Personalization and Social IR

M2R – MOSIG
2019-2020

Philippe Mulhem

Philippe.Mulhem@imag.fr

Outline

- Personalization in IR
- PIR : Personalized Information Retrieval
  – Vosecky et al 2014
- Socialized PIR
- Social documents IR
- Conclusion

Personalization in IR

- Different users, same query
  ➔Different answers

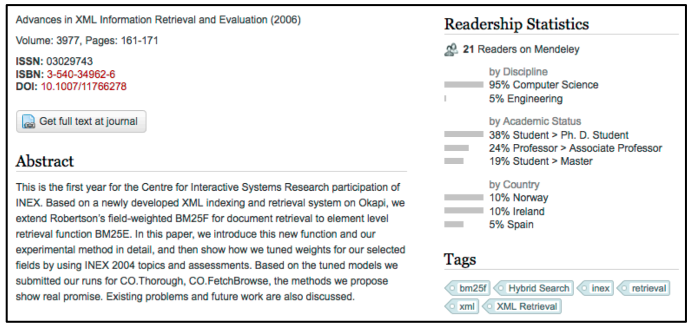
- Examples
  – User interested in Formula 1 Grand prix looking for « Singapore » wants to have infos about the grand prix in November
  – User interested in Orchids flowers looking for « Singapore » should get infos about Orchid Garden for instance

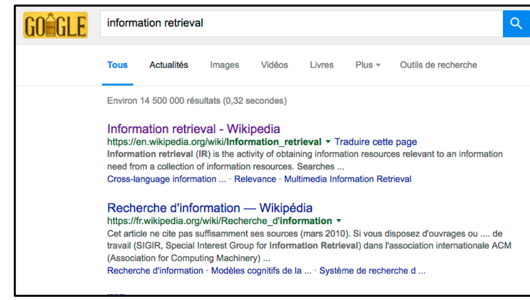
Personalization in IR

- Stages (from [Ghorab et al. 2013])
  - Information gathering
    - From where ?
  - Information representation
    - Into What ?
  - Usage of the representation
    - How ?

PIR – Information Gathering

- What sources may help to learn from the user's interests





  - Implicit
    - Logs (clicks, tags, bookmarks, queries)
      - [Jiang et al 2016]: 26 billions of clicks <query, doc>
      - [Bouadjenek 2013, Xu 2010, Vallet 2010]: tags
  - Explicit
    - User keywords, categories (age, living city, …)

PIR – Information Representation

- Usually based on vectors of <tag, weight>
  - Weighting: some kind of tf.idf of user's tags
    - [Xu et al. 2008]:

$$w_{t,u} = tf(t,u) * \log(\frac{N_u}{n(t,u)})$$

tf(t,u): term frequency of tag t for user u

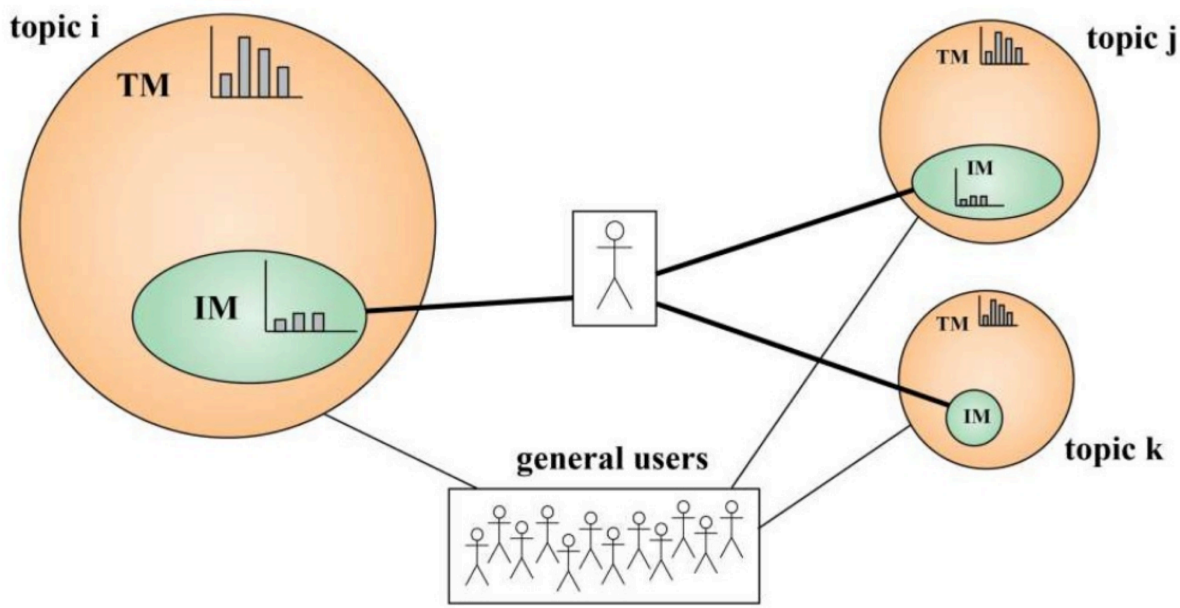
N_u: number of documents tagged by u

n(t,u) number of documents tagged by u with term t

  - How to cope with users that have several centers of interests?

PIR – Information Representation

- … or more complex representations, as in [Vosecky et al. 2014] on tweets
  - Hierachical representation: topics → words
  - Individual Model



From https://fr.slideshare.net/janvosecky/collaborative-personalized-twitter-search-with-topiclanguage-models

PIR – Information Representation

[Vosecky et al. 2014]: Individual user Model (IM)

Hierachical representation: topics → words

- Step 1. Apply Latent Dirichlet Allocation (LDA) on the whole tweet corpus (learn global topics): learn k latent topics (unobservable) and the distributions of probabilities of all words in these topics: ϕ_k^{TM}
- Step 2. Obtain inividual distribution of terms from a user u for each topic: using the tweets written by u
- Step 3. Fuse user specific and global LDA

LDA (short overview)

From http://blog.echen.me/2011/08/22/introduction-to-latent-dirichlet-allocation/

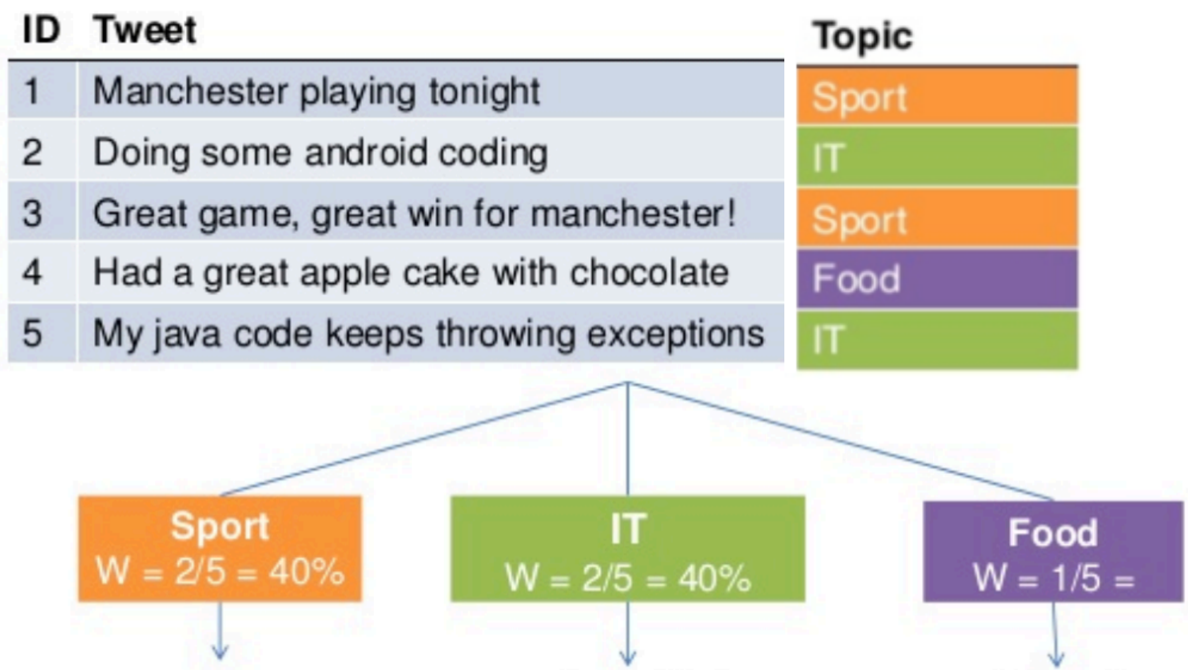
- Suppose we have the following set of 5 sentences:

1. I like to eat broccoli and bananas.	**4.** My sister adopted a kitten yesterday.
2. I ate a banana and spinach smoothie for breakfast.	**5.** Look at this cute hamster munching on a piece of broccoli.
3. Chinchillas and kittens are cute.	

- LDA is a way of automatically discovering the **topics** that these sentences contain
- Given these sentences and asking for 2 topics, LDA might produce:
  - **Sentences 1 and 2**: 100% Topic A
  - **Sentences 3 and 4**: 100% Topic B
  - **Sentence 5**: 60% Topic A, 40% Topic B
  - LDA learns words distribution per topic:
    - **Topic A**: 30% broccoli, 15% bananas, 10% breakfast, 10% munching, … (we could interpret topic A to be about *food*)
    - **Topic B**: 20% chinchillas, 20% kittens, 20% cute, 15% hamster, … (we could interpret topic B to be about *cute animals*)

Representation [Vosecky et al. 2014]

- Individual user Model (IM), step 2
  - For each tweet written by *u*, find (global LDA) topic, then compute the personalized terms distribution



Manchester: 2	Android: 1	Cake: 1
Play: 1	Coding:1	Apple: 1
Win: 1	Java: 1	Chocolate: 1

(from https://fr.slideshare.net/janvosecky/collaborative-personalized-twitter-search-with-topiclanguage-models)

Representation [Vosecky et al. 2014]

- Individual user Model, step 2
  - Assuming a topic k
    - Probability of word w for user u that wrote documents $\boldsymbol{D}_u$ (Max. Likelihood): (c(w,D) = tf of w in D written by u)

$$\theta^{IM}_{u,k,w} = \frac{\sum_{D:D\in\mathbf{D}_u \wedge z_D=k} c(w,D)}{\sum_{w'\in V}\sum_{D:D\in\mathbf{D}_u \wedge z_D=k} c(w',D)}.$$

  - Probability that user u chooses topic k

$$\theta^{IM}_{u,k} = \frac{|\{D : D\in\mathbf{D}_u \wedge z_D = k\}|}{|\mathbf{D}_u|}$$

Representation [Vosecky et al. 2014]

- Individual user Model, step 3
  - Assuming a topic *k*
    - Integration of unobserved words (smoothing by global topic model):

$$\theta^{I\hat{M}}_{u,k,w} = (1-\lambda)\theta^{IM}_{u,k,w} + \lambda P(w|\phi^{TM}_{k})$$

- Overall model with integration of topic choice:

$$\theta^{I\hat{M}}_{u,k,w} = (1-\lambda)\theta^{I\hat{M}}_{u,k,w}\theta^{IM}_{u,k} + \lambda P(w|\phi^{TM}_{k})\eta$$

η : prior probability of choosing a topic (a constant)

PIR – Usage of representation

- Document expansion
  - Use the profile words to expand documents
- Query expansion
  - Use the profile words to expand the query
- Personalized Matching
  - Integrate profile during the content-based matching
  - Reranking after non-personalized content-based matching

Documents expansion

- Not used… not scalable
  - Need to personalize each document d for each user u
    - A total of d × u personalized documents …
  - Not dynamic
    - For documents and users

Query expansion

- Difficult to expand the query without decreasing the quality of results…
- What terms of the profile to use ?
  - Terms that were co-tagged with the query terms [Mulhem et al. 2016]

$$q_{\exp} = q \cup \{w' \mid w' \in V, \exists w \in q; \exists d \in C, R(d,u,w) \wedge R(d,u,w')\}$$

with R(d,u,w) : user *u* tagged document *d* from corpus *C* with tag *w*

- Problems
  - How many terms, which weights for the expansion terms, …

Personalized Matching

- Integrate profile in matching expression
  - [Xu et al. 2008]

$$rsv(q,d,u) = \gamma.rsv_{content}(q,d) + (1-\gamma).rsv_{topic}(u,d)$$

    - Normalization questionnable (with BM25 for instance)
  - Difficult to control, but tractable dynamicity

Personalized Matching

- Reranking (most popular)
  - Process
    - Classical IR content-based matching (fast)
    - Reranking of the top-n documents in the result list (fast)
  - Pros:
    - we do focus, during the reranking, on already « potentially relevant » documents according to their content
    - We do not mix « apples » and « oranges » in the same step

[Vallet et al. 2010, Vosecky et al. 2014, Bouadjenek et al. 2013]

… Back to [Vosecky et al. 2014]

- Reranking using:

$$P(D,Q,u) \propto \left(\sum_{k=1}^{K} P(Q|\hat{\theta}^{IM}_{u,k,w}) P(D|\hat{\theta}^{IM}_{u,k,w}) \right) P(D)$$

  – with:
    - Similarity between user and query (for one topic *k*)

$$P(Q|\hat{\theta}^{IM}_{u,k,w}) = \prod_{w \in Q} P(w|\hat{\theta}^{IM}_{u,k,w})$$

    - Similarity between user and document (same as above for D)
    - P(D): Prior of document (may be constant, or popularity)

  – For efficiency: keep only "the" top topic for the query

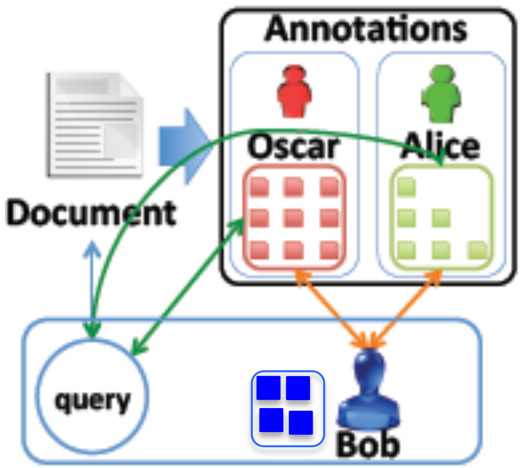
Socialized PIR

- Include social elements in personalization
  - « friends », followers, popular users…
  - Example: [Bouadjenek 2013]: SOPRA
    - Consider the tags of other users (VSM)

$$Rank(d,q,u) = \gamma \times \sum_{u_k \in U_d} \underline{Cos(\overrightarrow{p_{u_k}}, \overrightarrow{p_u}) \times Cos(\overrightarrow{p_u}, \overrightarrow{T_{u_k,d}})} + (1-\gamma)\times$$

$$\left[\beta \times \sum_{u_k \in U_d} Cos(\overrightarrow{p_{u_k}}, \overrightarrow{p_u}) \times \underline{Cos(\overrightarrow{q}, \overrightarrow{T_{u_k,d}})} + (1-\beta) \times \underline{Cos(\overrightarrow{q}, \overrightarrow{d})}\right]$$

  - U_d : set of users that annotated d
  - $T_{uk,d}$: tags of user u_k for d
  - p_u: user's profile for user u (all tags)
  - $\gamma \sim 0.6$, β=0.5



Social IR documents retrieval

- Kind of data
  - Documents, Tags, Users, Time
- The example of tweets
  - Vocabulary (abreviations, hashtags, mentions): « @Lesuperpanda @PlayHearthstone deck #SMOrc de @C4mlann avec 1 secret de chaque et les 2/1 chargeur divine pour 3. »

    …… about the game « Space Marine »…

Social IR documents retrieval

- Short documents: not classical with IR (remember the tf… still valid assumption?)
- Expand tweets to get more valuable information to apply IR
  - automatic hashtagging:

$$P(tag \mid post) = P(tag \mid topic).P(topic \mid post) \quad \text{[Si \& Sun. 2009]}$$

$$P(tag \mid post) = P(tag \mid word).P(word \mid post) \quad \text{[Ma et al. 2014]}$$

  - Wikification: putting tweets in context of Wikipedia pages
  - Use part of speech - example TweetNLP (next slide)

Social IR documents retrieval

- Part of speech - example TweetNLP (http://www.cs.cmu.edu/~ark/TweetNLP)

```
ikr smh he asked fir yo last name so he can add u on
fb lololol
```

!: interjection, G: abbreviation, O: pronoun, V: verb, P: pre/postposition; A: adjective, ^: proper noun

word	*tag*	*confidence*
ikr	!	0.8143
smh	G	0.9406
he	O	0.9963
asked	V	0.9979
fir	P	0.5545
yo	D	0.6272
last	A	0.9871
name	N	0.9998
so	P	0.9838
he	O	0.9981
can	V	0.9997
add	V	0.9997
u	O	0.9978
on	P	0.9426
fb	^	0.9453
lololol	!	0.9664

- *"ikr"* means "I know, right?", tagged as an interjection.
- *"so"* is being used as a subordinating conjunction, which our coarse tagset denotes *P*.
- *"fb"* means "Facebook", a very common proper noun (^).
- *"yo"* is being used as equivalent to *"your"*; our coarse tagset has posessive pronouns as *D*.
- *"fir"* is a misspelling or spelling variant of the preposition *for*.
- Perhaps the only debatable errors in this example are for *ikr* and *smh* ("shake my head"): should they be *G* for miscellaneous acronym, or *!* for interjection?

May be used to find out which terms to keep for IR…

Social IR documents retrieval

- Opinion mining
  - Finding trends for products or… elections for instance
- Event analysis
  - Get a broad view of a event according to the tweets
    - IR first, then deeper analysis for « smart presentation »
- Expert suggestion
  - Finding the « right » persons to follow about a given subject
    - A user is represented by its posts (+ popularity)

Conclusion

- Overview of some approaches for personalization
- Fast view of trends of IR on social networks data and problems
- TO KNOW :
  - Understand difficulties in IR personalization
  - Problems with microblogs retrieval

References

- M. Ghorab and D. Zhou and A. O'connor and V. Wade, Personalised Information Retrieval: Survey and ClassificationJournal of User Modeling and User-Adapted Interaction}, 23:4, 381--443, Kluwer Academic Publishers, 2013
- S. Jiang and Y. Hu and C. Kang, T. Daly, D. Yin, Y. Chang, C. Zhai, Learning Query and Document Relevance from a Web-scale Click Graph, SIGIR 2016, 185-194, 2016
- M. Bouadjenek, H. Hacid, M. Bouzeghoub, Sopra: a new social personalized ranking function for improving web search, SIGIR 2013, 861-864, 2013
- S. Xu, S. Bao, B. Fei, Z. Su, Y. Yu, Exploring folksonomy for personalized search, SIGIR 2008, 155-162, 2008
- D. Vallet, I. Cantador, J. Jose, Personalizing Web Search with Folksonomy-Based User and Document Profiles, ECIR 2010, 420-431, 2010
- J. Vosecky, K. Leung, W. Ng, Collaborative personalized Twitter search with topic-language models, SIGIR 2014, 53-62, 2014
- P. Mulhem, Philippe, N. Ould Amer, M. Géry, Axiomatic Term-Based Personalized Query Expansion Using Bookmarking System, DEXA 2016, 235-243, 2016

References

- X. Si, M. Sun, Tag-LDA for Scalable Real-time Tag Recommendation, Journal of Computational Information Systems 6(1) · November 2008
- Z. Ma, A. Sun, Q. Yuan, G. Cong, Tagging Your Tweets: A Probabilistic Modeling of Hashtag Annotation in Twitter, CIKM 2014, 999-1008, 2014