Content-based Filtering

PART I

Dr Suncica Hadzidedic
suncica.hadzidedic@durham.ac.uk

Lecture outline

Topics	Activities

• What is CBF?
• Applications
• Benefits, drawbacks
• Architecture
• Content analysis
  • Vector Space Model
    • TFIDF
	• Task: paper analysis
	• Task: computing a Vector Space Model
	• Task: explore datasets, frameworks, ChatGPT

What is CBF?

- **Content-based filtering**
  - analyses **features** of the items / **documents**
  - previously **rated** by a user
  - then builds a **model / profile** of user interests

- **Aim**
  - recommend items similar to the items this user has liked in the past

Durham University

Source: Lops (2011)

Applications, benefits, drawbacks

Content-Based Movie Recommendation System Using Genre Correlation, Reddy (2019)

- Features considered
  - genres user might prefer
- Approach
  - content-based filtering using genre correlation
- Dataset
  - Movie Lens

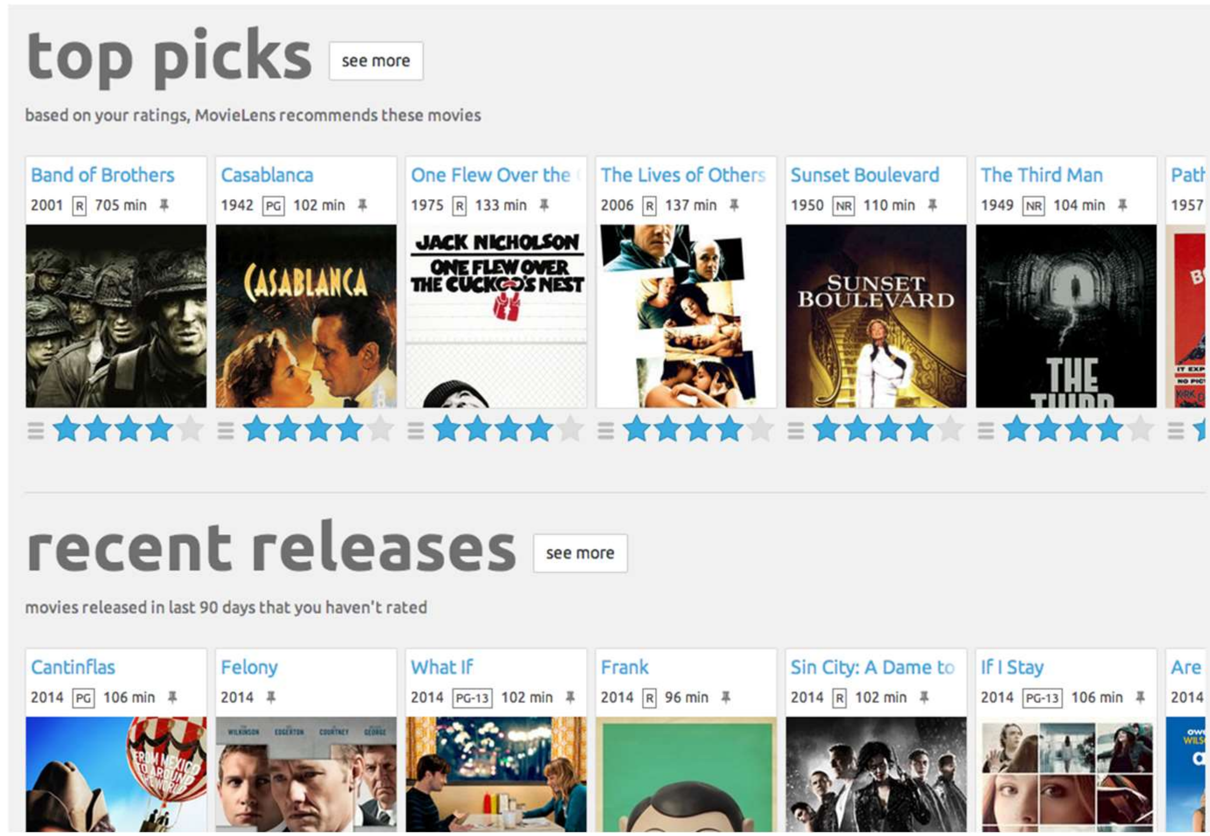


Image source: https://movielens.org/

Durham University

Publication Recommender System, Wang (2018)

- Finding relevant CS publication venues
  - 66 venues
  - 5 digital libraries (Springer, IEEE, ACM, AAAI and SIAM)
- Content-based filtering model
  - Feature selection model -> chi-square
  - Softmax regression model
- Data
  - abstract or whole manuscript

CORE Journal Portal
Computing Research & Education

Back to CORE homepage | search conferences

computer science Search by: All Source: ERA2010
Search
Showing results 1 - 30 of 30

Title	Source	Rank
Algorithmica: an international journal in computer science	ERA2010	A*
Journal of Computer and System Sciences	ERA2010	A*
Theoretical Computer Science	ERA2010	A
Science of Computer Programming	ERA2010	A
Computer Science Education	ERA2010	A
Logical Methods in Computer Science	ERA2010	A
Discrete Mathematics and Theoretical Computer Science	ERA2010	B
Journal of Computer Science and Technology	ERA2010	B
International Journal of Foundations of Computer Science	ERA2010	B
Innovations in Teaching and Learning in Information and Computer Sciences	ERA2010	B
Social Science Computer Review	ERA2010	B
International Journal of Applied Mathematics and Computer Sciences	ERA2010	C
Journal of Universal Computer Science	ERA2010	C
Mathematics in Computer Science	ERA2010	C
Egyptian Computer Science Journal	ERA2010	C
IAENG International Journal of Computer Science	ERA2010	C
International Journal of Computer and Information Science and Engineering	ERA2010	C
Automatic Control and Computer Sciences	ERA2010	C
International Journal of Computer Science and Engineering	ERA2010	C

University

Benefits of CBF

User independence

- Use only ratings to build user profile
- vector space models

Transparency

- Recommendations can be explained
- List content features / descriptions

New Item

- New items recommended
- Not susceptible to first-rater problem

Durham University

Source: Lops (2011)

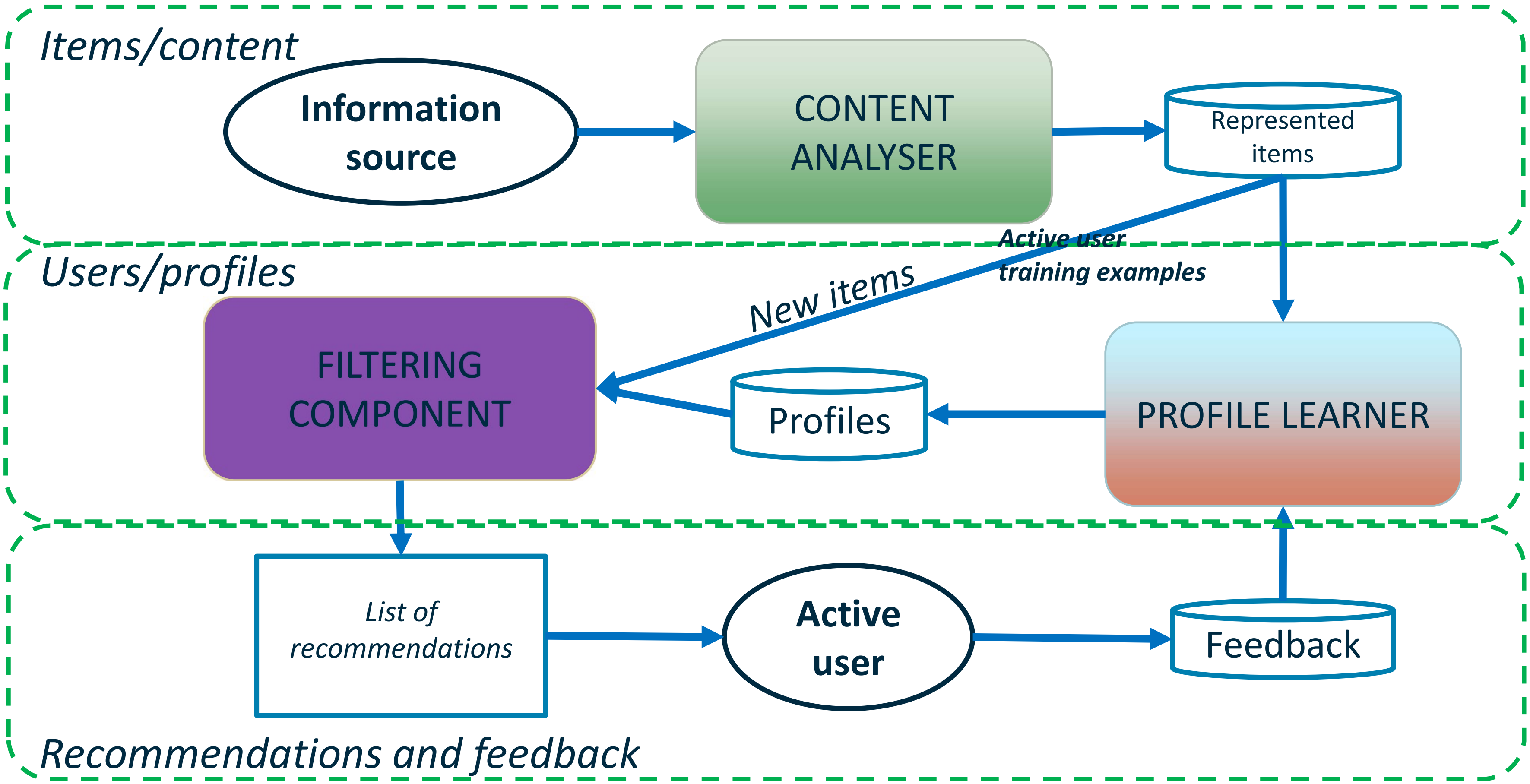
Drawbacks of CBF

Limited Content Analysis	• Limit in the number and type of features • Domain knowledge needed • Enough information required to discriminate between P and N items • Harder to find complements than substitutes • Weighing attributes and ratings; content interdependency
Over-specialisation	• No inherent method for finding unexpected items • Lack of serendipity • Limited novelty
New User	• Need to collect enough ratings • Recommendations for new user not reliable
User Profile Updating	• throw away/recompute • mix in new rating; decay old profile over time

Source: Lops (2011), Konstan (2019)

CBF Architecture

CBF Architecture



Durham University

Source: Lops (2011)

Content analyser

- **Aim**
  - Represent items' content in a structured form
    - Type 1: structured - same number of attributes, with known set of values
    - Type 2: unstructured data *(e.g., tags, posts, opinions, etc.)*
- **Feature extraction techniques**
  - Transforming original information space to target one
    - e.g., Web page -> keywords
- **Feature selection techniques**

Source: Lops (2011)

Profile learner

- **Aim**:
  - Collect and generalise user preference data
  - construct **user profile**
- **Represented items** repository
- **Feedback** repository
  - Explicitly defined user interests
  - Inferred from reactions to recommendations
- **User profile**
  - inferred preferences
  - supervised learning algorithm
    - Trained on set of item representations with the user's ratings

Durham University

Source: Lops (2011)

Filtering component

- **Aim:**
  - Apply user profile model to new item representations
    - Match user profile representation to item representation
    - Generate prediction / relevance judgement /score
    - Present a (ranked) list of item recommendations

Source: Lops (2011)

1. Content analysis

Item description

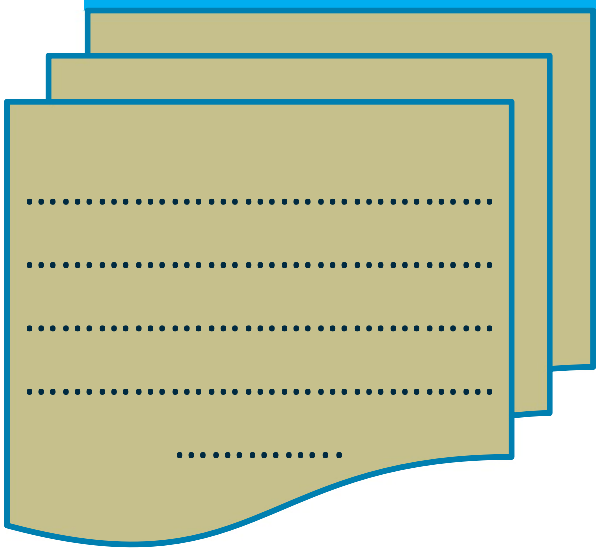
- Structured data
  - Attributes
- Unstructured data
  - **Keyword-based approach**
    - Issues / drawbacks
      - Requires training sets with a large number of examples
      - Lack of "intelligence" - polysemy, synonymy
  - **Semantic analysis**
    - knowledge bases (lexicons or ontologies)
      - "semantic" interpretation of user information needs

Durham University

Source: Lops (2011), Wang (2018)

Vector Space Model (VSM)

- Commonly used in CBF RSs
- Terminology
  - **Corpus** (set of documents)
    - $D = \{d_1, d_2, \ldots, d_N\}$
  - **Dictionary** (set of words in the corpus)
    - $T = \{t_1, t_2, \ldots, t_n\}$



Durham University

Source: Lops (2011)

Feature extraction, selection and weight

1. **Pre-processing**
   - tokenization, stopwords removal, stemming, lemmatization
2. Compute **selection** metric
   - Sort metric values
     - pick **rich informative features**
3. Combine all feature **vectors**
   - remove duplicate terms
   - generate **new feature vector space**
     - all documents represented by vectors of equal length
4. **Weighting** scheme

Durham University

Source: Lops (2011), Wang (2018)

Feature selection metrics

Chi-square (χ2)

$$\chi^2(t, c) = \frac{N \times (AD - BC)^2}{(A + C) \times (B + D) \times (A + B) \times (C + D)}$$

Information gain

$$IG(t, c) = Entropy(S) + \frac{A+B}{N_1+N_2}\left(\frac{A}{A+B}\log\left(\frac{A}{A+B}\right) + \frac{B}{A+B}\log\left(\frac{B}{A+B}\right)\right) + \frac{C+D}{N_1+N_2}\left(\frac{C}{C+D}\log\left(\frac{C}{C+D}\right) + \frac{D}{C+D}\log\left(\frac{D}{C+D}\right)\right)$$

$$Entropy(S) = -\frac{N_1}{N_1+N_2}\log\left(\frac{N_1}{N_1+N_2}\right) - \frac{N_2}{N_1+N_2}\log\left(\frac{N_2}{N_1+N_2}\right)$$

Mutual information

$$MI(t, c) = \log\left(\frac{A/(A+C)}{(A+B)/N}\right)$$

- A - number of documents including term t, which belongs to category c
- B - number of documents including t, which does not belong to c
- C - number of documents in category c, which does not include t
- D - number of documents in other categories and without term t
- N – size of the corpus; total number of documents

Durham University

Source: Wang (2018)

Keyword vector

- **Item/document representation**
  - All items represented by vectors of equal length (same terms)
  - a vector of term weights
  - in an *n*-dimensional space
    - **dimension** is a term from the dictionary of the corpus
      - E.g., in movies these could be: different genres, all possible actors and directors
    - **weight** is the degree of association between the item and the term

	present	upgrade	colour
item1	1	0	1
item2	0	1	1
item3	0	0	0

Source: Konstan (2019)

Weighting scheme: TF-IDF

- The most common weighting scheme
- Terms/features with **higher TF-IDF** are more important
  - rare terms (across documents/items) are not less relevant than frequent terms (**IDF**)
  - multiple occurrences of a term in a document not less relevant than single occurrence (**TF**)
  - long documents not preferred (**normalisation**)

Source: Lops (2011)

TF-IDF: Measures

- Term frequency (TF)
  - T - term frequency
  - L - count of unique words in the document
  - T_i - frequency of the most frequent word in the document

$$TF = \frac{T}{L} \text{ or } TF = \frac{T}{Ti}$$

Durham University

Source: Lops (2011), Wang (2018)

TF-IDF: Measures

- TF-IDF
  - N - number of all documents
  - n_k - number of documents with term k
    - Logarithm – result turned to a useful scale

$$\text{TF-IDF}(t_k, d_j) = \underbrace{\text{TF}(t_k, d_j)}_{\text{TF}} \cdot \underbrace{log \frac{N}{n_k}}_{\text{IDF}}$$

Durham University

Source: Lops (2011), Wang (2018)

TF-IDF: Measures

- (cosine) **Normalisation**
  - Weighted frequency $\mathbf{w}_{k,j}$ of term k in document j
  - documents represented by vectors of terms with weights in [0,1] interval

$$w_{k,j} = \frac{\text{TF-IDF}(t_k, d_j)}{\sqrt{\sum_{s=1}^{|T|} \text{TF-IDF}(t_s, d_j)^2}}$$

Durham University

Source: Lops (2011), Wang (2018)

Key topics to take away

- CBF
  - recommend items similar to the items the user has liked in the past
- Architecture
  - Content analyser
  - Profile learner
  - Filtering component
- Vector Space Model
  - TFIDF weighting
- User profile learning – classifier computation
  - Nearest neighbour algorithms
    - Similarity measures: cosine, Pearson correlation, Euclidean, etc.
  - Probabilistic methods
  - Rocchio’s method
  - Other classifiers
- Benefits / drawbacks

Durham University

Task

- Have a look at the datasets I posted on our Ultra page. Or search for publicly available datasets for RSs.
- Choose one dataset you would like to use for the assignment, i.e. for developing your RS.
  - E.g. Amazon reviews dataset
- Consider the unstructured data; how item features would be extracted; and what the item vector would look like.

r online_store	up	r	brand	category	sub_category	product_description	manufacturer	ma	m	ti	dimension1	dimension2
FRESHAMAZON	#	E	Dove Men+Care	Personal Care	Deos	Dove Men+Care Extra Fresh Anti-per	Unilever Global	UK			Deos	Male Anti-Perspirant Deodorant
FRESHAMAZON	#	E	Marmite	Foods	Savoury	Marmite Spread Yeast Extract 500g	Unilever Global	UK			Savoury	COTC Yeast Extract
FRESHAMAZON	#	E	Marmite	Foods	Savoury	Marmite Spread Yeast Extract 500g	Unilever Global	UK			Savoury	COTC Yeast Extract
FRESHAMAZON	#	E	Knorr	Foods	Savoury	Knorr Beef Stock Pot 8 x 28g	Unilever Global	UK			Savoury	Beef Stock/Pots/Cubes/Extract/Liquid/Concentrated
FRESHAMAZON	#	E	Cif	Homecare	HHC	Cif Citrus Bathroom Mousse 500ml	Unilever Global	UK			HHC	Bathroom Mousse
AMAZONPRIMEP	#	E	Marmite	Foods	Savoury	Marmite Spread Yeast Extract 500g	Unilever Global	UK			Savoury	Yeast Extract
AMAZONPRIMEP	#	E	Marmite	Foods	Savoury	Marmite Spread Yeast Extract 500g	Unilever Global	UK			Savoury	Yeast Extract
AMAZONPRIMEP	#	E	Knorr	Foods	Savoury	Knorr Beef Stock Pot 8 x 28g	Unilever Global	UK			Savoury	Beef Stock/Pots/Cubes/Extract/Liquid/Concentrated
FRESHAMAZON	#	E	Dove Men+Care	Personal Care	Deodorants & I	Dove Men+Care Clean Comfort Aer	Unilever Global	UK			Deodorants & Fra	Male Anti-Perspirant Deodorant
FRESHAMAZON	#	E	Knorr	Foods	Savoury	Knorr Chicken Stock Pot 8 x 28g	Unilever Global	UK			Savoury	Chicken Stock/Pots/Cubes/Extract/Liquid/Concentrated
FRESHAMAZON	#	E	Knorr	Foods	Savoury	Knorr Rich Beef Stock Pot 8 x 28g	Unilever Global	UK			Savoury	Beef Stock/Pots/Cubes/Extract/Liquid/Concentrated
FRESHAMAZON	#	E	Dove Men+Care	Personal Care	Deodorants & I	Dove Men+Care Clean Comfort Aer	Unilever Global	UK			Deodorants & Fra	Male Anti-Perspirant Deodorant
FRESHAMAZON	#	E	TRESemmé	Personal Care	Hair	Tresemme Moisture Rich Condition	Unilever Global	UK			Hair	Women General Cleanse Conditioner
AMAZON	#	E	Cif	Homecare	Household Car	Cif Citrus Bathroom Mousse 500ml	Unilever Global	UK			Household Care	Cleaning Spray
AMAZON	#	E	TRESemmé	Personal Care	Hair	Tresemme Silky Smooth Conditioner	Unilever Global	UK			Hair	Women General Cleanse Conditioner

Task

See the full task specification on the discussion board.

1. Vector Space Model for item and user profile vectors
2. Dataset *DS CBFRS*
3. Calculate 3 models
   - Model 1: Simple model/user profile for U1
     - For each attribute in the user profile, determine its score, by taking into account U1's evaluation of each article
     - Predict how much U1 will like each of the articles
   - Model 2: Normalised model
     - Normalise each article's attribute values
     - Repeat: build a user profile for U1 and predict article scores
   - Model 3: TF-IDF weighted model
     - Calculate IDF for each attribute
     - Use the normalised U1 user profile to predict U1's scores for each article weighted by the IDF

Adapted from: Konstan (2019)

Week 1 Task: Analytical Framework - Netflix

1. **Domain:** Streaming media; movies and television shows; homogenous items
2. **Purpose:** Content recommendations; revenue; longer engagement; encourage loyalty - new content for long-term involvement
3. **Recommendation context:** device accessibility; on the go; stay at home
4. **Knowledge source:**
   - Item descriptions
   - User interactions - Genres that the user frequently watch, movies/series bookmarked to 'My List' etc.
5. **Personalization level:**
   - Persistent
   - Non-personalized, demographic, session-based
6. **Data used for recommendation:**
   - Explicit **-** initially submit examples of preferred movies; provide feedback - dislike/like/love system
   - Implicit - user's watch history, length of watching, My List, context (inferred - location, age, gender)
7. **Recommendation techniques: Collaborative filtering** and content-based filtering.
8. **User interface:**
   - Input: initial selection; account completion; scrolling; clicking; viewing; rating; adding to my list; share; download; search; filter.
   - Output: Media icons; video preview; category of items/recommendations; rows or grid of items. Information relating to selected title.

Task

You will be divided into three groups for this task.

1. **Group 1.** Have a look at the datasets I posted on our Ultra page or search for publicly available datasets for RSs that would be suitable for the RS you want to develop.
   - What challenges do you notice? Size? Accessibility? Availability of data?
   - Does the dataset include relevant data for a recommender system: user and items ids, expression of preference?
2. **Group 2:** Explore RS frameworks and libraries.
   - List the frameworks you have discovered and what they are useful for.
   - Can you use any for your RS and how? To what extent will you need additional time and training to use the framework? Will it be applicable to a dataset you are considering?
3. **Group 3:** Explore the use of ChatGPT to collect information for your RS or understand the trends in RSs.
   - You can ask the above questions about the datasets or frameworks or example implementation of a CBF RS.
   - Or chat on the topic of the latest trends, state of the art, in content-based filtering.
   - Or ask for an explanation about the topics we have covered that might not yet be clear: semantic analysis, prediction vs recommendation, examples of feature selection methods, deep learning methods for feature extraction from image data, etc.
   - Or use the questions from the learning audit link.

Task

- Select, read and analyse **one** of the papers on CBF applications to movies, music or e-learning, DIY - attached below. Choose a paper that would be relevant for designing/developing your RS.
- Post a brief reflection on this discussion board on the following:
  - What are the domain's characteristics?
  - What dataset(s) is used?
  - How are items and user preferences represented?
    - Which data (type) is used?
    - If unstructured data – which approach is used for generating item representations and user profiles; which feature extraction and selection methods were used?
  - What methods were used for content-based filtering:
    - Which weighting scheme (if any) is used?
    - Which similarity measure is applied?

References and reading material

- Ahn, J., Brusilovsky, P., Grady, J., He, D., Syn, S.Y.: Open User Profiles for Adaptive News Systems: Help or Harm? In: C.L. Williamson, M.E. Zurko, P.F. Patel-Schneider, P.J. Shenoy (eds.) Proceedings of the 16th International Conference on World Wide Web, pp. 11–20. ACM (2007)
- Amatriain, X., Jaimes, A., Oliver, N., & Pujol, J. M. (2011). Data mining methods for recommender systems. In *Recommender systems handbook* (pp. 39-71). Springer, Boston, MA.
- Konstan, J. & Ekstrand, M. (2019). Introduction to Recommender Systems: Non-Personalized and Content-Based. Available: https://www.coursera.org/learn/recommender-systems-introduction/home/welcome
- Lops, P., De Gemmis, M., & Semeraro, G. (2011). Content-based recommender systems: State of the art and trends. In *Recommender systems handbook* (pp. 73-105). Springer, Boston, MA.
- Reddy, S. R. S., Nalluri, S., Kunisetti, S., Ashok, S., & Venkatesh, B. (2019). Content-Based Movie Recommendation System Using Genre Correlation. In *Smart Intelligent Computing and Applications* (pp. 391-397). Springer, Singapore.
- Wang, D., Liang, Y., Xu, D., Feng, X., & Guan, R. (2018). A content-based recommender system for computer science publications. *Knowledge-Based Systems*, *157*, 1-9.

Durham University