2/ClassifyDemo?search-standnum-txt=9780199022274&startRec=0
- https://www.oclc.org/research/areas/data-science/fast/applications.html
  - http://classify.oclc.org/classify2/Classify?isbn=9781863955799&detail=true

Linked Data

- https://www.oclc.org/developer/develop/web-services/fast-api/linked-data.en.html
  - http://experimental.worldcat.org/fast/search?query=cql.any+%3D+%22diabetes%22&httpAccept=application/xml

Three use cases or perspectives

- Recommend existing course material to similar subjects, or
- Recommend new reading material to existing subjects, or
- Provide a complete reading list of existing readings for a new subject.

Tasks

- Develop two NLP recommendation engines
  - Using the reading list material (supplied data)
  - Clean, omit, enhance dataset
  - NLP recommenders conform to one of the three perspectives
- Determine the quality of both NLP recommenders from Task 1
  - using test and training sets derived from the supplied data
- Compare the two NLP recommenders
  - Write report on assumptions, techniques, and result

Deliverables

- Two requirements:
  - Saved report file in the following format A2_NLP_Reccomender_firstname_lastname (PDF format)
    - Length: 3000 words (+/-10%)
    - 12pt font size with 1.5 spacing
    - APA referencing style applied.
  - Files:
    - Your transformed data file(s)
    - Python Notebook (.ipynb), Python scripts (.py) or OpenRefine GREL code (text file) with the information about the version of Python/OpenRefine that you have used and any associated package used.

System: Input and Output

- Input: a set of keywords
  - eg, computational biology, ethics in artificial intelligence
- Output: a list of recommended readings (Top N) sorted according to some relevance score
  - Book: <title>*<isbn/issn/doi><publisher><author?>*(77%)
  - Journal:<title>*<isbn/issn/doi><publisher><editor>*<(76%)
  - Journal:<title><isbn/issn/doi><publisher><editor>(75%)
- How to measure accuracy of the system?
  - Confusion matrix (sensitivity, specificity, F1-score)
  - Acceptance of the provided output (user study)
- Like any ML problem
  - 80% training data 20% test data

example approach: Recommendation System

- Statistical approach:
  - Most frequent words: course vs title
- User approach vs Product Approach
  - Consider “coursename” as ‘user’ and “title” (metadata like abstract/description) as ‘product’
  - Build ‘user’ profile
    - Use your own taxonomy, or FOE
    - grouping, labeling
  - Build ‘product’ profile
  - Associate each product to a N-many user profile
    - With a score
    - This is your knowledgebase

Expected solution: Recommendation System

- How it works
  - Input: transform your input query to a “user” like your user profile vector. In this case this is a new “user”
- Content based filtering
  - Recommend those which are preferred by similar ”users” to ”new user”
- Collaborative filtering
  - User – User: new user similar to existing users
  - Item – item: new item to recommend existing user (out of scope)

Expected solution: Recommendation System

- Preprocessing: Stemming, Lemmatisation
- Parsing → NP
- N-gram analysis → dictionary approach
- Text Similarity: Word2Vec embedding
- Vectorisation: BOW, TF-IDF
- Clustering
- Association rule: {bow} => {bow}
- ANN: bow for courses => bow for titles

Marking Rubric

- X-axis (50%)
  - Quality of Task: effective use of NLP resources and techniques
  - Quality of output: showing relevance, robustness etc.
- Y-axis (50%)
  - Report on techniques, approaches, assumptions