# Movie Genome Recommender: A Novel Recommender System Based on Multimedia Content


Yashar Deldjoo
*Polytechnic University of Bari, Italy*
yashar.deldjoo@poliba.it

Markus Schedl
*Johannes Kepler University Linz, Austria*
markus.schedl@jku.at

Mehdi Elahi
*Free University of Bozen-Bolzano, Italy*
meelahi@unibz.it



***Abstract*—We present a demo application of a web-based recommender systems that is powered by the "Movie Genome", i.e., a rich semantic description of a movie's content, including state-of-the-art audio and visual descriptors and metadata (genre and tags). The current version of the Movie Genome web application implements content-based filtering approaches. Due to its modular implementation and the free availability of its source code, it can be easily extended to various context-aware and user-aware scenarios. A personality questionnaire is already integrated into the web application, which allows it to also serve as testbed for personality-aware recommendation algorithms.**



***Index Terms*—multimedia, movie recommender system, content-based filtering, web application, Movie Genome**


## I. Introduction

Users base their decision making about which movie to watch typically on the movie's content, whether expressed in terms of metadata (e.g., genre, cast, plot, or reviews) or the feeling experienced after watching the corresponding movie trailer. In the latter case, the visual content (*e.g.,* color, lighting, motion) and the audio content (*e.g.,* music or spoken dialogues) play a crucial role in driving users' perceived affinity to the movie. The above examples underline that human interpretation of multimedia items is by nature *content-oriented*. In addition, from a business perspective, a movie's content (both semantic and style) and the casting crew play a key role in determining if the producing company can satisfy the consumers' expectations [1]. Consumers' interpretation of movies is directly linked to their expectations of the multimedia content (i.e., audio, visual, and metadata) and the latter can determine the final increase or decrease in the purchase volume of the movies.

Recommender systems (RS) are nowadays widely applied, in e-commerce websites, movies and music streaming platforms, or on social media to point users to items (products or services). Most real-word movie RS deployed to date rely on collaborative filtering (CF) models due to their state-of-the-art accuracy. CF models exploit the collaborative power of user-item interactions, either implicit (e.g., clicking or purchasing actions) or explicit (e.g., ratings), to compute recommendations and ignoring the role of item content in the recommendation process [2], [3].

As an alternative to CF, content-based filtering (CBF) methods recommend items that are similar in content to the items liked by the user [4], [5]. Since this approach requires almost no preference information of the users, CBF can be effectively used to alleviate cold-start issues and is more privacy-preserving. However, CBF models traditionally use metadata to describe movie content [5]. We argue that the perception of a movie in the eyes of spectators is influenced by many factors, not only related to its genre, cast, and plot, but also to the overall film style [6]. For example, the movies *Schindler's List* and *Empire of the Sun* are both dramatic movies directed by Steven Spielberg, both describing historical events. However, they are completely different in style, with the former shot documentary-like in black and white and the latter shot using bright colors and making heavy use of special effects. Although these two movies share high similarity with respect to metadata, their different styles are likely to affect the viewers' feelings and opinions differently [7].

In this demo paper, we therefore present a novel content-centric web-based framework for movie search and recommendation algorithms that:

1) is powered by a pure CBF model exploiting state-of-the-art audio and visual descriptors as well as editorial metadata (genre and tags), which we call "Movie Genome" [7],
2) supports users with a wide range of functionalities common in online video streaming services such as Netflix,[1]
3) can be easily configured to facilitate the execution of controlled empirical studies such as [7]–[9],
4) can serve as versatile testbed for implementations of personalized recommendation algorithms by embedding questionnaires for demographics, personality, and other user characteristics.

As another contribution, we publicly release the source code of the web application and the Movie Genome features [10].[2]

[1] https://www.netflix.com
[2] https://github.com/yasdel/RecMusicApiOpenShift.git


978-1-7281-4673-7/19/$31.00 ©2019 IEEE

## II. The Movie Genome Content Descriptors

The proposed Movie Genome system uses state-of-the-art *visual* and *audio* features, tested in several previous works to solve different movie recommendation tasks [7], [9], [10]. To the best of our knowledge, this is one of the first commercial-like movie RS exploiting state-of-the-art audio, visual, and metadata descriptors, while existing systems are limited either to CF or CBF based on metadata. The content descriptors are detailed in the following. Further details can be found in [7].

### A. Audio Features

Our system integrates two kinds of audio features: (i) *block-level features* and (ii) *I-vector features*. Both have been exploited successfully in tasks such as speaker identification, music classification and recommendation [11].

**Block-level features (BLF)** are extracted from audio segments of a few seconds, in contrast to frame-level features which operate on much shorter units. Therefore, BLF can capture temporal aspects of an audio recording to some degree. They have been shown to perform very well in audio and music retrieval and similarity tasks. We use the six features defined in the BLF framework [12], which capture: spectral aspects (spectral pattern, delta spectral pattern, variance delta spectral pattern), harmonic aspects (correlation pattern), rhythmic aspects (logarithmic fluctuation pattern), and tonal aspects (spectral contrast pattern). The extraction process results in a 9,948-dimensional feature vector per video.

**I-vectors** are a state-of-the-art representation learning technique adopted in different audio-related domains [13], such as speech processing, music recommendation, and acoustic scene analysis. An i-vector is a fixed-length and low-dimensional representation containing rich acoustic information. I-vectors are computed based on frame-level features such as Mel-frequency cepstral coefficients (MFCC) which reflect timbral aspects of an audio signal. They are latent variables that capture total variability to represent how much an audio excerpt is shifted from the average excerpt of a collection [7]. Here, we use 20-dimensional MFCCs, 512 components in the GMM, and final i-vectors of dimensionality 200.

### B. Visual Features

Two types of visual features are leveraged in the Movie Genome: (i) *aesthetic visual features* and (ii) *deep learning features* extracted with the AlexNet network [14].

**Aesthetic visual features (AVF)** were originally proposed in [15] for measuring the beauty of coral reefs. They are based on characteristics of image aesthetics in photography and paintings, and are driven by concepts such as image composition, color theory, or interestingness. The features have been grouped into three categories: color-based, texture-based, and object-based, as proposed by [15], adding up to a 107-dimensional vector for each video frame.

**Deep learning features** using AlexNet [14] have won the 2012 Image Net Large Scale Visual Recognition Competition (ILSVRC).[3] For our task, we use the output of the fc7 layer for each frame in the video which has also shown good results in tasks such prediction of media interestingness and emotion recognition. This yields a 4,096 dimensional descriptor vector for each frame. To aggregate the frame-level features into a movie-level descriptor for both types of visual features, we use a standard statistical averaging summarization.

[3] http://www.image-net.org/challenges/LSVRC

### C. Metadata Features

We also include metadata as one of the content categories since it is the one most commonly used. Two types of metadata features are leveraged: (i) genre and (ii) tags, where genre labels are provided by experts (editorial) whereas tags are crafted by users (user-generated). Adoption of these two complementary categories can add semantic value to the user profile created in the recommendation system.

**Genre features**: Each movie in our dataset is labeled with one or more genres out of 18: *Action*, *Adventure*, *Animation*, *Children*, *Comedy*, *Crime*, *Documentary*, *Drama*, *Fantasy*, *Film-Noir*, *Horror*, *Musical*, *Mystery*, *Romance*, *Sci-Fi*, *Thriller*, *War*, and *Western*. The final genre feature vector is a binary 18-dimensional vector.

**Tag features**: Tags are user-generated text labels (keywords) associated to movies or actors. We represent tag features adopting the common vector space model with TF-IDF weights (after a series of prepossessing steps such as punctuation removal, casefolding, stop-word removal and Porter stemming described in [9]). The final tag feature vector is of dimensionality 10,228 per video. Note that both genre and tag features are collected from the MovieLens 20M dataset [16].

## III. The Movie Genome Recommender

The presented *Movie Genome Recommender* is entirely web-based and can be run on a variety of devices, from tablets to smart TVs to laptops to smart phones. Next to its use as a full-fledged content-based movie recommender, it can serve as an evaluation framework for movie recommendation and retrieval algorithms, which can be conveniently set up to conduct controlled experimental studies. The Movie Genome Recommender provides access to a large catalog of thousands of Hollywood trailers, which can be browsed by users who can also obtain descriptive information about each movie. Users provide to the system their preference information about a few movies. They subsequently receive recommendations matching their taste based on the various content attributes of the movie, i.e., the Movie Genome. In the following, we describe the use of the system.

**Step 1: Demographics and background information.** The interaction of the user with the system starts with a sign-up process where she is initially asked to provide her e-mail address, user name, and password. To respect user's privacy, the system allows their users to remain anonymous by providing an option to conceal their true e-mail address. Afterwards, the user is asked to provide basic demographics (age, gender, education, nationality), indicators of her movie consumption behavior (number of movies watched per month, consumption channels), and some optional social media identifiers (such as Facebook and Instagram). This step is shown in Figure 1a.

**Step 2: Five factor personality assessment.** In this step, the user is invited to fill out the Ten Item Personality Inventory (TIPI) questionnaire from which the system can assess her Big Five personality traits, i.e., openness, conscientiousness, extroversion, agreeableness, and neuroticism [17]. This step is illustrated in Figure 1b. While the current version of the system does not implement a personality-aware recommendation engine, the implementation of this questionnaire enables an easy later integration. This step is shown in Figure 1b.

**Step 3: Preference elicitation.** Since the goal of a recommender system is to assist its user with decision making, it is essential to model the user's preferences. At the start of interaction, the user preferences are always incomplete and tend to alter in different contexts. Therefore, in this step, the system collects user preference information to help satisfying her entertainment/information need. Preference elicitation is realized in a two-fold manner. First, the user is invited to select her favorite genre. Then, tailored to the selected genre, she can scroll through productions from different years in a user-friendly manner and is asked to select four movies as her favorites. She can watch the trailers for the selected movies and provide her preference feedback using a 5-level Likert scale. These steps are illustrated in Figures 1d, 1e, and 1f.

**Step 4: User profile generation and recommendation.** The system now creates a user profile based on the preference information obtained in the previous steps to filter the items and generate a ranked list of items by computing the most similar item profiles with respect to the computed user profile. More precisely, the Movie Genome recommendation engine is based on a CBF recommendation model using an item-based nearest neighbor approach [5] where the unknown preference score (i.e., rating) for user $u$ and item $i$ is computed as

$$\hat{\mathbf{r}}_{ui} = \frac{1}{\sum_{j \in N_u(i)} s_{ij}} \sum_{j \in N_u(i)} s_{ij} \, \mathbf{r}_{uj} \tag{1}$$

in which $N_u(i)$ denotes the $k$ items in the profile of user $u$ most similar to item $i$ and $s_{ij}$ is the content similarity score between item $i$ and $j$. We use $k = 10$ and cosine similarity as similarity metric for all features except for the metadata genre features for which Jaccard coefficient is used. Three recommendation lists are generated based on the constituting features of the Movie Genome considering: (i) *only* audio features, (ii) *only* visual features, and (iii) *only* metadata.

Note that in previous work [18], we performed a study and showed that trailers and movies share similar characteristics in a movie recommendation scenario. Based on this finding, our system uses movie trailers instead of the entire movies to extract features, which makes it more versatile and effective as trailers are more easily available than the full movies.

To facilitate execution of empirical studies, the system also supplies a survey containing 22 questions originally adopted from [19] in order to measure the user's perception of the recommendation lists according to different quality metrics (*e.g.,* perceived relevance or novelty) [7]. Finally, we would like to invite the reader to see the proposed system in action, by watching a provided demo video.[4]

## IV. Conclusion and Future Work

In this paper, we described a web implementation of a CBF movie recommender system dubbed *Movie Genome Recommender*. It integrates rich semantic descriptors of the movies' content, including audiovisual features (state-of-the-art audio and image descriptors) and metadata (genre and tags). We call these features the Movie Genome [7].

The Movie Genome Recommender provides functionalities that are available on the vast majority of video-on-demand and streaming services such as Netflix. In addition to the implemented content-based filtering techniques, the system can be easily extended to various context-aware and user-aware scenarios thanks to its modular implementation and the freely available source code and Movie Genome features [10]. For this purpose, sociodemographics and personality questionnaires are already integrated into the web application. So are questionnaires to measure the perceived user experience in terms of beyond-accuracy metrics (e.g., perceived diversity, satisfaction, and novelty). Bringing these functionalities together in one place provides a testbed for personalized and personality-aware recommendation. Implementing respective algorithms as well as latest state-of-the-art models based on deep learning [20], [21] forms part of future work.

## References


[1] Y. Deldjoo, M. Schedl, P. Cremonesi, and G. Pasi, "Content-Based Multimedia Recommendation Systems: Definition and Application Domains," in *Proceedings of the 9th Italian Information Retrieval Workshop, Rome, Italy*, May 2018.

[2] C. C. Aggarwal, "Neighborhood-based Collaborative Filtering," in *Recommender systems*. Springer, 2016, pp. 29–70.

[3] Y. Koren and R. M. Bell, "Advances in Collaborative Filtering," in *Recommender Systems Handbook*. Springer, 2015, pp. 77–118.

[4] P. Lops, M. de Gemmis, and G. Semeraro, "Content-based Recommender Systems: State of the Art and Trends," in *Recommender Systems Handbook*. Springer, 2011, pp. 73–105.

[5] C. C. Aggarwal, "Content-based recommender systems," in *Recommender Systems*. Springer, 2016, pp. 139–166.

[6] D. Bordwell, K. Thompson, and J. Smith, *Film art: An introduction*. McGraw-Hill New York, 1997, vol. 7.

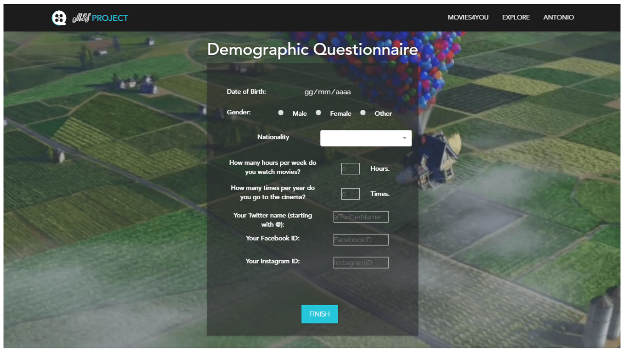
[7] Y. Deldjoo, M. F. Dacrema, M. G. Constantin, H. Eghbal-zadeh, S. Cereda, M. Schedl, B. Ionescu, and P. Cremonesi, "Movie Genome: Alleviating New Item Cold Start in Movie Recommendation," *User Modeling and User-Adapted Interaction*, Feb 2019.

[8] M. Elahi, Y. Deldjoo, F. B. Moghaddam, L. Cella, S. Cereda, and P. Cremonesi, "Exploring the semantic gap for movie recommendations," in *Proceedings of the Eleventh ACM Conference on Recommender Systems, RecSys 2017, Como, Italy*, August 2017, pp. 326–330.

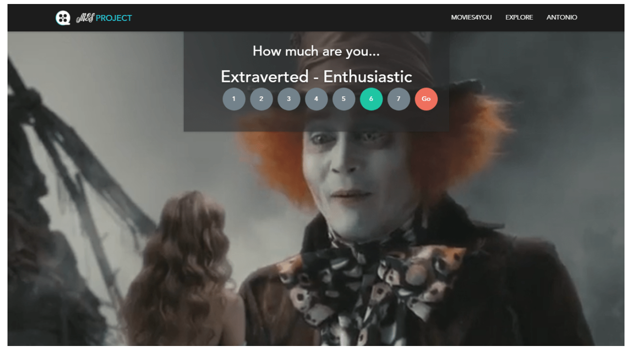
[9] Y. Deldjoo, M. G. Constantin, H. Eghbal-Zadeh, B. Ionescu, M. Schedl, and P. Cremonesi, "Audio-visual Encoding of Multimedia Content for Enhancing Movie Recommendations," in *Proceedings of the 12th ACM Conference on Recommender Systems, RecSys 2018, Vancouver, BC, Canada*, October 2018, pp. 455–459.

[10] Y. Deldjoo, M. G. Constantin, B. Ionescu, M. Schedl, and P. Cremonesi, "MMTF-14K: A Multifaceted Movie Trailer Feature Dataset for Recommendation and Retrieval," in *Proceedings of the 9th ACM Multimedia Systems Conference, MMSys 2018, Amsterdam, the Netherlands*, June 2018, pp. 450–455.


[4] https://youtu.be/vgeI4QmhBpQ

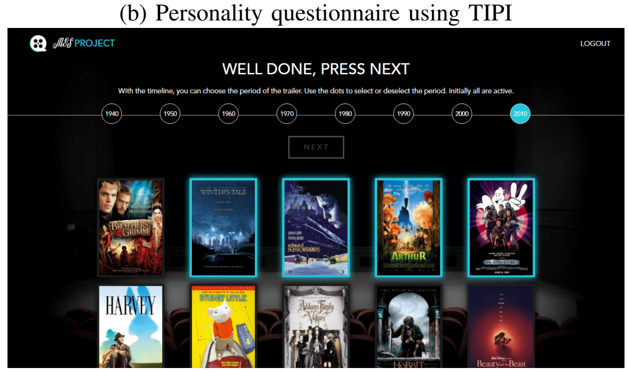



(a) Demographic questionnaire



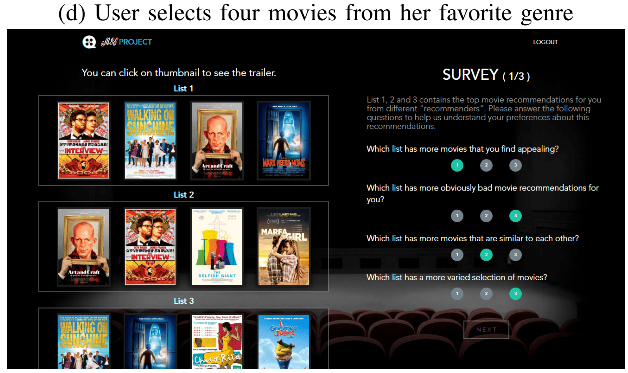

(b) Personality questionnaire using TIPI


(c) User selects her favorite genre




(d) User selects four movies from her favorite genre


(e) User is invited to watch and rate the selected trailers




(f) User receives recommendations

Fig. 1: Selected screenshots of the *Movie Genome Recommender*.


[11] P. Knees and M. Schedl, *Music Similarity and Retrieval - An Introduction to Audio- and Web-based Strategies*, ser. Information Retrieval Series. Springer, 2016, vol. 36.

[12] K. Seyerlehner, G. Widmer, M. Schedl, and P. Knees, "Automatic Music Tag Classification Based on Block-Level Features," in *Proceedings of the 7th Sound and Music Computing Conference, SMC 2010, Barcelona, Spain*, July 2010.

[13] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE TASLP*, vol. 19, no. 4, pp. 788–798, 2011.

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[15] A. F. Haas, M. Guibert, A. Foerschner, S. Calhoun, E. George, M. Hatay, E. Dinsdale, S. A. Sandin, J. E. Smith, M. J. Vermeij *et al.*, "Can We Measure Beauty? Computational Evaluation of Coral Reef Aesthetics," *PeerJ*, vol. 3, p. e1390, 2015.

[16] F. M. Harper and J. A. Konstan, "The MovieLens Datasets: History and Context," *ACM Transactions on Interactive Intelligent Systems (TIIS)*, vol. 5, no. 4, pp. 19:1–19:19, 2016.

[17] R. R. McCrae and O. P. John, "An Introduction to the Five-Factor Model and its Applications," *Journal of Personality*, vol. 60, no. 2, pp. 175–215, 1992.

[18] Y. Deldjoo, M. Elahi, P. Cremonesi, F. Garzotto, P. Piazzolla, and M. Quadrana, "Content-based video recommendation system based on stylistic visual features," *J. Data Semantics*, vol. 5, no. 2, pp. 99–113, 2016.

[19] M. D. Ekstrand, F. M. Harper, M. C. Willemsen, and J. A. Konstan, "User Perception of Differences in Recommender Algorithms," in *Eighth ACM Conference on Recommender Systems, RecSys '14, Foster City, Silicon Valley, CA, USA*, October 2014, pp. 161–168.

[20] H. Chen, Y. Wu, M. Hor, and C. Tang, "Fully Content-based Movie Recommender System With Feature Extraction Using Neural Network," in *2017 International Conference on Machine Learning and Cybernetics, ICMLC 2017, Ningbo, China*, July 2017, pp. 504–509.

[21] B. Hidasi, M. Quadrana, A. Karatzoglou, and D. Tikk, "Parallel Recurrent Neural Network Architectures for Feature-rich Session-based Recommendations," in *Proceedings of the 10th ACM Conference on Recommender Systems, Boston, MA, USA*, September 2016, pp. 241–248.