Hybrid Recommendation of Movies based on Deep Content Features

Preprint · November 2021

3 authors:

Tord Kvifte
University of Bergen

Mehdi Elahi
University of Bergen

Christoph Trattner
University of Bergen

Hybrid Recommendation of Movies based on Deep Content Features

Tord Kvifte, Mehdi Elahi, and Christoph Trattner

MediaFutures: Research Centre for Responsible Media Technology & Innovation,
Department of Information Science & Media Studies, University of Bergen,
Fosswinckels gate 6, 5007 Bergen, Norway Tord.Kvifte@uib.no,
mehdi.elahi@uib.no, Christoph.Trattner@uib.no

Abstract. When a movie is uploaded to a movie Recommender System (e.g., YouTube), the system can exploit various forms of descriptive features (e.g., tags and genre) in order to generate personalized recommendation for users. However, there are situations where the descriptive features are missing or very limited and the system may fail to include such a movie in the recommendation list. This paper investigates hybrid recommendation based on a novel form of content features, extracted from movies, in order to generate recommendation for users. Such features represent the visual aspects of movies, based on Deep Learning models, and hence, do not require any human annotation when extracted. We have evaluated our proposed technique using a large dataset of movies and shown that automatically extracted visual features can mitigate the cold-start problem by generating recommendation with a superior quality compared to different baselines, including recommendation based on human-annotated features.

Keywords: Recommender systems · Visually-aware · New item.

1 Introduction

Recommender systems are intelligent tools that can support users in their decision making process by suggesting a shortlisted set of items tailored to their personal needs and constraints [21, 39, 28, 1]. These systems can learn from the particular tastes and interests of the users and generate recommendation that can better match their interests and tastes [38, 13].

There exists a wide range of approaches that can be adopted to create personalized recommendations for users. Content-Based Filtering (CBF) is among popular approaches that can exploit the content features associated to videos (e.g., tag, and genre) and recommends to a target user the videos with the content similar to the videos that she liked in the past [5, 34, 40, 45]. Collaborative Filtering (CF), on the other hand, is another popular approach which focuses on exploiting patterns among the user preferences (e.g., ratings or likes) and recommends to a target user those videos that have been highly co-rated by like-minded users similar to her [22, 23, 11, 48].

While either of these approaches can be effective in generating relevant recommendation for users, they may fall short to recommend videos whose descriptive data is missing or very limited and hence the system do not have sufficient information about those videos [14, 16]. This is a common problem in recommender systems called *New Item* as part of a bigger challenge called *Cold Start*. New item problem in video streaming applications happens when a new video has been uploaded to the system where the users have not provided neither rating nor any other form of the data, e.g., tag or comments. In such a case, almost all recommender approaches may fail to include such a video when generating personalized recommendation for users. Apart from the new item problem, the process of collecting quality data to represent the videos is itself another major problem. Some forms of data (e.g., genre), a group of experts are essentially required to manually annotate every, and other forms (e.g., rating and tag) may need a large community of users willing to provide the data. This makes the aforementioned data to be very expensive and extremely sparse to collect [9, 30, 4, 3, 46].

In this paper, we address the above-mentioned problems by proposing a novel recommendation technique that exploits visual features to generate personalized recommendation for users. We have adopted a hybrid Matrix Factorisation (MF) algorithm [24], implementing different optimization methods, i.e., *BPR, WARP,* and *Logistic*. The proposed visual features can be extracted in a completely automatic way, using Deep Learning models and hence they require no (expensive) user annotation. This enables our proposed technique to effectively cope with the cold-start problem, when no or limited human-annotated data is available.

We have extracted a large dataset of visual features from 12,875 of the trailers of the movies that exist in the Movielens dataset. Movie trailers have shown to exhibit high visual similarity compared to their full length movies [8]. In addition to visual features, we have also collected a rich dataset of movie subtitles and generated recommendation based on them and considered it as one of the baselines. We evaluated our proposed recommendation technique using the dataset with hundreds of thousands of ratings. The results show the superior performance of our proposed technique compared with a number of baselines, i.e., recommendation based on tag, genre, and subtitle.

The main contributions of this work can be summarized as follows:

1. The proposal of a novel hybrid recommendation technique based on visual features considering different optimization methods. e.g., *BPR, WARP,* and *Logistic*, and comparing it with different baselines with regards to different evaluation metrics;
2. extracting a large dataset with visual features, using an advanced deep learning model; Dataset will be published publicly upon the acceptance of the paper;
3. collecting a large dataset of subtitles from full length movies and exploiting them in a baseline recommendation technique.

2 Related work

One of the most popular types of recommender systems are based on the Content-based Filtering (CBF) technique. In this technique, the items are represented by their content and the users by associating their preferences with the item content [28, 21, 29, 36, 18]. In movie domain, the item content are described with a set of representative features describing different aspects of the movie content. Traditional examples of content features are genre and tag, representing some form of *semantics* within the movies.

Recent approaches based on content-based filtering have adopted a novel form of movie content based on visual features [49, 8, 15] illustrating a more *stylistic* representation of the movies. This type of novel features, in contrast to the traditional features, does not need any expensive human-annotation and can be extracted automatically adopting *Computer Vision* methods. Hence, they could be a potential solution for movie recommendation in cold start, i.e., when recommending movies with no descriptive features[12]. Another advantage of the visual features is that they can be more representative of the production style and can enable movie recommender systems to become *style-aware* [26, 19, 50, 2].

Visual features, extracted from movie content, can have different classes, each of which illustrating a different representation of the movies [31]. One class of visual features can describe movies from a *high-level* perspective while another class can describe them from a *low-level* perspective. The former type of features typically provide a more semantic representation of the movies (e.g., sun shining in the a movie scene) while the latter type focus more on low level aspects (e.g., colorfulness and brightness in a movie).

A number of prior work have proposed recommender systems capable of using visual features. As an example, the authors in [49] proposed a recommendation approach by combining semantic and visual content features. Another example is [50] that proposed integration of multiple ranking lists, each of which generated by a set of semantic or visual features. The authors of [7] proposed a recommendation technique based on a selection of handcrafted visual features including shot length, object motion, color, and lighting. [41] is another work where authors explored the different potentials of visual features in movie recommender systems. In [6, 42] , a set of audio-visual features have been exploited to generate movie recommendation. In [27] and [35], the authors proposed a video recommender system that takes advantage of Deep Learning methods based on Convolutional Neural Networks (CNN). Finally, few prior works attempted to address the research gap between video classification, and search & recommendation by proposing a more unified solutions. An example is [25] where the authors proposed a model based on a deep learning approach (i.e., CNN) utilizing a set of audio-visual features and showed to be effective in the noted tasks.

Our work differs compared to the work mentioned above in the following aspects. First of all, these works adopted a one-size-fits-all approach by considering a single optimization method when building their recommendation model. However, different methods may better suit different type of content data (e.g., visual features, genre and tag). Hence, we adopted different optimization methods, based on different loss functions, for different types of data. We have used a large dataset of movies and compared the performances of different optimization methods for the task of recommendation. To the best of our knowledge, non of the prior works has performed such a comparison. Furthermore, we have considered a novel baseline, i.e., recommendation based on movie subtitle and compared it with our proposed recommendation technique (visual features) as well as more traditional baselines (genre and tags) taking into account different evaluation metrics, i.e., Precision@K, Recall@K, AUC, and Reciprocal Rank.

3 Methodology

We used a large dataset of key-frames from 12875 movie trailers collected from YouTube. According to prior work, there is a high similarity between the visual features extracted from full-length movies and their respective movie trailers [8]. The following list represents the entire methodology: *Extracting Visual Features*: Every key-frame is analyzed using a pre-trained CNN model [44], resulting in feature labels. *Aggregating Features*: Visual features are aggregated using two different methods, resulting in two different sets of feature vectors. *Training and predicting*: The feature vectors are used to train the prediction models.

3.1 Feature Extraction

Our feature extraction can be divided into two parts. First part includes the extraction of visual features from movie trailers, and the second part encompasses the collection of movie subtitles.

Visual Feature Extraction. We extracted visual feature labels by applying the VGG-19 image classification model [44], a 19-layer network trained on ImageNet, to the key-frames of every movie trailer in the key-frame dataset. The model was implemented in Python, using the Keras API, which is built on top of the TensorFlow framework [32]. The output of the model consists of a label, representing the predicted classes of the input image, as well as a confidence value representing the certainty of the prediction being correct. The resulting dataset of labels for 12,875 movies includes 997 unique feature labels in total.

Subtitle Collection and Pre-Processing. Subtitles were collected using a public API [33] [1], then parsed and pre-possessed, resulting in a dataset of English subtitles from 1514 different movies. Among the pre-processing steps were removal of timestamps and subtitle-specific data, stop word removal, part-of-speech filtering, and lemmatization. The resulting dataset includes 62664 unique features.

[1] http://www.opensubtitles.org

3.2 Feature Aggregation

To form the final feature embeddings of a movie, we have aggregated the extracted features. Visual features were aggregated using two different methods, producing two separate feature matrices, *Deep Visual-f* and *Deep Visual-c*.

Deep Visual-f. Visual features were weighted using *Term Frequency–Inverse Document Frequency (TF-IDF)* [43]. TF-IDF can recognize the importance of each word in a document in the context of a corpus of documents. If a word has low occurrence across the corpus, while having high frequency in one (or few) document, it likely plays a key role in that specific document. In our case, a movie is considered as a document, and the labels of the movie are considered as words of that document. Furthermore, the collection of all movies and their respective labels corresponds to the corpus of documents.

Deep Visual-c. Important elements in a movie can be assumed to be emphasized visually, and thereby more likely to be predicted with a higher confidence, computed by the image classification model. Based on this assumption, visual features were weighted according to the mean confidence value of each label occurring in a movie.

Subtitles. Subtitle features were weighted using the frequency of the words, occurring in subtitles for different movies, and normalized afterwards by applying *min-max* normalization.

3.3 Recommendation Algorithm

We built a hybrid recommender system that extends the Matrix Factorization model and enables it to exploit different types of data. Hence, the recommender system has become capable of using heterogeneous data including different types of side information (visual features & genre of movies, ratings & tags of users). The implementation of the hybrid recommender algorithm has been done using a popular library, i.e., *LightFM* [24]. The hybrid recommender system can learn the latent embeddings for users and items and encodes the user preferences over items. When these representations are multiplied together, they create scores for every item given a user. Representations of users and items are expressed by representations of their features. Feature representations are derived at by estimating an embedding for every feature and summing the embeddings together to arrive at user and item representations. The embeddings are learned with the use of stochastic gradient descent methods.

We considered different optimization methods with different loss functions: *Weighted Approximate-Rank Pairwise (WARP)* [47], *Bayesian Personalized Ranking (BPR)* [37], and *logistic loss*. The WARP loss function is defined as [47, 20]:

$$Err_{WARP}(\mathrm{x}_i, y_i) = L\left[rank(f(y_i|\mathrm{x}_i))\right] \tag{1}$$

where the function $rank(f(y_i|\mathrm{x}_i))$ measures the number of negative labelled instances that are "wrongly" given a higher rank than this positive example x_i :

$$rank(f(y_i|\mathrm{x}_i)) = \sum_{(\mathrm{x}',y')\in C_u^-} I\left[f(y'|\mathrm{x}') \geq f(y|\mathrm{x}_i)\right] \tag{2}$$

where $I(\mathrm{x})$ is the indicator function, and $L(\cdot)$ transforms this rank into a loss:

$$L(r) = \sum_{j=1}^{r} \tau_j, with \tau_1 \geq \tau_2 \geq \cdots \geq 0. \tag{3}$$

This class of functions allows one to define different choices of $L(\cdot)$ with different minimizers. Minimizing L with $\tau_1 = 1$ and $\tau_{i>1} = 0$, the precision at 1 is optimized, $\tau_j = \frac{1}{Y-1}$ would optimize the mean rank, while for $\tau_{i\leq k} = 1$ and $\tau_{i>k} = 0$ the precision at k is optimized. For $\tau_i = 1/i$ a smooth weighing is given, where the top position is given more weight, with rapidly decreasing weight for lower positions. This is useful when opimizing Precision@K for a range of different values at K is desirable.

BPR [37] is one of the state-of-the-art algorithms exploit homogeneous implicit feedbacks. It assumes that a user prefers a consumed item to an unconsumed item, denoted as $(u, i) \succ (u, j) or \hat{r}_{uij} > 0$. Mathematically, BPR solves the following minimization problem [37]:

$$\min_{\Theta} \sum_{(u,i,j):(u,i)\succ(u,j)} f_{uij}(\Theta) + \mathcal{R}_{uij}(\Theta) \tag{4}$$

where the loss function $f_{uij}(\Theta) = -ln\sigma(\hat{r}_{uij})$ is designed to encourage pairwise competition with $\sigma(\mathrm{x}) = 1/(1 + \exp(-\mathrm{x}))$ and $\hat{r}_{uij} = \hat{r}_{ui} - \hat{r}_{uj}$. Note that $\mathcal{R}_{uij}(\Theta) = \propto 2 \left\|U_{u\cdot}\right\|^2 + \propto 2(\left\|V_{i\cdot}\right\|^2 + \left\|V_{j\cdot}\right\|^2) + \propto 2(\left\|B_i\right\|^2 + \left\|B_j\right\|^2)$ is the regularization term used to prevent overfitting, and $\hat{r}_{ui} = \langle U_{u\cdot}, V_{i\cdot}\rangle + b_i$ is the prediction rule based on user u's latent feature vector $U_{u\cdot} \in R^{1\times d}$, item i's latent feature vector $V_{i\cdot} \in R^{1\times d}$ and item bias $B_i \in R$.

4 Experiments and Results

4.1 Evaluation Methodology

We have evaluated our proposed recommendation technique based on (automatic) visual features considering different optimization methods, i.e., WARP, BPR, and logistic loss functions utilizing both item features and user interactions. Each model was trained on one of two types of automatic features (i.e., item embeddings), namely *Deep Visual-f*, *Deep Visual-c*. For the baselines we, have considered recommendation based on *subtitles*, *tags*, or *genre*. While subtitle can be automatically extracted, both genre and tags requires human-annotation. In addition to item features, MovieLens1M dataset [17] has been utilized. In order to simulate the cold-start scenario, we have randomly sampled the dataset. The final result contained 272,515 ratings for 1514 items provided by 6040 users.

The train and test sets were built by following a hold-out methodology, i.e., randomly splitting the dataset into 80% (train) and 20% (test) disjoint subsets. The proposed recommendation models have been trained using the train set and evaluated using the test set. Hyperparameter tuning has been performed using a random search to fit LightFM models with random hyperparameter values and evaluating the model performance on the validation set. Based on the hyperparameter tuning result, models were trained over 25 epochs with AdaGrad [10] as learning rate schedule and learning rate of 0.06.

Feature	Type	Precision@K	Recall@K	AUC	Reciprocal Rank
Tag	*manual*	0.027	0.080	0.518	0.084
Genre	*manual*	0.040	0.024	0.698	0.118
Subtitle	*automatic*	0.070	0.048	0.849	0.179
Deep Visual-c	*automatic*	0.157	0.103	0.846	**0.342**
Deep Visual-f	*automatic*	**0.166**	**0.109**	**0.860**	0.354

Table 1: Comparison of the recommendation quality based on automatic features and manual features.

4.2 Experiment A: Recommendation Quality

In the first set of experiments, we have measured the quality of the recommendation based on automatic visual features, extracted by the deep learning model. Figure 1 represents the results obtained in this experiment.

First of all, as it can be seen, both version of our proposed recommendation technique (Deep Visual-f and Deep Visual-c), based on visual features, outperform all the other different baselines. In terms of Precision@K, Deep Visual-f achieves the score of 0.166 and Deep Visual-c achieves score of 0.157. The next best precision score is obtained by recommendation based on movie subtitles with the score of 0.070, where recommendation based on manual features, i.e., genre and tag, received the lowest scores, i.e., 0.040 and 0.027, respectively. In terms of Recall@K, similarly, both Deep Visual-f and Deep Visual-c achieved the best results with the scores of 0.109 and 0.103, respectively. The next best performance has been observed for recommendation based on the subtitle with the score of 0.048. The recommendation based on genre and tag have performed the worst with the scores of 0.24 and 0.080, respectively.

In terms of AUC, recommendation based on subtitle has achieved a great score of 0.849, however, Deep Visual-f still has obtained the best score of 0.860. Recommendation based Deep Visual-c has obtained the next best result with the score of 0.846. Recommendation based on genre and tag have received the lowest scores, i.e., 0.698 and 0.518, respectively. Finally, in terms of Reciprocal

Rank, again, proposed recommendation technique based on either Deep Visual-f and Deep Visual-c has achieved the highest scores. While the observed scores for Deep Visual-f and Deep Visual-c were 0.354 and 0.341, the next best score was almost half of these values, observed for recommendation based on subtitle with a score of 0.179. As expected, both genre and tag have shown the worst performance with the scores of 0.118 and 0.084.

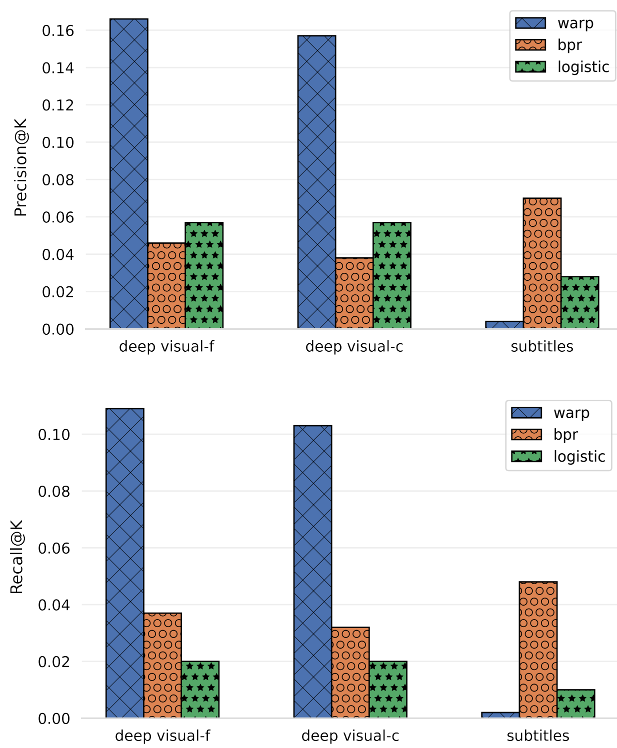

Fig. 1: Comparison of recommendation based on automatic features using different optimization methods in terms of (top) Precision and (bottom) Recall.

4.3 Experiment B: Comparing Loss Functions

In the second set of experiments, we have compared the recommendation based on automatic features when different types of optimization algorithms have used. The results have been illustrated in Figure 2 and 3.

First of all, as it can be seen, different loss function (hence optimization algorithm) can yield different recommendation quality for each type of automatic features. For the visual features, either deep visual-c or deep visual-f, the best results have been achieved using *warp* loss function, considering all metrics, i.e., Precision@K, Recall@K, AUC, and Reciprocal Rank. Surprisingly, *bpr* loss function does not perform well and in some cases (e.g., Precision) it yields the worst results.

For the subtitle features, on the other hand, the best results have been achieved by bpr loss function for all metrics. In contrary, the worst results are obtained by warp loss function. This is another surprising result as both types of visual and subtitle features are of categorical type and might be expected to share similarities in their nature. However, apparently, they represent different aspects of the videos that are perhaps different and hence shall be handled differently.

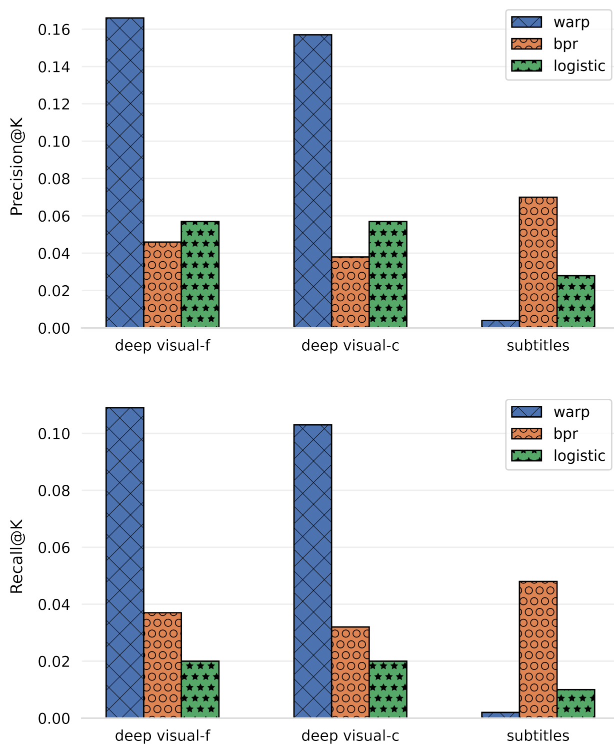

Fig. 2: Comparison of recommendation based on automatic features using different optimization methods in terms of (top) Precision and (bottom) Recall.

Overall, these promising results have shown the excellent performance of hybrid recommendation based on visual features, using different optimization methods. The results have clearly illustrated the substantial potential behind these features that can exploited when no other types of content features are provided to a movie recommender system.

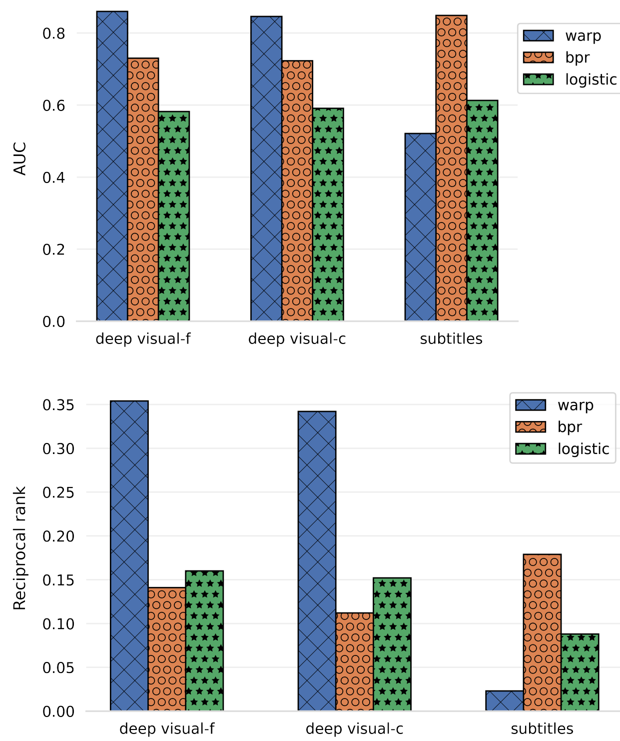

Fig. 3: Comparison of recommendation based on different automatic features using different optimization methods in terms of (top) AUC and (bottom) Reciprocal Rank.

5 Conclusions and Future Work

This paper focuses on the new item problem as part of cold start in recommender systems and proposes a hybrid technique to generate recommendation based on visual features, automatically extracted from movies. The visual features have been extracted using a deep learning network (i.e., CNN) and exploited to generate movie recommendation. The proposed technique can be fully automated and does not require any human involvement and hence can be utilized when recommending movies that have neither any rating nor content features.

The proposed hybrid technique has been evaluated using a large dataset of movie trailers and compared against recommendation based on other features, i.e., subtitle, genre and tags. The results have shown that our proposed recommendation technique can outperform the other techniques with regards to all the evaluation metrics.

In future, we would like to extend these experiments by taking into account the datasets, collected from other social networks (e.g., Instagram). In addition to that we will extend our feature set by considering other types of features that can be extracted automatically. Finally, we will adopt other feature fusions when aggregating the visual features.

6 Acknowledgements

This work was supported by industry partners and the Research Council of Norway with funding to MediaFutures: Research Centre for Responsible Media Technology and Innovation, through The Centres for Research-based Innovation scheme, project number 309339.

References

1. Charu C Aggarwal et al. *Recommender systems*, volume 1. Springer, 2016.
2. Luca Canini, Sergio Benini, and Riccardo Leonardi. Affective recommendation of movies based on selected connotative features. *Circuits and Systems for Video Technology, IEEE Transactions on*, 23(4):636–647, 2013.
3. Iván Cantador, Alejandro Bellogín, and David Vallet. Content-based recommendation in social tagging systems. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 237–240. ACM, 2010.
4. Iván Cantador, Ioannis Konstas, and Joemon M Jose. Categorising social tags to improve folksonomy-based recommendations. *Web semantics: science, services and agents on the World Wide Web*, 9(1):1–15, 2011.
5. Marco de Gemmis, Pasquale Lops, Cataldo Musto, Fedelucio Narducci, and Giovanni Semeraro. Semantics-aware content-based recommender systems. In *Recommender Systems Handbook*, pages 119–159. Springer, 2015.
6. Yashar Deldjoo, Mihai Gabriel Constantin, Hamid Eghbal-Zadeh, Bogdan Ionescu, Markus Schedl, and Paolo Cremonesi. Audio-visual encoding of multimedia content for enhancing movie recommendations. In *Proceedings of the 12th ACM Conference on Recommender Systems*, RecSys '18, page 455–459, New York, NY, USA, 2018. Association for Computing Machinery.
7. Yashar Deldjoo, Mehdi Elahi, P. Cremonesi, Franca Garzotto, Pietro Piazzolla, and Massimo Quadrana. Content-Based Video Recommendation System Based on Stylistic Visual Features. *Journal on Data Semantics*, 5:99–113, 2016.
8. Yashar Deldjoo, Mehdi Elahi, Paolo Cremonesi, Franca Garzotto, Pietro Piazzolla, and Massimo Quadrana. Content-based video recommendation system based on stylistic visual features. *Journal on Data Semantics*, pages 1–15, 2016.
9. Tommaso Di Noia, Roberto Mirizzi, Vito Claudio Ostuni, Davide Romito, and Markus Zanker. Linked open data to support content-based recommender systems. In *Proceedings of the 8th International Conference on Semantic Systems*, pages 1–8. ACM, 2012.

10. John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. 12(null):2121–2159, July 2011.
11. Mehdi Elahi. *Empirical evaluation of active learning strategies in collaborative filtering*. PhD thesis, Ph. D. Dissertation. Ph. D. Dissertation. Free University of Bozen-Bolzano, 2014.
12. Mehdi Elahi, Farshad Bakhshandegan Moghaddam, Reza Hosseini, Mohammad Hossein Rimaz, Nabil El Ioini, Marko Tkalcic, Christoph Trattner, and Tammam Tillo. Recommending videos in cold start with automatic visual tags. In *Adjunct Proceedings of the 29th ACM Conference on User Modeling, Adaptation and Personalization*, pages 54–60, 2021.
13. Mehdi Elahi, Amin Beheshti, and Srinivasa Reddy Goluguri. Recommender systems: Challenges and opportunities in the age of big data and artificial intelligence. In *Data Science and Its Applications*, pages 15–39. Chapman and Hall/CRC, 2021.
14. Mehdi Elahi, Matthias Braunhofer, Tural Gurbanov, and Francesco Ricci. *User Preference Elicitation, Rating Sparsity and Cold Start: Algorithms*, pages 253–294. 11 2018.
15. Mehdi Elahi, Yashar Deldjoo, Farshad Bakhshandegan Moghaddam, Leonardo Cella, Stefano Cereda, and Paolo Cremonesi. Exploring the semantic gap for movie recommendations. In *Proceedings of the Eleventh ACM Conference on Recommender Systems*, pages 326–330. ACM, 2017.
16. Mehdi Elahi, Francesco Ricci, and Neil Rubens. Active learning strategies for rating elicitation in collaborative filtering: a system-wide perspective. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 5(1):13, 2013.
17. F Maxwell Harper and Joseph A Konstan. The movielens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 5(4):19, 2016.
18. Bilal Hawashin, Mohammad Lafi, Tarek Kanan, and Ayman Mansour. An efficient hybrid similarity measure based on user interests for recommender systems. *Expert Systems*, page e12471, 2019.
19. Naieme Hazrati and Mehdi Elahi. Addressing the new item problem in video recommender systems by incorporation of visual features with restricted boltzmann machines. *Expert Systems*, 38(3):e12645, 2021.
20. Liang Jie Hong. Pairwise loss (warp), 2012. Online; accessed 2021-01-21.
21. Dietmar Jannach, Markus Zanker, Alexander Felfernig, and Gerhard Friedrich. *Recommender Systems: An Introduction*. Cambridge University Press, 2010.
22. Yehuda Koren and Robert Bell. Advances in collaborative filtering. In Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul Kantor, editors, *Recommender Systems Handbook*, pages 145–186. Springer Verlag, 2011.
23. Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8), 2009.
24. Maciej Kula. Metadata embeddings for user and item cold-start recommendations. In Toine Bogers and Marijn Koolen, editors, *Proceedings of the 2nd Workshop on New Trends on Content-Based Recommender Systems co-located with 9th ACM Conference on Recommender Systems (RecSys 2015), Vienna, Austria, September 16-20, 2015.*, volume 1448 of *CEUR Workshop Proceedings*, pages 14–21. CEUR-WS.org, 2015.
25. Joonseok Lee, Sami Abu-El-Haija, Balakrishnan Varadarajan, and Apostol (Paul) Natsev. Collaborative deep metric learning for video understanding. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '18, page 481–490, New York, NY, USA, 2018. Association for Computing Machinery.

26. Taras Lehinevych, Nikolaos Kokkinis-Ntrenis, Giorgos Siantikos, A Seza Dogruöz, Theodoros Giannakopoulos, and Stasinos Konstantopoulos. Discovering similarities for content-based recommendation and browsing in multimedia collections. In *Signal-Image Technology and Internet-Based Systems (SITIS), 2014 Tenth International Conference on*, pages 237–243. IEEE, 2014.
27. Y. Li, H. Wang, H. Liu, and B. Chen. A study on content-based video recommendation. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 4581–4585, 2017.
28. Pasquale Lops, Marco De Gemmis, and Giovanni Semeraro. Content-based recommender systems: State of the art and trends. In *Recommender systems handbook*, pages 73–105. Springer, 2011.
29. Eder F Martins, Fabiano M Belém, Jussara M Almeida, and Marcos A Gonçalves. On cold start for associative tag recommendation. *Journal of the Association for Information Science and Technology*, 67(1):83–105, 2016.
30. Aleksandra Klasnja Milicevic, Alexandros Nanopoulos, and Mirjana Ivanovic. Social tagging in recommender systems: a survey of the state-of-the-art and possible extensions. *Artificial Intelligence Review*, 33(3):187–209, 2010.
31. Farshad B Moghaddam, Mehdi Elahi, Reza Hosseini, Christoph Trattner, and Marko Tkalčič. Predicting movie popularity and ratings with visual features. In *2019 14th International Workshop on Semantic and Social Media Adaptation and Personalization (SMAP)*, pages 1–6. IEEE, 2019.
32. Open-source and Google. Keras, 2020. Online; accessed 2021-01-21.
33. opensubtitles.org. Opensubtitles, 2020. Online; accessed 2020-11-01.
34. Michael J. Pazzani and Daniel Billsus. The adaptive web. chapter Content-based Recommendation Systems, pages 325–341. Springer-Verlag, Berlin, Heidelberg, 2007.
35. Ralph Jose Rassweiler Filho, Jonatas Wehrmann, and Rodrigo C Barros. Leveraging deep visual features for content-based movie recommender systems. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 604–611. IEEE, 2017.
36. Sahin Renckes, Huseyin Polat, and Yusuf Oysal. A new hybrid recommendation algorithm with privacy. *Expert Systems*, 29(1):39–55, 2012.
37. Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence*, pages 452–461. AUAI Press, 2009.
38. Paul Resnick and Hal R. Varian. Recommender systems. *Commun. ACM*, 40(3):56–58, 1997.
39. Francesco Ricci, Lior Rokach, and Bracha Shapira. Recommender systems: Introduction and challenges. In *Recommender Systems Handbook*, pages 1–34. Springer US, 2015.
40. Mohammad Hossein Rimaz, Mehdi Elahi, Farshad Bakhshandegan Moghadam, Christoph Trattner, Reza Hosseini, and Marko Tkalčizzzzzzzzzzzzzzzzzz73269. Exploring the power of visual features for the recommendation of movies. In *Proceedings of the 27th ACM Conference on User Modeling, Adaptation and Personalization*, pages 303–308, 2019.
41. Mohammad Hossein Rimaz, Mehdi Elahi, Farshad Bakhshandegan Moghadam, Christoph Trattner, Reza Hosseini, and Marko Tkalčič. Exploring the power of visual features for the recommendation of movies. *ACM UMAP 2019 - Proceedings of the 27th ACM Conference on User Modeling, Adaptation and Personalization*, (June):303–308, 2019.

42. Mohammad Hossein Rimaz, Reza Hosseini, Mehdi Elahi, and Farshad Bakhshandegan Moghaddam. Audiolens: Audio-aware video recommendation for mitigating new item problem. In *International Conference on Service-Oriented Computing*, pages 365–378. Springer, 2020.
43. S. E. Robertson and K. Sparck Jones. Relevance weighting of search terms. *Journal of the American Society for Information Science*, 27(3):129–146, 1976.
44. Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, pages 1–14, 2015.
45. Márcio Soares and Paula Viana. Tuning metadata for better movie content-based recommendation systems. *Multimedia Tools and Applications*, 74(17):7015–7036, 2015.
46. Lichuan Wang, Xianyi Zeng, Ludovic Koehl, and Yan Chen. Intelligent fashion recommender system: Fuzzy logic in personalized garment design. *IEEE Trans. Human-Machine Systems*, 45(1):95–109, 2015.
47. Jason Weston, Samy Bengio, and Nicolas Usunier. Wsabie: Scaling up to large vocabulary image annotation. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume Three*, IJCAI'11, page 2764–2770. AAAI Press, 2011.
48. Bo Yang, Yu Lei, Jiming Liu, and Wenjie Li. Social collaborative filtering by trust. *IEEE transactions on pattern analysis and machine intelligence*, 39(8):1633–1647, 2016.
49. Bo Yang, Tao Mei, Xian-Sheng Hua, Linjun Yang, Shi-Qiang Yang, and Mingjing Li. Online video recommendation based on multimodal fusion and relevance feedback. In *Proceedings of the 6th ACM international conference on Image and video retrieval*, pages 73–80. ACM, 2007.
50. Xiaojian Zhao, Guangda Li, Meng Wang, Jin Yuan, Zheng-Jun Zha, Zhoujun Li, and Tat-Seng Chua. Integrating rich information for video recommendation with multi-task rank aggregation. In *Proceedings of the 19th ACM international conference on Multimedia*, pages 1521–1524. ACM, 2011.