Integrating Tags in a Semantic Content-based Recommender

김연아

2021.03.24

- Introduction
- ITem Recommender
- User Generated Content (UGC)
- Experimental Evaluation
- Conclusion

Introduction

- **Idea**
  - Enables a **content-based recommender** to infer user interests by applying machine learning techniques both on the **"official" item descriptions** provided by a publisher, and on **tags which user adopt** to freely annotate relevant items

- **Key ideas used in this paper**
  - **Folksonomy (fork + taxonomy)** → taxonomy generated by users who collaboratively annotate and categorize resources of interest with freely chosen keywords, **Tags**
  - important to capture the **semantics** of the user interests often hidden behind keywords within static content and tags

- **Goal**
  - Does the integration of **tags** cause an **increase** of the prediction accuracy in the process of **recommending item to users**?

ITem Recommender

- **ITem Recommender (ITR)**
  - A content-based recommender system developed at the University of Bari
  - Three steps for recommendation process
  1. **Content analyzer**
  2. **Profile learner**
  3. **Recommender**

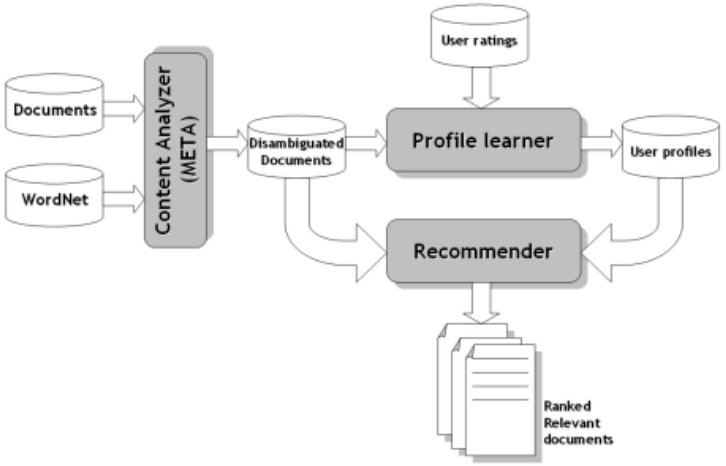


Figure 1: ITR architecture

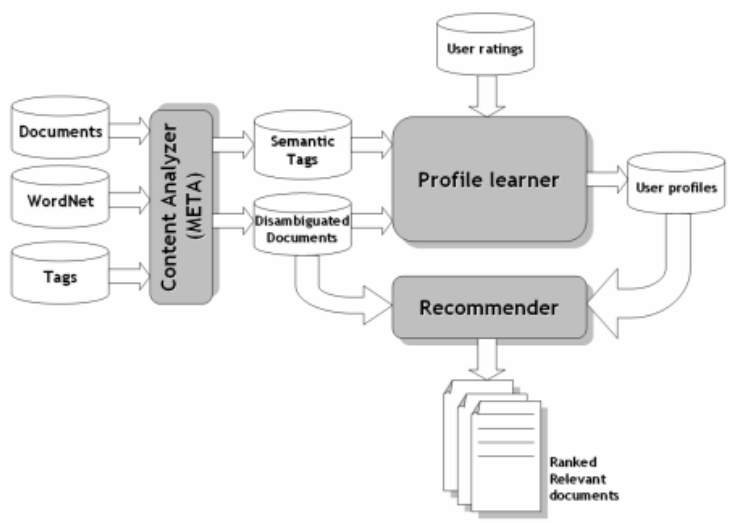


Figure 2: ITR 2.0 architecture

Semantic Indexing of Documents

- **WORDNET version 2.0**
  - Repository for word senses
  - Semantic indexing module

- **JIGSAW algorithm**
  - Automated procedure for assigning the proper sense to each word occurring in a document

  - Input: document d = $[w_1, w_2, \ldots, w_h]$
  - Return: a list of WORDNET synsets X = $[s_1, s_2, \ldots, s_k]$ $(k \leq h)$

    - Obtained by disambiguating the target word w based on the **semantic similarity** of w with the words in its context, that is a set of words that precede and follow w

  - Bag-of-synsets (BOS): Synset-based vector space representation

Semantic Indexing of Documents

- **JIGSAW algorithm**
  - **Bag-of-synsets (BOS):** Synset-based vector space representation
  - Item properties can be represented in form of **textual slots** → suggest potentially relevant items to users
    - *K*-th synset in slot *s* of document *d_n*

$$d_n^s = \langle t_{n1}^s, t_{n2}^s, \ldots, t_{nD_{ns}}^s \rangle$$

    - Weight of the synset *t_k* in the slot *s* of document *d_n*

$$f_n^s = \langle w_{n1}^s, w_{n2}^s, \ldots, w_{nD_{ns}}^s \rangle$$

Learning User Profile

- **Multivariate Poisson model for naïve Bayes text classification**
  - **Probability** of a document *d_j* belongs to a class *c* (user-likes or user-dislikes)
    → Calculated by Bayes' theorem

$$P(c|d_j) = \frac{P(d_j|c)P(c)}{P(d_j|c)P(c) + P(d_j|\bar{c})P(\bar{c})}$$
$$= \frac{\frac{P(d_j|c)}{P(d_j|\bar{c})}P(c)}{\frac{P(d_j|c)}{P(d_j|\bar{c})}P(c) + P(\bar{c})}$$

If we set:

$$z_{jc}^s = log\frac{P(d_j^s|c)}{P(d_j^s|\bar{c})} \quad (7)$$

then Eq. (6) can be rewritten as:

$$P(c|d_j) = \frac{\prod_{s=1}^{M} e^{z_{jc}^s} P(c)}{\prod_{s=1}^{M} e^{z_{jc}^s} P(c) + P(\bar{c})} \quad (8)$$

$$\lambda_{ic}^s = \frac{1}{|D_c|}\sum_{j=1}^{|D_c|}\hat{w}_{ij}^s \qquad \mu_{ic}^s = \frac{1}{|D_{\bar{c}}|}\sum_{j=1}^{|D_{\bar{c}}|}\hat{w}_{ij}^s \qquad s = 1, \ldots, M \quad (10)$$

where D_c ($D_{\bar{c}}$) is the number of documents in class c ($\bar{c}$),

$$\hat{w}_{ij}^s = \frac{w_{ij}^s}{\alpha \cdot avgtf^s + (1-\alpha)\cdot avgtf_j^s} \quad (11)$$

Learning User Profile

- **Multivariate Poisson model for naïve Bayes text classification**

  - **User rating**

  Like : (MIN + MAX) / 2 <= user rating

  Dislike : (MIN + MAX) / 2 > user rating

  - **Test**
    - Given a new document d_j, calculate **classification scores** $P(c_+|d_j)$ and $P(c_-|d_j)$

    → Classification scores for class c+ are used to produce a ranked list of potentially interesting items

User generated Content

- **Augmenting item recommender with user generated content (UGC)**
  - Include folksonomies in ITR by integrating **static** text describing items with **dynamic** UGC (tags)
  - Hybrid content-collaborative paradigm → Social tags + personal profile of a user

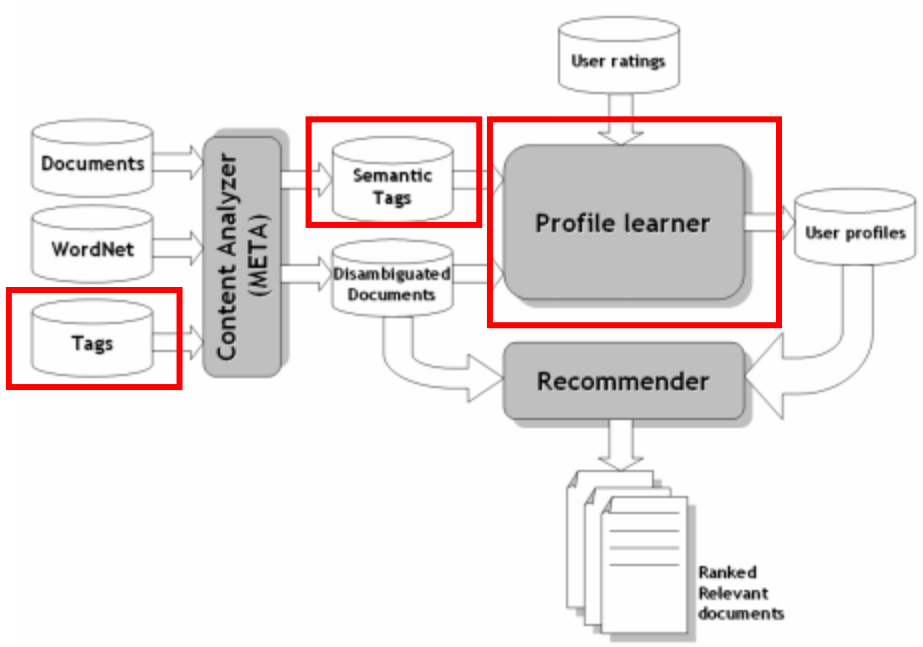


Figure 2: ITR 2.0 architecture

User generated Content

- **Different sets of Tags**
  - SocialTags(I): given an item I, the set of tags provided by all the users who rated I
  - PersonalTags(U,I): the set of tags provided by a specific user U on item I
  - PersonalTags(U): the set of tags provided by U on all items in the collection

- **Semantic tags**
  - Disambiguating tags in a folksonomy → synset-based folksonomy
    - Ex) **the set of synsets** obtained by disambiguating the set of tags provided by all user who rated item I

User generated Content

- **Profile Learner**

  Profile learning Process for user U

  1. Selecting all items (disambiguated documents) and corresponding rating provided by U

  2. Items → **positive(TR+) or negative(TR-)** training set depending on the rating

  3. **one-to-one user profile** (personal preference)

     For each document $d_j \in TR_+ \cup TR_-$, the additional slot is Semantic personal tags

     OR

     content collaborative profile

     For each document $d_j \in TR_+ \cup TR_-$, the additional slot is semantic social tags

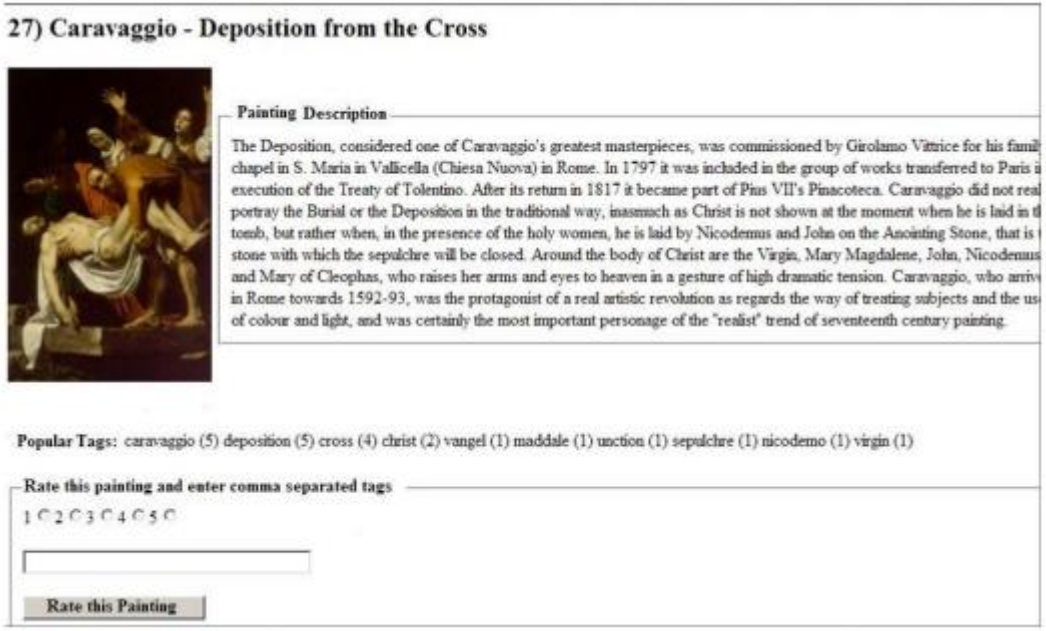
  → Each part of the profile is structured in four slots : 3 static & **1 dynamic (tags)**

Experimental Evaluation

- **Users and Datasets**
  - 45 paintings from gallery
    - Three textual properties → title / artist / description
  - 30 users (non-expert)
    - 5-point scaled rating on 45 paintings
    - Total of 4300 User-annotated tags

Table 1: Tag distribution in the dataset

Type of tags	Average
PersonalTags(U,I)	3.18
PersonalTags(U)	143.33
SocialTags(I)	95.55



Experimental Evaluation

- **Goal**

  1. Evaluate the predictive accuracy of ITR when **different types of content** are used in the training step

Conditions

Exp #1: STATIC CONTENT - only title, artist and description of the painting, as collected from the official website of the Vatican picture-gallery
Exp #2: SEMANTICPERSONALTAGS(U,I)
Exp #3: SEMANTICSOCIALTAGS(I)
Exp #4: STATIC CONTENT+SEMANTICPERSONALTAGS(U,I)
Exp #5: STATIC CONTENT+SEMANTICSOCIALTAGS(I)

Results

Table 2: Results of the K-fold Cross Validation

Type of content	Pr	Re	$F_{\beta=0.5}$
Static Content	75.86	94.27	78.94
SemanticPersonalTags(U,I)	75.96	92.65	78.80
SemanticSocialTags(I)	75.59	90.50	78.17
Static Content+SemanticPersonalTags(U,I)	**78.04**	93.60	**80.72**
Static Content+SemanticSocialTags(I)	**78.01**	93.19	**80.64**

Experimental Evaluation

- **Goal**

  2. Investigate which type of content produces the most accurate recommendations when a **small training set is available**

  Conditions

  - decide number **H** of folds to use for training → when H = 1, **9** training data are used

  Results

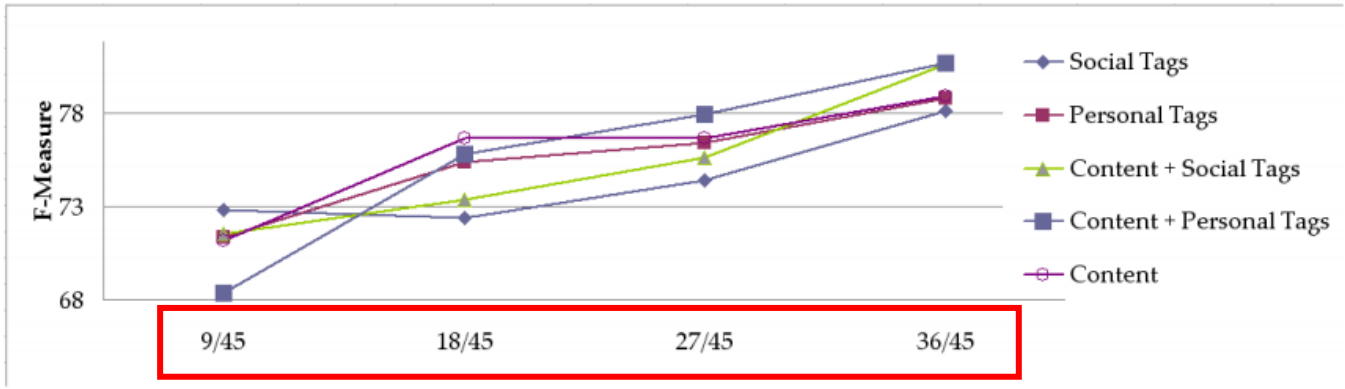


Figure 4: F_β Measure Learning Curve

Conclusion

1. **Hybrid strategy**

   - Use both static content and tags (dynamic) associated with items rated by users

2. **Personal + other users' tags**

   - Use personal tags and other users' tags (who rated the same items)

3. **Semantically interpreted tags using WORDNET**

   - Solution to freely chosen tags which usually have unclear meaning

Thank You