Toward Building a Content-Based Video Recommendation System Based on Low-Level Features

Yashar Deldjoo(✉), Mehdi Elahi, Massimo Quadrana, and Paolo Cremonesi

Politecnico di Milano, Milan, Italy
{yashar.deldjoo,mehdi.elahi,massimo.quadrana,paolo.cremonesi}@polimi.it
http://www.polimi.it

Abstract. One of the challenges in video recommendation systems is the *New Item* problem, which happens when the system is unable to recommend video items, that no information is available about them. For example, in the popular movie-sharing websites, such as *Youtube*, everyday, hundred millions of hours of videos are uploaded and big portion of these videos may not contain any meta-data, to be used by the system to generate recommendations.

In this paper, we address this problem by proposing a method, that is based on automatic analysis of the video content in order to extract a number representative low-level visual features. Such features are then used to generate personalized content-based recommendations. Our evaluation shows that our proposed method can outperform the baselines, by producing more relevant recommendations. Hence, a set low-level features extracted automatically can be more descriptive and informative of the video content than a set of high-level expert annotated features.

Keywords: Recommender systems · Content based · Low level · Video

1 Introduction

Recommender Systems (RSs) are tools and techniques that suggest to users, a set of items that may be of their interest [25]. Several approaches have been already proposed and used for recommendation generation [3,11,26,28]. *Content-based* recommendation [5,22] is the classical approach that suggests items based on their associated features. For instance, news recommender systems consider the terms in the news articles as features and recommend to user the news articles that have features similar to the ones the user preferred before.

In order to generate this type of recommendations, the system must have some information about the items, beforehand. Accordingly, the system may not be able to recommend items that are new and no information is available about them. For example, in video recommendation, it may not be feasible for the system to recommend videos that no meta-data is given. Such meta-data can be of different forms, such as, the movie genre, the cast, date of production, reviews, etc.

H. Stuckenschmidt and D. Jannach (Eds.): EC Web 2015, LNBIP 239, pp. 1–12, 2015.
DOI: 10.1007/978-3-319-27729-5_4

In this paper, we propose exploitation of low-level visual features, extracted from videos, in order to generate relevant recommendations. This can be used in two scenarios: (i) *New Item* scenario, i.e., there are videos, such as user generated videos, that the system has no content rather than the video file itself, and (ii) *Existing Item* scenario, i.e., some information is available for videos such as description, genre, or cast and the low-level features are used in order to improve the quality of the recommendation system. Indeed, to the best of our knowledge, all the very few related works [30,31], focused only on the second scenario. This is while, the new item scenario, is even more important to address, since in such scenario, the typical recommender systems may completely fail to generate personalized recommendations for users.

In this paper, we mainly focus on the first scenario, i.e., new item scenario, and propose a method, that automatically extracts the low-level visual features from the video content and use it for recommendation propose. We form and test the following hypothesis: a content-based recommender system, which uses a set of representative visual features of video contents, may have led to a higher accuracy in comparison to the genre based recommender system. Our offline evaluation, described later, has shown promising results, and verified our hypothesis.

The main contributions of the paper are the followings:

- we propose a method to remedy the (extreme) *New Item* problem [14] in video recommendation domain, i.e., when a new video item is added to the database, with absolutely no meta-data provided
- we assume a more realistic scenario, i.e., an up-and-running video recommender with thousands of users rather than only tens of users, that has been typically considered in the related work
- we propose a novel application of the video classification in the recommendation systems, that has been explored marginally
- we test our proposed method with the state-of-the-art evaluation methodology and measure its performance with respect to a well known *Recall* metric

The rest of the paper is organized as follows: The next section reviews the research works that are related to content-based recommender systems and existing video recommender systems. Afterwards, in Sect. 3 we describe our novel method for representing the videos based on low-level visual features, as well as our recommendation algorithm in detail. In Sect. 4 we describe the offline evaluation strategy, we conducted, to compare our proposed method with other competing methods, and in Sect. 5 we discuss the obtained results. Finally in Sect. 6, we conclude the paper and outline the future work.

2 Related Work

2.1 Content-Based Recommender Systems

Content-based recommender systems analyze a set of descriptions of the items, previously rated by a user, to build a profile of her preferences and interests

according to the attributes of the objects rated by her. Indeed, recommendations are generated by matching up the attributes of the user profile (i.e., a structured representation of her interests) against the attributes of a item. In order to do so, most of the content-based recommender systems build a Vector Space Model (VSM) representation of item features. Each item is represented by a vector in a n-dimensional space, where each dimension represents an attribute from the overall set of attributes used to describe the items. Using this model, the system computes a relevance score that represents the user's degree of interest toward that item [19]. For example, in a movie recommender system, the features that represents an items can be actors, director, or genre. This strict connection with the description of items in the catalogue, also allows content-based recommender systems to produce explanations to recommendations and to naturally handle the new item problem [14].

There are various content-based recommendation algorithms. For example, classical "k-nearest neighbor"approach (KNN) computes the interest of a user for an unseen item by comparing it against all the items seen by the user in the catalogue. Each seen item contributes to predict the interest score in a way proportional to its similarity with the unseen item; this similarity is computed by means of a similarity function like *cosine similarity* or *Pearson correlation* over items' VSM representation [7,20]. Other approaches try to model the probability for the user to be interested to a target item using a Bayesian approach [21], or exploits other techniques adapted from Information Retrieval like the Relevance Feedback method [4].

Regardless of which recommendation algorithm is used, in media recommendation, the recommender system can generate recommendation based on two different types of item attributes (or features): i.e., *High-Level* (or semantic) features (HL) or *Low-Level* features (LL). The high-level features can be collected both from structured sources, such as databases, lexicons and ontologies, and from unstructured sources, such as reviews, news articles, item descriptions and social tags [4,7,12,20,21]. The low-level features, on the other hand, can be extracted directly from the media itself. For example, in music recommendation many acoustic features, e.g. rhythm and timbre, can be extracted and used to find perceptual similar tracks [8,9,17,27].

2.2 Video Recommendation and Retrieval

In video recommendation, a few works in the past have leveraged the low-level features directly extracted from the visual content itself within the recommendation process [18,30,31]. Yang et al. [30] presented a video recommender system, VideoReach, which combines textual, visual and aural video features to increase click-through-rate. Zhao et al. [31] propose a multi-task learning algorithm to integrate multiple ranking lists generated by exploring different information sources, visual content included. However, none of these previous works has considered how visual features can effectively replace the other typical content information when they are not available. Indeed, they did not address the

new item scenario, where no or very little information about a video is provided to a recommender system. Instead, they considered the scenario where the low-level content is given in addition to other information and it is used to improve the quality of the recommendation. However, in this paper we address the extreme new item problem where absolutely no information is available for a video, and the system may fail to recommend this video to the users.

It worth noting that, while usage of low-level feature based video representation has been studied marginally in recommender systems community, it has been extensively researched in the other communities such as Computer Vision [24], and it has been used in a number of similar applications such as Content-Based Video Retrieval systems (CBVR). In this case, although the objectives of content-based video retrieval and video recommendation system might be different [30], they share a main approach for dealing with their specific problems which is searching for the best informative features that can represent a video. Hence, we also review briefly the literature in related research areas.

A few comprehensive surveys can be found in [10,16]. These surveys provide a good frame of reference for reviewing the literature related to video content analysis providing a large body of low-level features that can be used for video content analysis. These features are derived from either visual, auditory or textual modalities or combination of them. For example, in [24], Rasheed et al. proposed a practical movie genre classification scheme based on solely computable visual cues. In [23], the authors proposed a similar approach by considering also the audio features. Finally, in [32] Zhou et al. propose a framework for automatic classification using a temporally-structured feature based on intermediate level representation of scenes.

3 Method Description

The first step in order to build a content-based video recommendation system is search for the features that can bridge the gap between high-level concepts and low-level contents in videos. These features must comply with human norms of perception and abide by the grammar of the film - the rules creators of movies use to make a movie. In general, a movie M can be represented by three main modalities, visual, audio and text $M = M(M_V, M_A, M_T)$. In this work, we only focus on visual features, therefore

$$M = M(M_V) \tag{1}$$

where the visual modality M_V can be represented by a set of features

$$M_V = M_V(f_v) \tag{2}$$

where $f_v = (f_{v1}, f_{v2}, ..., f_{vn})$ is a set of n features obtained from the visual content. By carefully studying the features commonly used in the literature, we selected the features studied by the authors in the vision community [24] under mild modifications. We later analyzed the accuracy of features selected by performing a classification analysis and features selection based on exhaustive search.

3.1 Visual Features

A total of four main low-level visual features were used in our experiment from which a feature vector of length six ($n = 6$) was extracted to represent each video. They include

- *Average shot length*: A shot is a single camera action and the number of shots in a video can provide useful information about the pace at which a movie is being created. Average shot length is defined by

$$\overline{L}_{sh} = \frac{n_f}{n_{sh}} \tag{3}$$

  where n_f is the number of frames and n_{sh} the number of shots in a movie. For example, action movies usually contain rapid movements of the camera (therefore they contain higher number of shots or shorter shot lengths) compared to dramas which often contain conversations between people (thus longer average shot length). Because movies can be made a different frame rates, $\overline{L}_{sh}$ is further normalized by the frame rate of the movie.
- *Color variance*: The variance of color has a strong correlation with the genre. For instance, directors tend to use a large variety of bright colors for comedies and darker hues for horror films. For each key frame represented in Luv color space we compute the covariance matrix:

$$\rho = \begin{pmatrix} \sigma_L^2 & \sigma_{Lu}^2 & \sigma_{Lv}^2 \\ \sigma_{Lu}^2 & \sigma_u^2 & \sigma_{uv}^2 \\ \sigma_{Lv}^2 & \sigma_{uv}^2 & \sigma_v^2 \end{pmatrix} \tag{4}$$

  The generalized variance can be used as the representative of the color variance in each key frame given by

$$\Sigma = det(\rho) \tag{5}$$

  in which a key frame is a representative frame within a shot (e.g. the middle shot).
- *Motion*: Motion within a video can be caused mainly by the camera movement (*i.e.* camera motion) or movements on part of the object being filmed (*i.e.* object motion). While the average shot length captures the former characteristic of a movie, it is desired for the motion feature to also capture the latter characteristic. A motion feature descriptor based on optical flow [6,15] was used which provides a robust estimate of the motion in sequence of images based on velocities of images being filmed. Because motion features are based upon sequence of images, they are calculated over the entire frames rather on solely key frames.
- *Lightening*: Lightening is another distinguishing factor between movie genres in such a way that the director use it as a factor to control the type of emotion they want to be induced to a user. For example, comedy movies often adopt lightening which has abundance of light (*i.e.* high gray-scale mean) with less contrast between the brightest and dimmest light (*i.e.* high gray-scale standard deviation). This trend is often known as *high-key* lightening.

On the other hand, horror movies or noir films often pick gray-scale distributions which is low in both gray-scale mean and gray-scale standard deviation, known by *low-key* lightening. In order to capture both of these parameters, after transforming all key-frames to HSV color-space [29], we compute the mean μ and standard deviation σ of the value component which corresponds to the brightness. The scene lightening key ξ defined by Eq. 6 is used to measure the lightening of key frames

$$\xi = \mu.\sigma \tag{6}$$

For instance, comedies often contain key-frames which have a well distributed gray-scale distribution which results in both the mean and standard deviation of gray-scale values to be high therefore for comedy genres one can state $\xi > \tau_c$, whereas for horror movies the lightening with poorly distributed lighting the situation is reverse and we will have $\xi < \tau_h$. In the situation where $\tau_h < \xi < \tau_c$ other movie genres (*e.g.* Drama) exists where it is hard to use the above distinguish factor for them.

3.2 Recommendation Algorithm

To generate recommendations using our Low-Level descriptors we adopted a classical "k-nearest neighbor" content-based algorithm. Given a set of users U and a catalogue of items I, a set of preference scores r_{ui} has been collected. Moreover, each item $i \in I$ is associated to its feature vector $\boldsymbol{f_i}$. For each couple of items i and j, a similarity score s_{ij} is computed using *shrunk cosine similarity* as follows

$$s_{ij} = \frac{\boldsymbol{f_i}^T \boldsymbol{f_j}}{\|\boldsymbol{f_i}\| \, \|\boldsymbol{f_j}\| + \lambda} \tag{7}$$

where $\lambda > 0$ is the shrinkage factor. For each item i the set of its nearest neighbors NN_i is built, $|NN_i| < K$. Then, for each user $u \in U$, the predicted preference score $\hat{r_{ui}}$ for an unseen item i is computed as follows

$$\hat{r_{ui}} = \frac{\sum_{j \in NN_i, r_{uj} > 0} r_{uj} s_{ij}}{\sum_{j \in NN_i, r_{uj} > 0} s_{ij}} \tag{8}$$

4 Evaluation Methodology

We have formulated the following hypothesis: the content-based recommender system, that exploits a set of representative visual features of video contents, may have led to a higher accuracy in comparison to the genre based recommender system. Hence, we speculate that a set low-level features extracted automatically may be more informative of the video content than a set of high-level expert annotated features.

In order to test our hypothesis, we evaluate the recommendation quality of each of the considered content-based recommender system it terms of *Recall(K)*,

where K is the size of the recommendation list. If a user u has N_u relevant items, the recall in its recommendation list of size K is computed as

$$Recall(K) = \sum_{u \in U} \sum_{i=1}^{k} \frac{rel(i)}{N_u} \tag{9}$$

where $rel(i) = I[r_{ui} >= 4]$ is the item relevance function and I is the indicator function. We evaluated the *Recall(K)* Leave-One-Out Cross-Validation (LOOCV) [13]. At each step, one relevant sample (i.e., an item having rating greater than 4) is removed from each user profile. We also removed the ratings for top-10 most popular items to discount the effect of the well known popularity effect. Then, the recommendation model is built on the remaining samples and the quality of the recommendation lists is evaluated. Results are finally averaged over all the splits that have been generated.

We have used a set of movie trailers, that the were sampled randomly from all the genres, i.e., Action, Comedy, Drama and Horror. The movie titles were selected from Movielens dataset [1], and the files were obtained from *YouTube* [2]. The dataset contained over all 210 movies, 120 of which belonging to a single genre and 90 movies belonging to multiple genres.

5 Results

Figure 1 illustrates the system's recall for different content-based recommendation methods, i.e., visual feature based (Low Level-LL), genre-based (High Level-HL Genre), and Hybrid (LL-HL Genre). We performed a feature selection based on exhaustive search and chose the best configuration as shown in the figure. Comparing the results, it is clear that our proposed method, i.e., visual feature based outperforms the other methods in terms of recall. As it can be seen in Fig. 1, the recall values for all the methods, initially begins with 0 for $N = 1$, and, as N is incremented, it increases monotonically and reaches 0.36 for visual feature based (LL), 0.14 for genre based (HL genre), and 0.11 for hybrid method (LL + HL Genre), respectively. We have also performed t-test and realized that the recall values of our method (LL) is significantly higher than (HL - Genre) with p-value = 0.01912, and hybrid (LL + HL Genre) with p-value = 0.00736. However, no significant difference has been observed between the visual feature based (HL - Genre) and hybrid (LL + HL Genre) methods (p-value = 0.30581). Hence, it is clear that our proposed method can perform the best among all the other methods. In fact, it shows that, our extracted features can represent very well the videos, allowing the system to make personalized recommendations, that better match the users tastes, specially in the extreme new item cold start situation.

In addition to the figure, in Table 1, we report more detailed results, for the different system parameters, i.e., the size of the neighborhood set (k) and the size of the recommendation list (N). In this table, we report the performance when all the proposed LL features are used (not only the best feature combination).

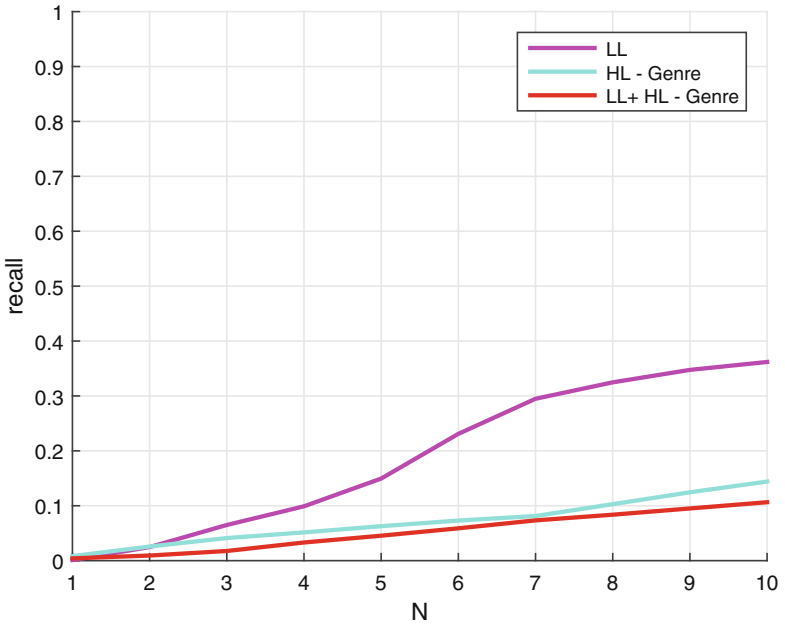

Fig. 1. Performance comparison of different CB methods under best feature combination

We have compared the results of the methods for all possible neighborhood size (k), and realized that for instance, when $N = 10$, there is an inverse proportional relation between k and recall values of LL method whereas this relation is directly proportional for hybrid method. The same condition almost exists when $N = 5$.

Moreover, it could observed that with small k, the performance of the LL method is much better than HL genre and the hybrid method for both $N = 5$ and $N = 10$. When k is high, the recall values of the hybrid method are greater than the recall values of LL and HL genre methods. However, the obtained recall values of hybrid method are still lower than the recall values of the LL method when k is small. Also, since the number of items (videos) used in our catalog is limited to only 210 items, high values of k are not justifiable for use in the knn algorithm. For these reasons, we can conclude the best performance is obtained for LL method. Indeed addition of the visual features to the genre information, do not improve the quality of the genre based method. This can be due to the fact that there is a strong correlation between the genre of a movie and the visual features that represent that movie. In fact, as noted before, it has been shown that different genres of movies, differ significantly in terms of visual characteristics.

In order to better understand this, we have also analyzed the observations and tried to classify the videos into different genres, exploiting the extracted visual features. We note that here we assumed every video to belong to only a single

genre and hence considered only a subset of 120 single genre videos. We have tried different classifiers and realized that the best classification accuracy has been achieved by *Decision Tables*. We conducted 10 fold cross-validation and obtained accuracy of 73.33 %. Indeed, using the visual features, the classifier managed to successfully classify most of the videos in their correct genre. We have observed the best classification was done for comedy movies. Indeed, 27 out of 30 movies were successfully classified in their corresponding comedy genre. On the other hand, the most erroneous classification happened for the horror genre. Indeed, 8 out of 30 horror movies have been mistakenly classified as action genre. This is a phenomenon, that was expected, since typically there are many action scenes occurred in horror movies, and this may make the classification very hard.

Table 1. Performance comparison of different CB methods, in terms of Recall metric, for different neighborhood size (k) and recommendation list size (N) when all LL features are used.

	k = 4			k = 5			k = 10			k = 15		
	LL	HL	hybrid	LL	HL	hybrid	LL	HL	hybrid	LL	HL	hybrid
N = 5	**0.1476**	0.0669	0.0400	**0.1385**	0.0552	0.0393	0.0806	0.0575	**0.0835**	0.0731	0.0581	**0.1239**
N = 10	**0.2286**	0.1113	0.0628	**0.2044**	0.1025	0.0831	**0.1448**	0.1339	0.1224	0.1198	0.1430	**0.1891**

Having considered all the results, we remark that our considered hypothesis has been successfully validated, i.e., a proper extraction of the visual features of videos may have led to higher accuracy of video recommendation, than the typical expert annotation method. Indeed, it is very promising to achieve higher accuracy with automatic method than a manual method (i.e., experts analyses and annotation of videos) since the later method is very costly and in some cases even impossible (e.g., in huge datasets).

Finally, it is worth noting that, our results has been obtained by using the trailer of the movies that are only a small sample of videos themselves. Watching the trailers of even few movies, one can simply notice that the structure of them are not far different and hence the trailers of the movies do actually share many similarities which makes it much more difficult for our method to work properly. Indeed, it is much more difficult to use low-level visual features to classify movies or generate relevant movie recommendations, with their trailers, than the movies themselves. Hence, achieving high accuracy in either of the tasks, indicates the great effectiveness of our proposed method.

6 Conclusion and Future Work

In this paper, we have address the *New Item* problem by presenting a novel content-based method for video recommendation task. The proposed method extracts and uses the low-level visual features from video content in order to

provide a user with personalized recommendations, without relying on any high-level features, such as, meta-data, genre, cast, or reviews, that are more costly to collect and are not available in new item cold-start situation.

We have developed the a research hypotheses, i.e., a proper extraction of the visual features of videos may have led to higher accuracy of video recommendation, than the typical expert annotation method. Based on the experiments, we conducted, we successfully verified the hypothesis and shown that the recommendation accuracy is significantly higher when using the low-level visual features than high-level genre data.

Our future work comprises the further analysis with bigger and different datasets, that we will prepare, in order to better understand the performance differences among the compared methods. We would like to also investigate the impact of using different content-based recommendation algorithms, such as those based on Bayesian, or SVD, on the performance of our method. We would like to also include additional sources of information, such as, audio features, in order to farther improve the quality of our content based recommendation method. Last but not least, we plan to perform a feature selection study in order to better understand the role and importance of the features in the performance of the CB video recommendation algorithm(s).

Acknowledgments. This work is supported by Telecom Italia S.p.A., Open Innovation Department, Joint Open Lab S-Cube, Milan.

References

1. Datasets – grouplens. http://grouplens.org/datasets/, Accessed: 01 May, 2015
2. Youtube. http://www.youtube.com. Accessed: 01 April, 2015
3. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. IEEE Trans. Knowl. Data Eng. **17**(6), 734–749 (2005)
4. Ahn, J.-W., Brusilovsky, P., Grady, J., He, D., Syn, S.Y.: Open user profiles for adaptive news systems: help or harm? In: Proceedings of the 16th international conference on World Wide Web, pp. 11–20. ACM (2007)
5. Balabanović, M., Shoham, Y.: Fab: content-based, collaborative recommendation. Commun. ACM **40**(3), 66–72 (1997)
6. Barron, J.L., Fleet, D.J., Beauchemin, S.S.: Performance of optical flow techniques. Int. J. Comput. Vis. **12**(1), 43–77 (1994)
7. Billsus, D., Pazzani, M.J.: User modeling for adaptive news access. User Model. User-Adap. Inter. **10**(2–3), 147–180 (2000)
8. Bogdanov, D., Herrera, P.: How much metadata do we need in music recommendation? a subjective evaluation using preference sets. In: ISMIR, pp. 97–102 (2011)
9. Bogdanov, D., Serrà, J., Wack, N., Herrera, P., Serra, X.: Unifying low-level and high-level music similarity measures. IEEE Trans. Multimedia **13**(4), 687–701 (2011)
10. Brezeale, D., Cook, D.J.: Automatic video classification: a survey of the literature. IEEE Trans. Syst. Man Cybern. Part C Appl. Rev. **38**(3), 416–430 (2008)

11. Burke, R.: Hybrid recommender systems: Survey and experiments. User Model. User-Adap. Inter. **12**(4), 331–370 (2002)
12. Cantador, I., Szomszor, M., Alani, H., Fernández, M., Castells, P.: Enriching ontological user profiles with tagging history for multi-domain recommendations (2008)
13. Deshpande, M., Karypis, G.: Item-based top-n recommendation algorithms. ACM Trans. Inf. Syst. (TOIS) **22**(1), 143–177 (2004)
14. Elahi, M., Ricci, F., Rubens, N.: Active learning strategies for rating elicitation in collaborative filtering: a system-wide perspective. ACM Trans. Intell. Syst. Technol. (TIST) **5**(1), 13 (2013)
15. Horn, B.K., Schunck, B.G.: Determining optical flow. In: 1981 Technical Symposium East, pp. 319–331. International Society for Optics and Photonics (1981)
16. Hu, W., Xie, N., Li, L., Zeng, X., Maybank, S.: A survey on visual content-based video indexing and retrieval. IEEE Trans. Syst. Man Cybern. Part C Appl. Rev. **41**(6), 797–819 (2011)
17. Knees, P., Pohle, T., Schedl, M., Widmer, G.: A music search engine built upon audio-based and web-based similarity measures. In: Proceedings of the 30th annual international ACM SIGIR conference on Research and Development in Information Retrieval, pp. 447–454. ACM (2007)
18. Lehinevych, T., Kokkinis-Ntrenis, N., Siantikos, G., Dogruöz, A.S., Giannakopoulos, T., Konstantopoulos, S.: Discovering similarities for content-based recommendation and browsing in multimedia collections
19. Lops, P., De Gemmis, M., Semeraro, G.: Content-based recommender systems: state of the art and trends. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) Recommender Systems Handbook, pp. 73–105. Springer, Heidelberg (2011)
20. Middleton, S.E., Shadbolt, N.R., De Roure, D.C.: Ontological user profiling in recommender systems. ACM Trans. Inf. Syst. (TOIS) **22**(1), 54–88 (2004)
21. Mooney, R.J., Roy, L.: Content-based book recommending using learning for text categorization. In: Proceedings of the Fifth ACM Conference on Digital libraries, pp. 195–204. ACM (2000)
22. Pazzani, M.J., Billsus, D.: Content-based Recommendation Systems. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) Adaptive Web 2007. LNCS, vol. 4321, pp. 325–341. Springer, Heidelberg (2007)
23. Rasheed, Z., Shah, M.: Video categorization using semantics and semiotics. In: Rosenfeld, A., Doermann, D., DeMenthon, D. (eds.) Video Mining, pp. 185–217. Springer, Heidelberg (2003)
24. Rasheed, Z., Sheikh, Y., Shah, M.: On the use of computable features for film classification. IEEE Trans. Circ. Syst. Video Technol. **15**(1), 52–64 (2005)
25. Ricci, F., Rokach, L., Shapira, B.: Introduction to recommender systems handbook. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) Recommender Systems Handbook, pp. 1–35. Springer Verlag, Heidelberg (2011)
26. Ricci, F., Rokach, L., Shapira, B.: Introduction to recommender systems handbook. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P. (eds.) Recommender Systems Handbook, pp. 1–35. Springer Verlag, Heidelberg (2011)
27. Seyerlehner, K., Schedl, M., Pohle, T., Knees, P.: Using block-level features for genre classification, tag classification and music similarity estimation. Submission to Audio Music Similarity and Retrieval Task of MIREX 2010 (2010)
28. Su, X., Khoshgoftaar, T.M.: A survey of collaborative filtering techniques. Adv. Artif. Intell. **2009**, 4:2 (2009)
29. Radu, V.: Application. In: Radu, V. (ed.) Stochastic Modeling of Thermal Fatigue Crack Growth. ACM, vol. 1, pp. 63–70. Springer, Heidelberg (2015)

30. Yang, B., Mei, T., Hua, X.-S., Yang, L., Yang, S.-Q., Li, M.: Online video recommendation based on multimodal fusion and relevance feedback. In: Proceedings of the 6th ACM International Conference on Image and Video Retrieval, pp. 73–80. ACM (2007)
31. Zhao, X., Li, G., Wang, M., Yuan, J., Zha, Z.-J., Li, Z., Chua, T.-S.: Integrating rich information for video recommendation with multi-task rank aggregation. In: Proceedings of the 19th ACM International Conference on Multimedia, pp. 1521–1524. ACM (2011)
32. Zhou, H., Hermans, T., Karandikar, A.V., Rehg, J.M.: Movie genre classification via scene categorization. In: Proceedings of the International Conference on Multimedia, pp. 747–750. ACM (2010)