Multimed Tools Appl (2015) 74:1175–1197
DOI 10.1007/s11042-013-1840-y

Combining content with user preferences for non-fiction multimedia recommendation: a study on TED lectures

Nikolaos Pappas · Andrei Popescu-Belis

Published online: 6 March 2014

Abstract This paper introduces a new dataset and compares several methods for the recommendation of non-fiction audio visual material, namely lectures from the TED website. The TED dataset contains 1,149 talks and 69,023 profiles of users, who have made more than 100,000 ratings and 200,000 comments. The corresponding metadata, which we make available, can be used for training and testing generic or personalized recommender systems. We define content-based, collaborative, and combined recommendation methods for TED lectures and use cross-validation to select the best parameters of keyword-based (TFIDF) and semantic vector space-based methods (LSI, LDA, RP, and ESA). We compare these methods on a personalized recommendation task in two settings, a cold-start and a non-cold-start one. In the cold-start setting, semantic vector spaces perform better than keywords. In the non-cold-start setting, where collaborative information can be exploited, content-based methods are outperformed by collaborative filtering ones, but the proposed combined method shows acceptable performances, and can be used in both settings. For the generic recommendation task, LSI and RP again outperform TF-IDF.

Keywords Content-based multimedia indexing · Recommender systems · Multimedia recommendation · TED lectures

We are grateful for the support received for this work from the European Union through the inEvent project FP7-ICT n. 287872 (see http://www.inevent-project.eu).

N. Pappas (✉)
Idiap Research Institute and École Polytechnique Fédérale de Lausanne, Rue Marconi 19, 1920 Martigny, Switzerland
e-mail: nikolaos.pappas@idiap.ch

A. Popescu-Belis
Idiap Research Institute, Rue Marconi 19, 1920 Martigny, Switzerland
e-mail: andrei.popescu-belis@idiap.ch

1 Introduction

The recommendation of multimedia content to users can leverage either the content descriptors (*content-based methods, CB*) or information from the preferences of users (*collaborative filtering, CF*) or both types of information (*hybrid systems*). While in some domains, such as movie recommendation, content descriptors and user ratings are available on a large scale, such as in the Movielens data with millions of ratings, in other domains these can be scarce.

In this paper, we compare recommendation techniques for lecture recordings, that is, non-fiction audiovisual material with informative purposes, the content of which plays a significant part in deciding what to recommend. We compare the merits of CB and CF methods and propose a new method for combining semantic features (based on distances in semantic vector spaces) with user preferences (defined as the list of recordings explicitly marked as favorites, following common practice in recommender systems [43]). Following appropriate training to identify the best performing features, we show that CB recommendation using Explicit Semantic Analysis [13] is the best performing method in a cold-start setting, when no user preferences are known, including the case of anonymous viewers. In a non-cold-start setting, pure CF methods perform best, but only slightly above the combined CB and CF method with keyword-based distance, showing the importance of using content features in both settings.

The methods are tested on a new dataset acquired from the TED web-based repository of lectures on scientific and social topics. We show how this dataset can be used for the evaluation of lecture recommendations, given its rich content and metadata (to be used as features) along with explicit feedback from users (to be used as ground truth for training and testing). Our results thus constitute the first benchmark scores on this promising data set, which we made public.

The paper is organized as follows. We introduce the TED dataset and the metadata we extracted from it in Section 2. Then, we define the generic and personalized recommendation tasks that can be tested using this data in Section 3. We present semantic vector spaces in Section 4 and use them to define CB recommendation methods, as well as combined CB + CF ones, in Section 5. The results of feature selection are given in Section 6, while results over test data are given in Section 7 for personalized recommendations, and in Section 8 for generic ones, i.e. for anonymous users. Finally, in Section 9, we discuss our proposal in the light of the state of the art in multimedia recommendation using CB, CF and hybrid methods.

2 The TED collection: a dataset for recommendation evaluation

The TED website is the online repository of audio visual recordings of the popular TED lectures given by prominent speakers (see www.ted.com). The recordings and the metadata accompanying them are made available under a Creative Commons non-commercial license. The website provides extended metadata as well as user-contributed material such as discussion threads related to the talks. The TED speakers are scientists, writers, journalists, artists, and businesspeople from all over the world who are generally given a maximum of 18 minutes to present their ideas. The talks are given in English and are usually transcribed and then translated into several other languages by volunteer users. The quality and interest of the talks has made TED one of the most popular online lecture repositories. An important characteristic of TED is that the metadata for the audio visual content is human-made.

In Fig. 1 an example of a TED talk page is shown. On the left, the main audio visual player which displays the talk is at the top, just below the speaker's name and title of the talk. On the right, a short description of the talk is provided, along with the speaker's bio and the number of total views of the talk. Below the video player is the transcript of the talk, in a separate sub-frame that can be scrolled. To the right of the transcript, the TED website recommends to the user three talks that are related to the one that is currently displayed, which are presented as "what to watch next". The major part of the area below the player and the transcript is dedicated to the user comments, organized in threads.

2.1 Metadata structure and statistics

We crawled the TED dataset in April 2012 and gathered the metadata (excluding audio visual recordings) into two main entry types: talks and users. The talks have the following data fields: identifier, title, description, speaker name, TED event at which they were given, transcript, publication date, filming date, number of views. Each talk has user comments, organized in threads. In addition, we consider three metadata fields that were assigned by the TED editorial staff: related tags, related themes, and pointers to related talks (generally three per talk). For 95 % of the talks, a high-quality manual transcript is available. Table 1 provides the main statistics of the dataset, which includes 1,149 talks from 961 speakers.

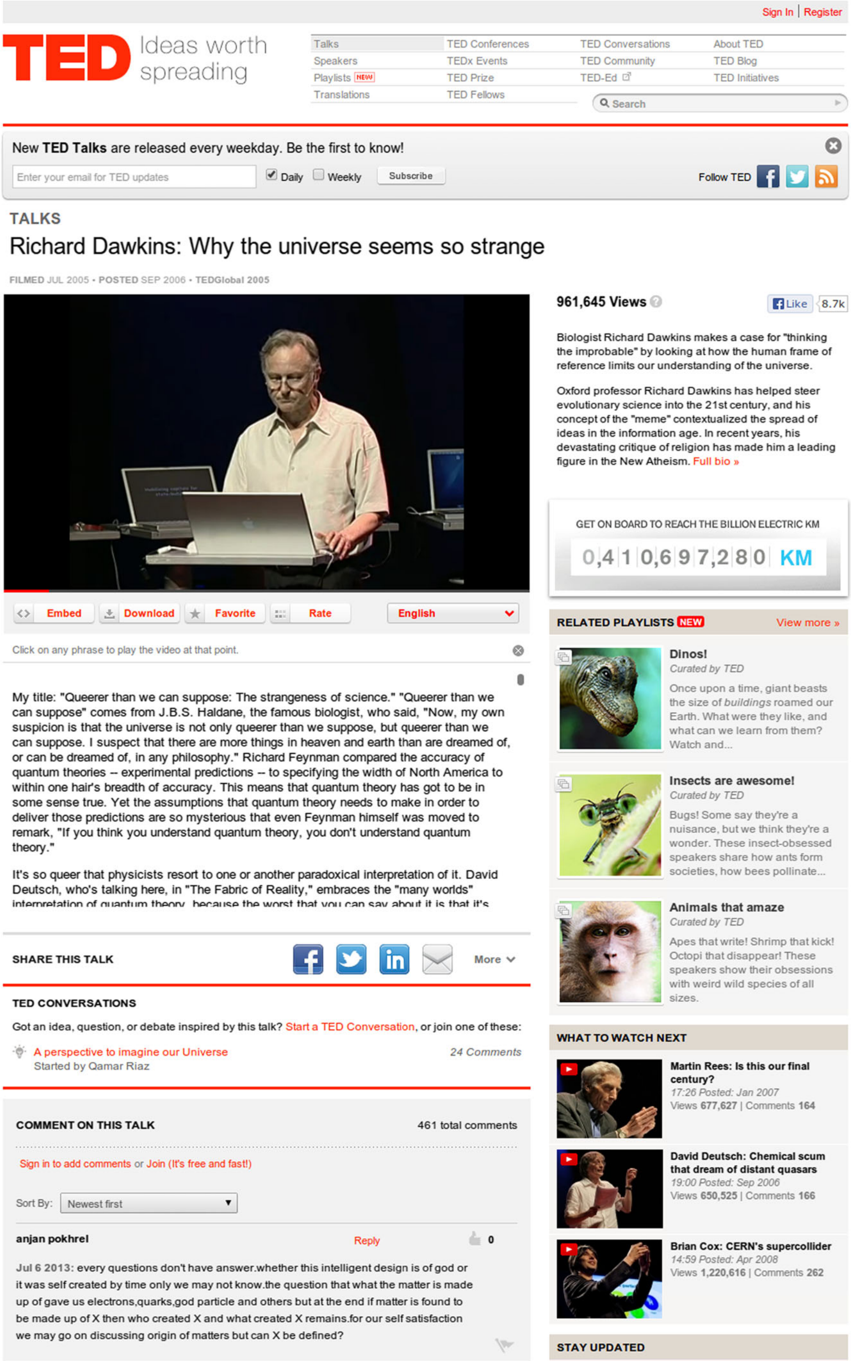
The *users* are the visitors of the TED website who have created an individual profile and have indicated a list of talks as public favorites. Although 69,023 users are registered, only 10,962 of them (i.e. 14 %) have explicitly indicated one or more favorite talks, and we will refer to them as *active users*, for reasons related to ground truth and evaluation which will be explained in the next section. Moreover, we will only use the subset of 2,427 users who have made 12 or more ratings each. This value strikes a balance between having enough ratings per user and enough users in the subset, according to standard practice in recommendation system evaluation. All lists of favorites (more than 100,000) and comments (more than 200,000) are included in the metadata set.

We made available the TED metadata set[1] under the same Creative Commons non-commercial license as the TED talks, and by permission of the TED website managers. The metadata, excluding audio and video signals, was acquired using two web crawlers developed with the Scrapy toolkit (from http://scrapy.org), one for the talks and one for the user profiles. The data was anonymized in the process, by replacing the public user IDs with hashes and discarding full names. With a polite rate of one request per second, the crawling lasted a couple of hours on April 27, 2012. The extraction of the attributes from talks and user profiles was done with hand-crafted patterns that exploit HTML attributes and CSS classes using the XPath query language.

2.2 Ground truth

The explicit user preferences in a given dataset constitute the ground truth which can be used for training and evaluating recommendation algorithms for *personalized recommendations*. A common form of such preferences are numeric ratings (e.g. from 1 to 5) that are assigned by users to items. In the TED dataset, the fact that a user has listed a talk among her favorite talks will count as the explicit preference. This corresponds to a binary numeric rating, coded as '1' for a favorite talk, and '0' for a talk not included in the list of favorites. The

[1] https://www.idiap.ch/dataset/ted/.



New TED Talks are released every weekday. Be the first to know!

Enter your email for TED updates ☑ Daily □ Weekly Subscribe

TALKS

Richard Dawkins: Why the universe seems so strange

FILMED JUL 2005 • POSTED SEP 2006 • TEDGlobal 2005

961,645 Views

Biologist Richard Dawkins makes a case for "thinking the improbable" by looking at how the human frame of reference limits our understanding of the universe.

Oxford professor Richard Dawkins has helped steer evolutionary science into the 21st century, and his concept of the "meme" contextualized the spread of ideas in the information age. In recent years, his devastating critique of religion has made him a leading figure in the New Atheism. Full bio »

Click on any phrase to play the video at that point.

My title: "Queerer than we can suppose: The strangeness of science." "Queerer than we can suppose" comes from J.B.S. Haldane, the famous biologist, who said, "Now, my own suspicion is that the universe is not only queerer than we suppose, but queerer than we can suppose. I suspect that there are more things in heaven and earth than are dreamed of, or can be dreamed of, in any philosophy." Richard Feynman compared the accuracy of quantum theories -- experimental predictions -- to specifying the width of North America to within one hair's breadth of accuracy. This means that quantum theory has got to be in some sense true. Yet the assumptions that quantum theory needs to make in order to deliver those predictions are so mysterious that even Feynman himself was moved to remark, "If you think you understand quantum theory, you don't understand quantum theory."

It's so queer that physicists resort to one or another paradoxical interpretation of it. David Deutsch, who's talking here, in "The Fabric of Reality," embraces the "many worlds" interpretation of quantum theory, because the worst that you can say about it is that it's

Dinos!
Curated by TED
Once upon a time, giant beasts the size of buildings roamed our Earth. What were they like, and what can we learn from them? Watch and...

Insects are awesome!
Curated by TED
Bugs! Some say they're a nuisance, but we think they're a wonder. These insect-obsessed speakers share how ants form societies, how bees pollinate...

Animals that amaze
Curated by TED
Apes that write! Shrimp that kick! Octopi that disappear! These speakers show their obsessions with weird wild species of all sizes.

TED CONVERSATIONS

Got an idea, question, or debate inspired by this talk? Start a TED Conversation, or join one of these:

A perspective to imagine our Universe 24 Comments
Started by Qamar Riaz

Martin Rees: Is this our final century?
17:26 Posted: Jan 2007
Views 677,627 | Comments 164

David Deutsch: Chemical scum that dream of distant quasars
19:00 Posted: Sep 2006
Views 650,525 | Comments 166

Brian Cox: CERN's supercollider
14:59 Posted: Apr 2008
Views 1,220,616 | Comments 262

COMMENT ON THIS TALK 461 total comments

Sign in to add comments or Join (It's free and fast!)

Sort By: Newest first

anjan pokhrel Reply 0

Jul 6 2013: every questions don't have answer.whether this intelligent design is of god or it was self created by time only we may not know.the question that what the matter is made up of gave us electrons,quarks,god particle and others but at the end if matter is found to be made up of X then who created X and what created X remains.for our self satisfaction we may go on discussing origin of matters but can X be defined?

Fig. 1 Presentation of a lecture on the TED website. The audio visual player (*top*) is followed by the transcript (in its own sub-frame) and by user comments (not shown here entirely), while on the right side is a short description followed by suggestions of related playlists and talks. (Screen shot from http://www.ted.com/talks/richard_dawkins_on_our_queer_universe.html used here for illustrative purposes only)

Table 1 Statistics for the TED data: total counts and averages ('avg') with standard deviations ('std') per talk, user and 'active user', for each of the attributes

Attribute	Total Count	Talk		User		Active user	
		Avg	Std	Avg	Std	Avg	Std
Talks	1,149	–	–	–	–	–	–
Speakers	961	–	–	–	–	–	–
Users	69,023	–	–	–	–	–	–
Active Users	10,962	–	–	–	–	–	–
Tags	300	5.83	2.11	–	–	–	–
Themes	48	2.88	1.06	–	–	–	–
Rel. Videos	3,002	2.62	0.74	–	–	–	–
Transcripts	1,102	0.95	0.19	–	–	–	–
Favorites	108,476	94.82	114.54	1.57	8.94	9.90	20.52
Comments	201,934	176.36	383.87	2.92	16.06	4.87	23.42

Active users are those who have indicated at least one favorite talk

latter case can mean two things: either the talk was not seen, or it was seen but was not liked. The ambiguity cannot be solved because viewing information for each profile is not available.

Therefore, we are not interested in predicting explicit rating values, but rather in ordering items according to the user's preferences as defined by favorite lists [43]. We should note that this evaluation is considerably different from conducting user studies to judge the performance of recommender systems and from modeling detailed user preferences recorded with ontology-based approaches [8, 25, 48]. The former, aside from the biases, is time-consuming and challenging. The latter is based on fine-grained semantic modeling of user preferences, but such models are difficult to construct and cannot be compared directly. Instead, modeling user preferences only based on individual properties (e.g. favorites, purchases) is typical of large-scale collaborative filtering systems and are helpful to compare the output of such systems. However, ontologies can be used to extract user and item features (see Section 9.1).

As the goal of our recommender system is to predict favorite videos, we will evaluate it, following common practice, by hiding some of the favorite talks of active users and measuring how well the system predicts them (comparison of system output with the ground-truth). For this measure, only the profiles of active users can be used, because for the others, no favorites are available. Moreover, personalized recommendation algorithms must be tested on user profiles that contain a sufficient number of ratings to serve as training data for each profile (see [14, 43]). This is why only active users with at least 12 favorites are kept in our experiments.

Things would be different if we tried to predict the commenting behavior, because this task is distinct from recommendation. In our view, commenting does not always signal positive interest—though it likely signals that the talk has been at least partly viewed—because the meaning of comments is uncertain: they may indicate that a talk was liked or disliked, or they may be mere replies to an argument from previous comments. Given that the goal of most recommender systems is to predict purchase, we consider that this is more closely mimicked by talks marked as favorites rather than just commented, and we did not experiment here with prediction of commenting behavior. However, we have shown elsewhere that the *polarity* of user comments can be used to augment rating information [33].

2.3 Distributions of user feedback

Figure 2 displays the distributions of favorites and comments in the TED dataset. The favorite talks are less sparse than comments, since the percentage of the former is higher than the percentage of the latter for the same percentage of items. In Fig. 3, the TED talks are displayed in a three-dimensional space, which shows more clearly the density of favorites and comments. The majority of the talks receive feedback from 1 to 500 unique users, with 1 to 250 favorites and 1 to 400 comments (including comments on comments, etc.). As explained above, in this paper, we use favorites as explicit ratings for training and testing, while noting that comments could be used as additional ratings on condition that their polarity is analyzed.

According to the well known long-tail distribution of rated items found in data from many commercial systems, the majority of ratings are condensed over a small fraction of the most popular items [2]. We examined the TED dataset to find out whether this property applied to its distribution of explicit ratings (favorites) as well, and found that 23 % of the ratings apply to the top 5 % of the items (short-tail) and the rest are distributed over the remaining set of 77 % less popular items (long-tail). Hence, the ratings in the TED dataset do follow a long-tail distribution, but it is less long-tailed than other distributions known in the literature: for instance, 33 % of ratings apply to the top 5.5 % movies in the Movielens dataset, and 33 % of the ratings apply to the top 1.7 % movies in the Netflix dataset. The fact that the distribution of ratings is less skewed, is likely due to the young age of the TED dataset (6 years old) and the slow rate of increase in talks.

A marked long-tail distribution may introduce a bias to the recommendation process since an algorithm which recommends only the most popular items may have good performance, but does not always bring benefits to the users because the recommendations may

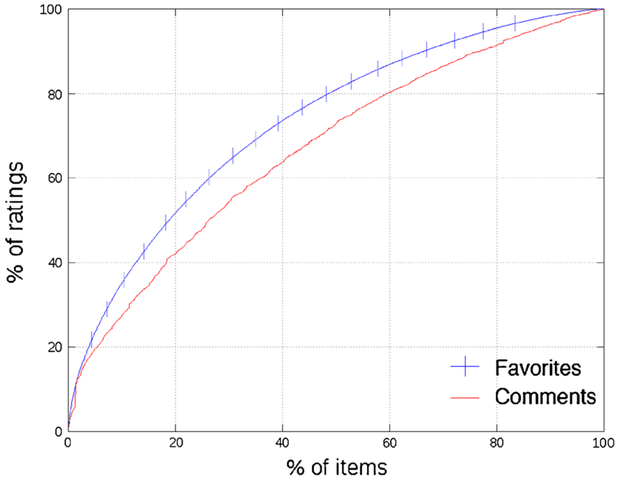


Fig. 2 Distributions of user feedback (favorites and comments). The percentage of items covered is on the *x*-axis and the percentage of ratings is on the *y*-axis

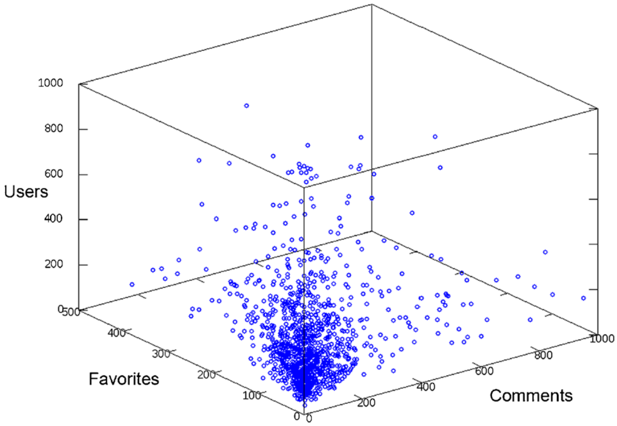


Fig. 3 Three-dimensional representation of the numbers of favorites and comments, and the unique users that made them for each talk, showing the skewed distribution of user feedback. The number of comments is on the x-axis, the number of favorites is on the y-axis, and the number of unique users that gave feedback is on the z-axis

not be novel to them, as shown in [7]. In the TED dataset, this effect should be less observed since the distribution of ratings is less long-tailed.

2.4 Comparison with other collections

The aforementioned properties of the TED data cannot be easily found in other alternative lecture repositories such as Khan Academy,[2] VideoLectures.NET,[3] YouTube EDU,[4] or Dailymotion[5]—as shown in Table 2, which compares various properties of these data sets. Khan Academy is an online learning community that contains more than 3,200 videos on scholarly topics. It shares some properties with TED in terms of providing transcripts and offering commenting capabilities, but it lacks descriptive fields, annotation with thematic tags and explicit feedback. Similarly, Video-Lectures.NET, Youtube EDU or Dailymotion do not provide transcripts and do not provide all the TED metadata fields. The dataset provided for the VideoLectures.NET recommender system challenge [3] includes the viewing history of the lectures as a ground truth for predicting future views of each lecture, along with content-related features, author and event information. However, information that is particularly useful for recommendation tasks such as explicit user feedback and detailed content information such as lecture transcripts is not made available.

The TED dataset thus appears as particularly valuable since it provides ground truth from explicit user preferences along with human-made recommendations, which are critical for evaluating, respectively, personalized and generic recommendation tasks. Besides, the dataset has been used for evaluating other tasks such as automatic speech recognition and machine translation [12].

[2] http://www.khanacademy.org/.

[3] http://www.videolectures.net/.

[4] http://www.youtube.com/education/.

[5] http://www.dailymotion.com/.

Table 2 Comparison of TED with other repositories in terms of available metadata and user feedback

Collection	Basic	Speaker	Trs.	Tags	Implicit	Explicit	CC
VideoLectures	✓	✓	✓		✓		
KhanAcademy	✓	✓			✓		
Youtube EDU	✓		✓		✓	✓	
DailyMotion	✓				✓	✓	
TED	✓	✓	✓	✓	✓	✓	✓

The properties are: **Basic**: Title and Description, **Speaker**, **Transcript**, **Tags**: Categories in form of keywords, **Implicit**: Implicit feedback (e.g. comments or views), **Explicit**: Explicit feedback (e.g. ratings, favorites or bookmarks), and **CC**: Creative Commons Non-Commercial License

3 Definition of recommendation tasks

In this section, we specify two complementary recommendation tasks that can be evaluated using the TED dataset, namely a personalized and a generic one. The first one considers the global history of each user (embodied in the list of favorites) to recommend new content of interest, while the second one aims at recommending content that is related to a given talk, regardless of the user watching it. Of course, a combined task could also be defined, in which a given user watching a given talk receives further recommendations—an instance of context-aware recommendation [1]. However, the available TED metadata does not offer ground-truth data to evaluate such a task, though it could be derived using additional assumptions (such as chronological ordering or topical clustering) which are beyond the scope of this paper.

3.1 Personalized recommendations

Given a set of binary ratings as a ground truth, the goal of the personalized recommendation task is to predict whether unseen items will be interesting or not for the users [43], or more simply to predict the N most interesting ones (top-N recommendation task [7], also known as one-class collaborative filtering task [30]). Such problems are particularly challenging due to the fundamental uncertainty of the '0' class. In such a scenario of offline prediction, the recommendation models are classically trained on fragments of user's histories, and evaluated by hiding some of the preferred user items and then trying to predict them. The performance is evaluated using classification accuracy metrics.[6]

For the TED dataset, we suggest that for each user u in the set of users U (or a subset of it, such as users having made more than a number of ratings, as in Section 2.1), her ratings (favorites) are randomly split into training and test sets, noted M and T, typically 80 % vs. 20 %. A recommendation model is trained (possibly with cross validation) on M, and then tested on the held-out set T by comparing its output R with the actual ratings of user u over T.

[6] The scenario of this task does not presuppose that the user is currently viewing a talk, but considers only the user's past history. As a consequence, if a user is interested in several different topics, it is likely that in the resulting recommendations each topic will be present with its probability of appearance in the user's past history. On the contrary, in a contextual recommendation task as mentioned above, the topic of the talk that is currently viewed should be considerably boosted with respect to the others in the resulting recommendations.

3.2 Generic recommendations

The generic or user-independent recommendation task corresponds to scenarios in which the users' history of ratings is absent, e.g. for anonymous users. The goal of this task is to predict the most similar items to a given one, which can also be seen as a non-personalized top-N recommendation task. Given the set of human-made, user-independent recommendations for each item in a dataset—the three related videos (or "what to see next") for each TED talk—a model can be trained and evaluated using this information as ground truth, ignoring user preferences or the talks previously viewed. Again, the set of items I can be split into a training set M and a testing set T for evaluation.

3.3 Evaluation metrics

For the top-N personalized recommendation task, error metrics such as RMSE are not the most appropriate ones, since a top-N recommender is not necessarily able to infer the exact rating of a user $u \in U$ for any item $i \in I$ [7]. Instead, this task can be evaluated more informatively by using the classification accuracy metrics of precision, recall and f-measure (see [43]). Precision and recall at N are respectively given by:

$$P(N) = \frac{1}{|U|} \sum_{u \in U} \frac{|T_u \cap R_{u@N}|}{N}; \; R(N) = \frac{1}{|U|} \sum_{u \in U} \frac{|T_u \cap R_{u@N}|}{|T_u|}, \tag{1}$$

where N is the bound of top recommendations, $|U|$ is the total number of users in U, T_u is the set of items in user's u history and $R_{u@N}$ are the top-N recommendations of the model for the user u. Recall is computed by dividing by the number of items in user's u history, $|T_u|$, instead of N. The F-measure is the harmonic mean $F(N)$ of $P(N)$ and $R(N)$, which is computed as $2 \cdot (P \cdot R)/(P + R)$.

Similarly, applying (1) directly to the items I in a test set T, we obtain the definitions of precision and recall for generic recommendations as follows:

$$P(N) = \frac{1}{|I|} \sum_{i \in I} \frac{|T_i \cap R_{i@N}|}{N}; \; R(N) = \frac{1}{|I|} \sum_{i \in I} \frac{|T_i \cap R_{i@N}|}{|T_i|}, \tag{2}$$

where T_i are the ground-truth items related to i, R_i are the recommended items for i and the remaining variables are defined as above.

4 Semantic vector space models

Content-based recommender systems use similarity measures between items that rely on their content descriptors. Here, we investigate semantic vector space models (VSM) to define such similarities, and later in Sections 7–9 we compare their merits for recommendation over the TED dataset. Benchmarking these models is a contribution to the ongoing debates on semantic-based approaches to recommendation [22]. Semantic VSMs are considered to be able to reduce the effect of the curse of dimensionality and data sparseness of standard VSMs, such as those based on TF-IDF weighting [37]. The proximity of two vectors in the semantic space (usually computed with cosine similarity) can be interpreted as a measure of the semantic relatedness between the objects that are represented by those vectors, which can then be used to model user preferences in recommendations tasks.

When using a VSM, each document d_i is represented as a feature vector $(w_1, w_2, \ldots, w_{ij})$, where each position j corresponds to a word of the vocabulary V. The weights w_{ij} can be computed using various models: Boolean values ('1' if the document contains the word, '0' if it does not), counts of words, term frequencies, inverse document frequencies, or TF-IDF coefficients. For example, TF-IDF is computed as follows: $w_{ij} = tf_{ij} \cdot idf_j$, where tf_j is the term frequency of word j in document d_i and idf_j is the inverse document frequency of word j. The TED talks, noted as items I, can thus be represented by creating vectors of words from their text attributes, which can be pre-processed to remove stop words or to apply stemming. In our experiments we performed the following pre-processing steps:

I → Tokenization → Stop words removal → Stemming → V

There are several techniques in the literature for creating semantic representations in VSMs. In our experiments, we consider a VSM with TF-IDF as the baseline weighing model [38] and four representative semantic VSMs from the three main existing categories, as follows: (1) two dimensionality reduction methods, namely Latent Semantic Indexing (LSI) [15] and Random Projections (RP) [36]; (2) a topic modeling approach, namely Latent Dirichlet Allocation (LDA) [6]; and (3) a concept space based on external knowledge, namely Explicit Semantic Analysis (ESA) [13]. These techniques have generalization capabilities, as they project the data from the original vector space to a topic or concept space with a reduced number of dimensions – apart from ESA which actually augments the dimensionality to the number of Wikipedia concepts. In terms of free parameters, LSI, RP and LDA rely on the number of topics t (latent factors). Moreover, LDA relies on two parameters traditionally noted α and β for the Dirichlet priors of topic and word distributions.

For the implementation of LSI, RP and LDA, we used the Python Gensim library [34], while for ESA we used the Wikipre-ESA Python implementation of the method described in [13], over a 2005 snapshot of Wikipedia.

5 Recommendation algorithms

We benchmark on the TED data two main types of recommendation methods, namely content-based and collaborative filtering ones, using item-based similarity [31] in both cases.

5.1 Content-based algorithms

For content-based methods, we first pre-compute an item similarity matrix for each of the VSMs above, noted respectively S_{TF-IDF}, S_{LSI}, S_{RP}, S_{LDA} and S_{ESA}. Each matrix S is an $m \times m$ matrix, m being the number of talks. The value of each element s_{ij} of each S is the cosine similarity of the vectors representing items i and j in the given VSM.

We then define a ranker based on content similarities, noted as *CB*. Given a similarity function that outputs a score for two items (two TED talks), *CB* recommends to a user u a list of ranked items based on the k most similar items to those already known to be her favorites from the training data M_u. Therefore, *CB* recommends items to user u based on their estimated relevance $\hat{r}_{ui}$ defined as:

$$\hat{r}_{ui} = \sum_{j \in D^k(u;i)} s_{ij} \tag{3}$$

where $D^k(u; i)$ are the k most similar items from I to the ones in the training set of the user M_u and s_{ij} is the similarity between items i and j according to one of the five matrices S. The summation is limited to a set of k neighbors only $(D^k(u; i))$ principally for tractability or efficiency reasons.

5.2 Collaborative filtering algorithms

For collaborative filtering methods, we first pre-compute the item similarity matrices based on the common ratings between pairs of items in the user-item matrix (built from the training set) by using two common metrics, namely Pearson correlation yielding the S_{PC} matrix (as in [24]), and cosine similarity yielding the S_{COS} matrix (as in [7]) as follows:

$$S_{COS_{ij}} = \frac{\mathbf{i} \cdot \mathbf{j}}{||\mathbf{i}||_2 \times ||\mathbf{j}||_2}; \; S_{PC_{ij}} = \frac{E[(\mathbf{i} - \mu_i)(\mathbf{j} - \mu_j)]}{\sigma_i \sigma_j}, \tag{4}$$

where $\mathbf{i}$ and $\mathbf{j}$ are the feature vectors of items i and respectively j derived from the item-item co-rating matrix (or, in other formulations, from the user-item matrix, where each item is represented by a vector of user ratings).

Then, we use a neighborhood model defined in (5), which is commonly used for collaborative filtering. The prediction function $\hat{r}_{ui}$ estimates the rating of a user u for an unseen item i, based on the bias estimate b_{ui} of u for item i, computed using (7), and on a score that is calculated from the k most similar items to i (according to either S_{PC} or S_{COS}) which the user u has already rated, i.e. the neighborhood $D^k(u; i)$ as above. The denominator ensures that the predicted ratings will fall in the same range of values as the known ones.

$$\hat{r}_{ui} = b_{ui} + \frac{\sum_{j \in D^k(u;i)} d_{ij}(r_{uj} - b_{uj})}{\sum_{j \in D^k(u;i)} d_{ij}} \tag{5}$$

The term r_{uj} is the rating value of a user u for a given item j. The coefficient d_{ij} expresses the similarity between item i and item j, and is computed as in (6) by using the similarity s_{ij} between items i and j multiplied by a factor varying from 1 (when the number of common raters n_{ij} is considerably larger than λ) to 0 (when n_{ij} is considerably smaller than λ). Typically, $\lambda \approx 100$.

$$d_{ij} = s_{ij} \frac{n_{ij}}{n_{ij} + \lambda} \tag{6}$$

The bias estimate b_{ui} is the sum of the average ratings μ of items in the dataset, the average of the ratings of a user u, noted b_u, and the average of the ratings for a given item i, noted b_i, as shown in (7):

$$b_{ui} = \mu + b_u + b_i \tag{7}$$

We consider two representative variants of this model. First, we use a normalized neighborhood model (as defined in (5)) with Pearson Correlation for vector similarity; this model is noted as *CF(PC)*. Second, we use a non-normalized model, noted with a preceding ‘u’ for ‘unnormalized’, obtained by removing the denominator in (5) and using the cosine similarity distance, hence this model is referred to as *uCF(COS)*. In previous studies [7], non-normalized models were found to perform better for the top-N recommendation task than normalized ones.

5.3 Combining collaborative filtering with content similarity

We incorporate in the neighborhood model presented above information about content-based similarity, by replacing in (5) the d_{ij} similarity with the content-based one from (3)), and using the non-normalized version. Hence the estimated rating in the combined model is:

$$\hat{r}_{ui} = b_{ui} + \sum_{j \in D^k(u;i)} s_{ij}(r_{uj} - b_{uj}) \tag{8}$$

This new model allows us to exploit at the same time the semantic-based similarities and the bias estimate, therefore to combine the two types of information, content and collaborative. This is especially useful when collaborative information is sparse, and the similarity computed using it is less reliable than the content-based one.

We consider only the non-normalized versions of the model, noted again with ‘u’, and indicate the type of content-based similarity that is used in combination to the CF neighborhood model. Hence, these combined models are referred to as *uCF*(*TFIDF*), *uCF*(*LSI*), *uCF*(*LDA*), *uCF*(*RP*) and *uCF*(*ESA*).

For comparison purposes, we finally consider a user-independent recommender noted *TopPopular*, which always recommends the items with the highest popularity, based on the total number of ratings, regardless of a user’s preferences.

6 Parameter and feature selection

We determine the optimal parameters and features of the content-based methods using 5-fold cross-validation over the training set *M*, which includes 80 % of the ratings for each of the 2,427 TED users that have made 12 or more ratings. The remaining 20 % of the ratings from these users are kept as an unseen test set *T*, which is used in Section 7.

The CB methods use lexical features (word vectors) extracted from one or more fields of each TED talk, represented schematically in Fig. 4, and several meta-parameters for each of the semantic representations (TF-IDF, LSI, RP, LDA, and ESA) as described in Section 4. Exploring all possible combinations of features to find out which subset performs best is not tractable. Therefore, we grouped individual features into four groups: title and description (TIDE), related tags and themes (RTT), transcript (TRA), and speaker plus TED event (TESP). Along with all individual features, we tested these sets, and all their combinations, organized as in Fig. 4.

For LSI and RP we optimized the values of the parameter t (number of topics) by varying it from 10 to 5,000 and for LDA from 10 to 200 only, for tractability reasons. Additionally, for LDA, we varied the α and β parameters from 0 to 1, and the optimal ones were found to be $\alpha = 1$ and $\beta = 0.002$. We fixed the neighborhood size at $k = 3$, which is a trade-off between computational cost and expected prediction accuracy [19].

Figure 5 displays the ranking of features and their combinations, ordered by the average f-measure (F@5) over *all* the tested methods (i.e. TF-IDF, LSI, RP, LDA, and ESA) and all the parameters of methods stated in the previous paragraph. These results thus indicate which features perform well over *all* methods, as opposed to features that are optimal for *each* method, which will be shown below. As seen on the standard deviations obtained from cross-validation and averaging over the five methods (segments over the bars in Fig. 5),

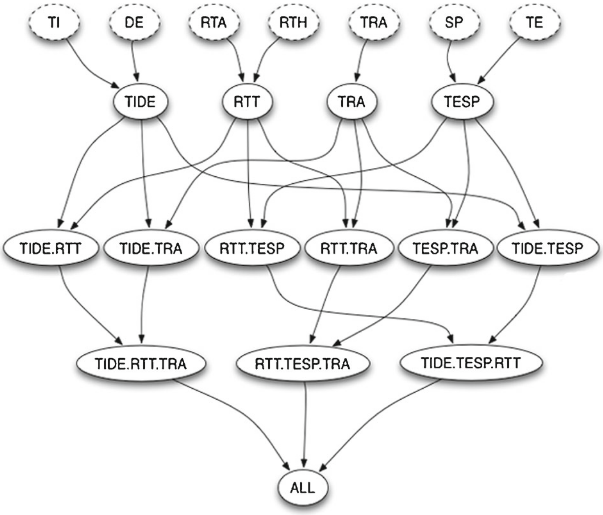


Fig. 4 Combinations of features for comparison. Atomic features are title (TI), description (DE), related tags (RTA), related themes (RTH), transcript (TRA), speaker (SP) and TED event (TE)

the non-overlapping segments indicate important differences between single or composite features. For instance, the four top-level features are clearly better than the four bottom ones.

The results show that the human-made description of talks (DE), the title (TI), and their combinations with other features (TIDE, TIDE.RTT, and TIDE.TESP.RTT) are the most useful features on average for content-based personalized recommendations. In addition, knowledge of the speaker (SP) is useful too (ranked sixth). However, these metadata fields come to a cost because they must be entered by the editors of the lecture repository. The

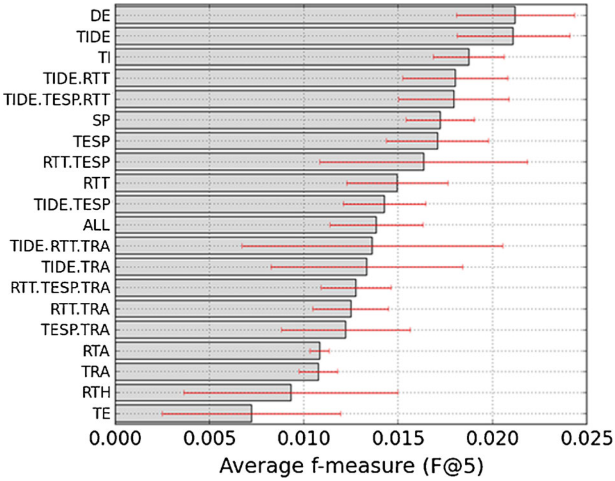


Fig. 5 Ranking of individual and combined features based on the decreasing average of f-measure over all five methods. Atomic features are title (TI), description (DE), related tags (RTA), related themes (RTH), transcript (TRA), speaker (SP) and TED event (TE). The segments over the bars represent the standard deviations when averaging over 5-fold cross-validation and five methods with all tested parameters

Table 3 Optimal features for content-based methods found using 5-fold cross-validation on the training set

Method	Optimal features	Performance (%)		
		P@5	R@5	F@5
LDA ($t = 200$)	Title, description, TED event, speaker (TIDE.TESP)	1.63	1.96	1.78
TF-IDF	Title (TI)	1.70	2.00	1.83
RP ($t = 5000$)	Description (DE)	**1.83**	**2.25**	**2.01**
LSI ($t = 3000$)	Title (TI)	**1.86**	**2.27**	**2.04**
ESA	Title, description (TIDE)	**2.79**	**3.46**	**3.08**

Scores in bold are significantly higher than TF-IDF ones and ESA is significantly above RP and LSI (pairwise t-statistic, $p < 0.05$)

description, in particular, requires a significant human effort, though it is likely that TED presenters write their own descriptions.

The lowest performing features were the name of the TED event (TE) and the related themes assigned by TED editors (RTH), which presumably lack specificity for recommendation. In fact, the related themes were recently removed from the TED website, keeping only the related topics assigned by TED editors (a different and more relevant feature). Somewhat surprisingly, the transcript (TRA) decreases the performance of all methods and most of the combinations that include it are in the middle of the ranking. One possible explanation is that the huge size of the transcript's vocabulary introduces a lot of noise.

Table 3 shows the optimal features and parameters for each semantic representation used with *CB*, together with the scores (precision, recall and f-measure at 5) that they enable the recommender system to reach (5-fold cross-validation on the development data). All the semantic-based methods except LDA outperform significantly the TF-IDF baseline (pairwise t-statistic, $p < 0.05$): 11 % improvement for LSI, 7.6 % for RP and up to 64 % by ESA, which reaches the best score. While two semantic-based methods (LSI and RP) perform without significant differences, ESA is significantly above them (pairwise t-tests, $p < 0.05$). The performance of ESA shows that the external-knowledge-based representation of the items is significantly more useful to our task than the domain knowledge captured intrinsically by the other methods.

7 Personalized lecture recommendation

In this section, we compare recommendation performance of CB, CF and combined methods over the held-out test set T, considering two different settings: (i) a cold-start setting where the collaborative rating information for the items is not available and (ii) a non-cold-start setting where it is. Note that when testing, we only hide the rating information for the user currently tested, but use the information from the other users to make our recommendations, following current practice in the field.

7.1 Cold-start recommendations (CB methods only)

The cold-start setting is characterized by sparse user ratings, with many items not having been rated at all, which makes it impossible for CF methods to recommend these items (e.g.

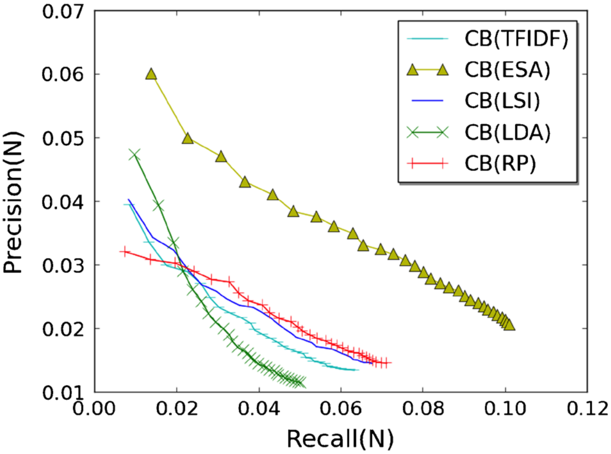


Fig. 6 Scores of content-based methods in a cold-start setting, in terms of precision and recall at N ($1 \leq N \leq 30$) on the held-out set T. The ESA-based distance outperforms by far all the others

new TED lectures). In such a situation, only content-based methods can help making recommendations. In Fig. 6, we show the performance of our CB methods in terms of precision and recall over the held-out set T. Most of the semantic-based representations perform significantly better (t-statistic, $p < 0.05$) than TF-IDF, with +62 % for ESA, +7 4% for LSI and +8 % RP. LDA does not improve over TF-IDF (as also seen in Table 3) except at the top 1 to 4 recommendations; it was also the most difficult method to tune.

The scores obtained appear to be overall quite small, though in line with previous work (see [7, 30] and Section 9.3). These scores must be interpreted in the light of the following two facts. Firstly, the probability of having the correct item ranked by chance first (P@1) among 1,149 candidates is only 0.08 %, while our *lowest* score (for Random Projections) was 40 times higher at 3.20 % (Fig. 6). Moreover, the precision of random guessing decreases dramatically at higher ranks (e.g. P@5). Secondly, we consider here only the positive ratings (favorites) to calculate precision, and discard the scores of unseen items, which would have a much higher baseline.

The improvement brought by ESA appears to be again much greater than that of LSI and RP, allowing us to conclude that similarity based on concept spaces from external knowledge captures more effectively the content similarity and, consequently, the user preferences than the other semantic spaces and the baseline TF-IDF. Semantic-based approaches are thus more effective than keyword-based ones for cold-start personalized recommendations.

7.2 Non-cold-start recommendations (all methods)

In a non-cold-start setting, where the items have been rated by many users, the collaborative filtering information and the bias introduced by the popularity of items can be specifically exploited. As the CB methods do not have such information, their performance was found to be lower than that of CF methods, and will not be reported here. However, the combinations of CB and CF proposed in Section 5.3 (noted $uCF(\cdot)$ with '$\cdot$' indicating the similarity method) allow content-based similarity to take into account the bias estimate, and their results are only slightly below pure CF methods in the non-cold-start scenario, while being operational both in cold-start and non-cold-start settings.

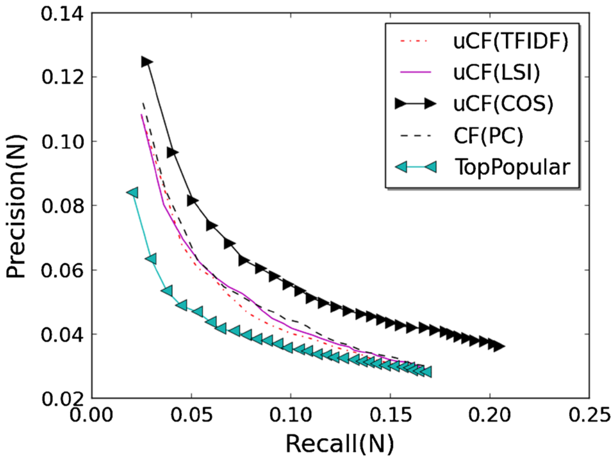


Fig. 7 Lecture recommendation scores for two collaborative filtering methods, *CF*(*PC*) and *uCF*(*COS*), and two combined methods namely neighborhood with TFIDF, uCF(TFIDF), and LSI distances, uCF(LSI), in a non-cold-start setting. Precision and recall at $1 \leq N \leq 30$ are computed on the held-out test set *T*. Collaborative filtering using cosine similarity in a neighborhood model scores highest, but the combined model using neighborhoods and TFIDF is not far behind

Figure 7 displays the performance of two neighborhood models used for collaborative filtering: the normalized one using Pearson Correlation (*CF*(*PC*)) and the unnormalized one using cosine similarity (*uCF*(*COS*)). We also represent the two best performing combined methods, unnormalized, using TFIDF and LSI distances (*uCF*(*TFIDF*) and *uCF*(*LSI*)), as well as the *TopPopular* baseline. The best performance is achieved by the non-normalized neighborhood model with cosine similarity, *uCF*(*COS*) (+34 % on average with respect to *TopPopular* over all data points in Fig. 7). The *CF*(*PC*) model is slightly below it, but is still significantly better than *TopPopular* (+15 %). The CB methods have insignificant differences with each other and with *uCF*(*PC*). All these comparisons are based on pairwise t-tests over the values of the P-R curves from 1 to 30.

The combined models, *uCF*(*TFIDF*) and *uCF*(*LSI*), perform similarly to *CF*(*PC*) and are also significantly better (t-statistic, $p < 0.05$) than *TopPopular*, respectively +10.5 % and +13 % above it. The other content-based similarities (RP, LDA, ESA) perform slightly below TF-IDF, but the difference is not statistically significant. Using the bias introduced by the item popularity thus decreases the difference in performance between the content-based similarity models, i.e. *uCF*(*LSI*) and *uCF*(*TFIDF*), compared to their differences in the cold-start setting.

8 Generic recommendations

The goal of generic or user-independent recommendation is to predict items that are related to a given one, without any knowledge of user profiles. We use here unsupervised methods, namely rankers based on content similarities, defined in Section 5. As a ground-truth, we use the human-made lists of related videos that are available in the TED data set. In most of the cases, TED editors have indicated three related talks for each talk, or sometimes fewer.

Using classification accuracy metrics (F1-score), we evaluate various content-based rankers, namely semantic-based and keyword-based ones, through their overlap with the ground-truth ranking. Table 4 shows that, similarly to personalized recommendations, the LSI and RP semantic-based methods significantly outperform the keyword-based one using TF-IDF and the other methods as well (pairwise t-test on 5-fold c-v., $p < 0.05$). However, the difference between LSI and RP is not significant. The parameters of the methods were set to the optimal values found for the personalized recommendation task in Section 7, which means the results that are obtained from these rankers might be even improved if we optimize them for the generic task. Results might also improve when supervised methods (rather than unsupervised ones) are used for learning to rank, such as SVM-Rank [17]. The main conclusion at this stage is that the semantic information is beneficial over keyword-based only methods for generic recommendation, as it was for personalized recommendation.

Figure 8 displays the ranking of features and their combinations, ordered by the average f-measure (F@3) for TF-IDF content-based ranker. The ranking of the features for this task is quite different from the one for personalized recommendations (displayed in Fig. 5 above). For generic recommendations, the combination of all features appears to be the second best performing set of features, while the set that actually performs best is RTT.TESP, which includes the related tags and themes, the speaker and the TED event. These sets were ranked in the middle for the personalized recommendation task. When considered independently, the related themes (RTH) and the TED event (TE) fields rank very low (respectively 19th and 20th), while the other two features, namely the related tags (RTA) and the speaker (SP) have have relatively low rank as well (respectively 15th and 17th). We presume that when put together, these features capture complementary properties, because their combination leads to the best recommendation performance. Note that the combination of some other fields does not lead to improvement, implying that they capture overlapping properties, for example description (DE) compared to title plus description (TIDE). A possible explanation for these differences is that individual user preferences in the personalized task are more difficult to capture than the preferences of the TED editors which defined the related talks used as ground-truth for generic recommendations.

Table 4 Evaluation of unsupervised methods (content-based rankers) for generic recommendation, in terms of overlap with the related talks recommended by TED editors (first line)

Methods	TED	TopPopular	TF-IDF	LSI	RP	LDA	ESA
TED	1.000	0.006	0.129	0.156	0.143	0.091	0.124
TopPopular	–	1.000	0.003	0.003	0.004	0.004	0.006
TF-IDF	–	–	1.000	0.510	0.323	0.195	0.523
LSI	–	–	–	1.000	0.419	0.220	0.442
RP	–	–	–	–	1.000	0.200	0.299
LDA	–	-	–	–	–	1.000	0.193
ESA	–	–	–	–	–	–	1.000

The matrix provides also the overlap values between all methods for comparison purposes, showing for instance that ESA and LSI provide the most similar recommendations to TF-IDF (0.523 and 0.510). The metric is the f-measure, and underlined scores are significantly higher than TF-IDF ones (pairwise t-statistic: $p < 0.05$)

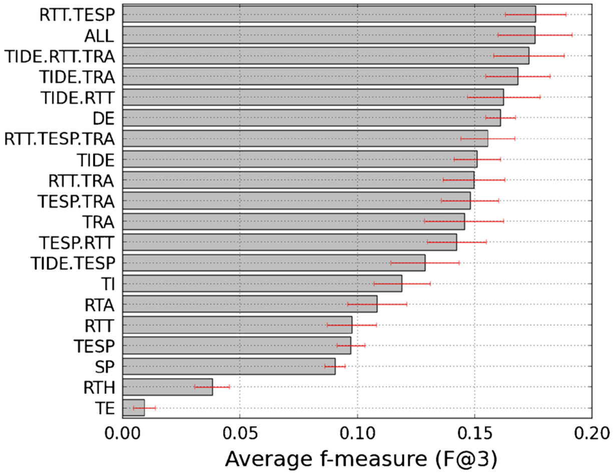


Fig. 8 Ranking of atomic and combined features (see combinations in Fig. 5) based on the decreasing average f-measure for TF-IDF similarities. The atomic features are: title (TI), description (DE), related tags (RTA), related themes (RTH), transcript (TRA), speaker (SP) and TED event (TE). The segments over the bars are the standard deviations from 5-fold cross-validation, with non-overlapping intervals indicating important differences

9 Related work on recommender systems

In this section, we review previous research work in line with the present study, related to top-N recommendation and to content analysis. We begin with recommendation methods that incorporate semantic content information. Next, we provide an overview of methods that integrate multimodal content information, and lastly we review studies on top-N recommendation (a crucial problem in recommender systems). More extensive overviews of content-based and collaborative filtering methods for recommendation are available in [19, 22, 39].

9.1 Semantic information for user and item representations

Semantic analysis enables learning accurate profiles of users and items thanks to references to external knowledge bases, such as ontologies or semi-structured encyclopedic knowledge. A recommender system can benefit from such analyses, which provide conceptual, linguistic and cultural background knowledge. Several research works build semantic representations based on ontologies. SiteIF [23] is a personal agent for a multilingual news Web site using MultiWordNet[7] as an external knowledge source to model user interests. The ITem Recommender system or ITR [9, 41] can provide recommendations for items in several domains, using Wordnet together with a document representation model called bag-of-synsets, which is an extension of the bag-of-words model [40]. QuickStep [27] is a system for the recommendation of on-line academic research papers using an ontology obtained from DMOZ open directory project and semantically annotates documents using k-nearest neighbor classification.

[7] A multilingual lexical database where English and Italian senses are aligned.

Other semantic analysis approaches make use of semi-structured encyclopedic knowledge sources such as Wikipedia or the Yahoo! Web Directory. Wikipedia was used to estimate similarity between movies [20] in order to provide recommendations for the Netflix Prize competition by using a k-nearest neighbor and a pseudo-SVD algorithm. In [47], an approach for filtering RSS feeds and e-mails is presented, which makes use of Wikipedia to automatically generate the user profiles from the user's document collection. Another approach which uses the WordSpace model and Wikipedia for content analysis was presented in [42]. The dimensions of the WordSpace model represent semantic concepts and the points in the space represent documents [37].

9.2 Multimodal information for recommendation

Several authors have highlighted the need for integrating various modalities in the process of item recommendation. MadFilm [18] is a multimodal movie recommendation system that uses both modalities from natural language and direct manipulation. In [5], a multimodal video recommendation system was proposed, which predicts the topical relevance of a video by analyzing affective aspects of user behavior. In [45], the authors present a digital TV content recommendation system based on descriptive metadata collected from versatile sources. They used a combined multimodal approach which integrates classification-based and keyword-based similarity predictions. In [26], the authors present a contextual video recommendation system which was based on multimodal content relevance and user feedback based on visual, audio and textual information.

In [4], the authors proposed a multimodal recommender system which can predict topical relevance, by exploiting interaction data, contextual information as well as users' affective responses. In [11], authors used multimodal information from radio and television channels, websites, written and spoken content. The personal interests are inferred using natural language processing of the users' blogs. Latent semantic analysis was used to find relationships between user's interests and items to recommend. Authors in [49] presented a video recommendation system based on multimodal fusion and relevance feedback. They defined the multimodal relevance as a textual, visual and aural relevance and calculated the different intra-weights for each modality and inter-weights among them.

9.3 Top-N recommendation

In contrast to recommender systems that operate on numerical ratings, top-N recommender systems focus on a fixed number of items *N* that might be of interest to users [7]. They usually operate on unary feedback, either explicit or implicit (obtained from user behavior data). The methods for top-N recommendation can be broadly divided in two categories: neighborhood-based vs. model-based collaborative filtering [10]. Typically, these methods are derived from traditional recommender systems [19, 22] though some are specific to the top-N task, as follows. Collaborative filtering was achieved with implicit feedback in [16], by treating data as indication of positive and negative preferences with varying confidence levels, which were used to provide explanations to users. Sparse linear methods were proposed in [28] to generate top-N recommendations by solving a regularized optimization problem. Other works have formulated this as a ranking problem. In [35], the authors adopted a Bayesian perspective and proposed an optimization criterion to solve the task (Bayesian Personalized Ranking). Another model was trained by maximizing directly the Mean Reciprocal Rank evaluation metric for top-N recommendations [44].

In [29, 30], the one-class collaborative filtering problem is formulated, i.e. dealing only with positive instances of user feedback. Several schemes were proposed to weigh the negative class in a discriminative fashion, formulated under a matrix factorization framework. These weighting mechanisms performed better in the one-class task than the assumptions that treat all the missing instances as negative or unknown. Authors in [46] suggested to treat zero-valued pairs as optimization variables computed from the training data. Thus, instead of making an assumption about the negative class, the distribution of the negative class is learned. In [21] the authors demonstrate how to incorporate rich user information (history of search, browsing and purchasing) to improve one-class collaborative filtering.

10 Conclusion and future work

In this paper, we introduced a new dataset, the TED lectures, and formulated two benchmark tasks for non-fiction multimedia recommendation utilizing the available ground truth. The feature selection experiments over 80 % of the most active TED users indicated that the most informative data fields for CB methods are the description and the title of each lecture. Using cross-validation, CB using Explicit Semantic Analysis was found to outperform all other CB methods.

We compared in detail content-based, collaborative-filtering, and combined recommendation methods over the test set in two different settings: a cold-start one and a non-cold-start one. The benchmark scores obtained for lecture recommendation are comparable to similar studies on other tasks, e.g. movie recommendation [7]. We showed that the semantic-based methods (ESA, RP and LSI) were able to make more relevant recommendations than keyword-based ones (TFIDF) in a cold-start setting, making them particularly applicable to multimedia datasets into which new items are inserted frequently. Even though we focused on the text modality, the proposed similarities can be potentially used for audio and visual modalities as well. However, the CB methods were outperformed by CF ones in a non-cold-start setting, although a combined method using a neighborhood model, user/item biases and TF-IDF similarity achieved reasonable performance compared to pure CF by utilizing only the popularity bias. The proposed method can be used when newly-added and older items are both present, as it does not rely entirely on collaborative rating similarities.

According to our knowledge, no other dataset with factual audio visual material contains both content metadata and explicit user feedback (favorites)—a fact that points to the potential value of the TED dataset for multimedia recommendation. If other audio visual collections with explicit feedback such as favorites are made available, then the algorithms proposed in this paper are directly applicable to them. If no explicit feedback is available, we have shown elsewhere [33] how to leverage other user-generated information such as comments.

We will further explore algorithms inspired from such tasks, in particular hybrid ones, especially given that the TED dataset has rich content information to be exploited. We will also use semantic spaces with other learning models, such as matrix factorization, and improve the fusion of CB and CF information. Lastly, we will assess recommendation performance when automatically-assigned values are available for metadata fields, for instance through automatic speech recognition (for TRA), speaker detection (for SP), or automatic summarization (for DE).

Acknowledgments We would like to thank the managers of the TED website for their support in accessing and distributing the TED metadata, as well as the anonymous MTAP reviewers for their insightful suggestions. This paper includes material presented at the CBMI 2013 workshop [32], and we are grateful to the organizers for this special issue of the MTAP journal related to CBMI.

References

1. Adomavicius G, Tuzhilin A (2011) Context-aware recommender systems. In: Ricci F, Rokach L, Shapira B, Kantor PB (eds) Recommender systems handbook. Springer, New York, pp 217–253
2. Anderson C (2006) The long tail: why the future of business is selling less of more. Hyperion, New York
3. Antulov-Fantulin N, Bošnjak M, Žnidaršič M, Grčar M, Morzy M, Šmuc T (2011) ECML/PKDD 2011 discovery challenge overview. In: Proceedings of the ECML/PKDD 2011 discovery challenge workshop, Athens
4. Arapakis I, Moshfeghi Y, Joho H, Ren R, Hannah D, Jose JM (2009) Enriching user profiling with affective features for the improvement of a multimodal recommender system. In: Proceedings of the ACM international conference on image and video retrieval, Santorini, CIVR '09, pp 29:1–29:8
5. Arapakis I, Moshfeghi Y, Joho H, Ren R, Hannah D, Jose JM (2009) Integrating facial expressions into user profiling for the improvement of a multimodal recommender system. In: Proceedings of the 2009 IEEE international conference on multimedia and expo, New York, ICME'09, pp 1440–1443
6. Blei DM, Ng AY, Jordan MI (2003) Latent dirichlet allocation. J Mach Learn Res 3(30):993–1022
7. Cremonesi P, Koren Y, Turrin R (2010) Performance of recommender algorithms on top-n recommendation tasks. In: Proceedings of the 4th ACM conference on recommender systems, Barcelona, RecSys '10
8. Dasiopoulou S, Tzouvaras V, Kompatsiaris I, Strintzis M (2010) Enquiring MPEG-7 based multimedia ontologies. Multimed Tools Appl 46(2–3):331–370
9. Degemmis M, Lops P, Semeraro G (2007) A content-collaborative recommender that exploits wordnet-based user profiles for neighborhood formation. User Model User-Adap Inter 17(3):217–255
10. Deshpande M, Karypis G (2004) Item-based top-N recommendation algorithms. ACM Trans Inf Syst 22(1):143–177
11. Di Massa R, Montagnuolo M, Messina A (2010) Implicit news recommendation based on user interest models and multimodal content analysis. In: Proceedings of the 3rd international workshop on automated information extraction in media production, Firenze, AIEMPro '10, pp 33–38
12. Federico M, Cettolo M, Bentivogli L, Paul M, Stüker S (2012) Overview of the IWSLT 2012 evaluation campaign. In: Proceedings of the international workshop on spoken language translation, Hong-Kong, IWSLT '12
13. Gabrilovich E, Markovitch S (2007) Computing semantic relatedness using Wikipedia-based explicit semantic analysis. In: Proceedings of the 20th international joint conference on artificial intelligence, Hyderabad, IJCAI'07, pp 1606–1611
14. Herlocker JL, Konstan JA, Terveen LG, Riedl JT (2004) Evaluating collaborative filtering recommender systems. ACM Trans Inf Syst 22(1):5–53
15. Hofmann T (1999) Probabilistic latent semantic indexing. In: Proceedings of the 22nd annual international ACM SIGIR conference on research and development in information retrieval, Berkeley, SIGIR '99
16. Hu Y, Koren Y, Volinsky C (2008) Collaborative filtering for implicit feedback datasets. In: Proceedings of the 8th IEEE international conference on data mining, ICDM '08, pp 263–272
17. Joachims T (2006) Training linear SVMs in linear time. In: Proceedings of the ACM conference on knowledge discovery and data mining, Philadelphia, KDD '06, pp 217–226
18. Johansson P (2003) Madfilm—a multimodal approach to handle search and organization in a movie recommendation system. In: Proceedings of the 1st nordic symposium on multimodal communication, Copenhagen, pp 53–65
19. Koren Y, Bell R (2011) Advances in collaborative filtering. In: Ricci F, Rokach L, Shapira B, Kantor PB (eds) Recommender systems handbook. Springer, New York, pp 145–186
20. Lees-Miller J, Anderson F, Hoehn B, Greiner R (2008) Does Wikipedia information help Netflix predictions? In: Proceedings of the 7th international conference on machine learning and applications, San Diego, ICMLA '08, pp 337–343
21. Li Y, Hu J, Zhai C, Chen Y (2010) Improving one-class collaborative filtering by incorporating rich user information. In: Proceedings of the 19th ACM international conference on information and knowledge management, Toronto, CIKM '10, pp 959–968
22. Lops P, Gemmis M, Semeraro G (2011) Content-based recommender systems: state of the art and trends. In: Ricci F, Rokach L, Shapira B, Kantor PB (eds) Recommender systems handbook. Springer, New York, pp 73–105

23. Magnini B, Strapparava C (2001) Improving user modelling with content-based techniques. In: Bauer M, Gmytrasiewicz P, Vassileva J (eds) User modeling 2001. Springer, New York, pp 74–83
24. Mahmood T, Ricci F (2009) Improving recommender systems with adaptive conversational strategies. In: Proceedings of the 20th ACM conference on hypertext and hypermedia, Torino, HT '09, pp 73–82
25. Martinez J (2002) Standards—MPEG-7 overview of MPEG-7 description tools, part 2. IEEE Multimed 9(3):83–93
26. Mei T, Yang B, Hua XS, Li S (2011) Contextual video recommendation by multimodal relevance and user feedback. ACM Trans Inf Syst 29(2):10:1–10:24
27. Middleton SE, Shadbolt NR, De Roure DC (2004) Ontological user profiling in recommender systems. ACM Trans Inf Syst 22(1):54–88
28. Ning X, Karypis G (2011) SLIM: sparse linear methods for top-N recommender systems. In: Proceedings of the 11th IEEE international conference on data mining, Vancouver, ICDM '11, pp 497–506
29. Pan R, Scholz M (2009) Mind the gaps: weighting the unknown in large-scale one-class collaborative filtering. In: Proceedings of the 15th ACM SIGKDD international conference on knowledge discovery and data mining, Paris, KDD '09, pp 667–676
30. Pan R, Zhou Y, Cao B, Liu N, Lukose R, Scholz M, Yang Q (2008) One-class collaborative filtering. In: Proceedings of the 8th IEEE international conference on data mining, Pisa, ICDM '08, pp 502–511
31. Papagelis M, Plexousakis D (2005) Qualitative analysis of user-based and item-based prediction algorithms for recommendation agents. In: Engineering applications of artificial intelligence, Pergamon, pp 152–166
32. Pappas N, Popescu-Belis A (2013) Combining content with user preferences for TED lecture recommendation. In: Proceedings of the 11th international workshop on content based multimedia indexing, Veszprém, Hungary, CBMI '13, pp 47–52
33. Pappas N, Popescu-Belis A (2013) Sentiment analysis of user comments for one-class collaborative filtering over TED talks. In: Proceedings of the 36th ACM SIGIR conference on research and development in information retrieval, Short papers, Dublin, SIGIR '13, pp 773–776
34. Řehůřek R, Sojka P (2010) Software framework for topic modeling with large corpora. In: Proceedings of the LREC 2010 workshop on new challenges for NLP frameworks, Valletta, pp 45–50
35. Rendle S, Freudenthaler C, Gantner Z, Schmidt-Thieme L (2009) BPR: Bayesian personalized ranking from implicit feedback. In: Proceedings of the 25th conference on uncertainty in artificial intelligence, Montreal, UAI '09, pp 452–461
36. Sahlgren M (2005) An introduction to random indexing. In: Proceedings of the 7th international conference on terminology and knowledge engineering, methods and applications of semantic indexing workshop, vol 5. Copenhagen
37. Sahlgren M (2006) The word-space model: using distributional analysis to represent syntagmatic and paradigmatic relations between words in high-dimensional vector spaces. PhD thesis, Stockholm University, Stockholm
38. Salton G, Buckley C (1988) Term-weighting approaches in automatic text retrieval. Inf Process Manag 24(5):513–523
39. Sarwar B, Karypis G, Konstan J, Riedl J (2001) Item-based collaborative filtering recommendation algorithms. In: Proceedings of the 10th international conference on World Wide Web, Hong Kong, WWW '01, pp 285–295
40. Semeraro G, Degemmis M, Lops P, Basile P (2007) Combining learning and word sense disambiguation for intelligent user profiling. In: Proceedings of the 20th international joint conference on artificial intelligence, Hyderabad, IJCAI '07, pp 2856–2861
41. Semeraro G, Basile P, De Gemmis M, Lops P (2009) User profiles for personalizing digital libraries. In: Theng Y, D G, Foo S, Na J (eds) Handbook of research on digital libraries design development and impact. Information Science Reference, pp 149–158
42. Semeraro G, Lops P, Basile P, de Gemmis M (2009) Knowledge infusion into content-based recommender systems. In: Proceedings of the 3rd ACM conference on recommender systems, New York, RecSys '09, pp 301–304
43. Shani G, Gunawardana A (2011) Evaluating recommendation systems. In: Ricci F, Rokach L, Shapira B, Kantor PB (eds) Recommender systems handbook. Springer, New York, pp 257–297
44. Shi Y, Karatzoglou A, Baltrunas L, Larson M, Oliver N, Hanjalic A (2012) CLiMF: learning to maximize reciprocal rank with collaborative less-is-more filtering. In: Proceedings of the 6th ACM conference on recommender systems, Dublin, RecSys '12, pp 139–146
45. Shin H, Lee M, Kim E (2009) Personalized digital TV content recommendation with integration of user behavior profiling and multimodal content rating. IEEE Trans Consum Electron 55(3):1417–1423
46. Sindhwani V, Bucak SS, Hu J, Mojsilovic A (2009) A family of non-negative matrix factorizations for one-class collaborative filtering problems. In: Proceedings of the 3rd ACM conference on recommender systems, recommender based industrial applications workshop, New York, RecSys '09

47. Smirnov AV, Krizhanovsky A (2008) Information filtering based on Wiki index database. CoRR arXiv:08042354
48. Tsinaraki C, Christodoulakis S (2006) A multimedia user preference model that supports semantics and its application to MPEG 7/21. In: Proceedings of the 12th international conference on multi-media modelling, Beijing, p 8
49. Yang B, Mei T, Hua XS, Yang L, Yang SQ, Li M (2007) Online video recommendation based on multimodal fusion and relevance feedback. In: Proceedings of the 6th ACM international conference on image and video retrieval, Amsterdam, CIVR '07, pp 73–80

Nikolaos Pappas is a PhD student at Ecole Polytechnique Fédérale de Lausanne (EPFL) and a research assistant at Idiap Research Institute. He received his Dipl.Eng. in Information and Communication Systems Engineering in 2009 and his M.Sc. in Information Management in 2011 at University of the Aegean, Greece. His research interests are in information retrieval and natural language processing based on machine learning and focused on recommender systems. His PhD thesis is about extracting preference information from user-generated text and modeling user preferences from multiple sources, supervised by Dr. Andrei Popescu-Belis and Prof. Hervé Bourlard.

Andrei Popescu-Belis is a senior researcher at the Idiap Research Institute in Martigny, Switzerland, and the head of Idiap's Natural Language Processing group. A graduate of the Ecole Polytechnique, Paris, he has a PhD from LIMSI-CNRS and the University of Paris XI. His research interests are in natural language processing, information retrieval, multimedia systems, and multimodal resources. He has been involved in several large Swiss and European projects, and has over 100 reviewed publications. As a lecturer at Ecole Polytechnique Fédérale de Lausanne (EPFL), he teaches the doctoral course on Human Language Technology.