



SemVidRec: A Semantic Approach to Annotations Driven Video Recommendation Model Incorporating Machine Intelligence

Arya Adesh¹, Gerard Deepak²(✉), and A. Santhanavijayan³

¹ Department of Computer Science Engineering, RV College of Engineering, Bengaluru, India

² Department of Computer Science Engineering, Manipal Institute of Technology Bengaluru, Manipal Academy of Higher Education, Manipal, India
gerard.deepak.christuni@gmail.com

³ Department of Computer Science Engineering, National Institute of Technology, Tiruchirappalli, India

Abstract. Video recommendation ensures that viewers get content more relevant to their choices and taste. With the aggregation of a diverse variety of content on video streaming platforms, there is a need to improve the existing recommendation model to increase accuracy in predicting the best content analogous to past choices of the user. The model proposed in this paper employs semantic similarity to generate recommendations based on metadata and annotations. The semantic inference-based model ensures the incorporation of cognitive knowledge of user inputs and past activities, yielding better insights into user preferences. The model utilizes web-sourced metadata to augment its dataset with supplementary information. The metadata is classified using RNN and is stored in a semantic network for feature selection. Topic modeling on query terms allows the discovery of hidden topics in the user query and the use of a knowledge store like DBpedia allows for enriching the query terms based on the topics discovered. The video dataset is categorized using a combination of Random Forest and decision tree-bagging techniques, which are based on the extracted features of the videos. The proposed SemVidRec model is compared with other video recommendation models like VAVR, PVRRC, CCVR, etc. SemVidRec was discovered to outperform the baseline models based on several performance metrics, achieving an accuracy of 96.45%, precision of 95.43%, and an nDCG value of 0.97.

Keywords: Bagging · Semantic Similarity · Semantic Network · DBpedia · Structural Topic Modeling

1 Introduction

Video recommendations are video-content suggestions made to the user on a video streaming platform, narrowed down to best suit the needs and taste of that user. Video recommendations help filter out videos most likely to be watched, hence separating irrelevant content. This builds up a productive environment for the user, by reducing

time spent on searching and selecting relevant content. Video recommendation systems draw inferences about user preferences based on watch history or search queries. With a rise in popularity and content in video aggregation and streaming platforms, for instance, YouTube and OTTs like Hulu, Amazon Prime, Netflix, and others, there is a need for an efficient and improvised video recommendation model. The two most frequently used recommendation models are content-based filtering and collaborative filtering.

Content-based recommendation model extracts visual features from individual frames of video and suggests content by investigating and matching these features with other videos. Collaborative filtering is a recommended approach that utilizes data obtained from multiple users to suggest items based on the similarity of users' preferences. The hybrid model for recommendation is a combination of the aforementioned video recommendation models, which either cumulatively adds abilities from both models or by merging their separate results.

However, these video recommendation models are not feasible to make predictions, when the video content on the platform is huge. Hence there is a need for a recommendation model that takes in auxiliary data and cognitive insights to make predictions accurately. Annotations-based video recommendation model uses video labels as the basis for prediction. This model can handle huge video content more effectively than content-based or collaborative filtering models. It is also important to make the recommendation model compatible with Semantic web standards. The semantic web has all content organized in machine-understandable form and hence allows the encoding of semantics with data. The inclusion of semantic insight into the recommendation model helps learn about user preferences, better, yielding accurate predictions. In the semantic web, metadata describing the content available in the World Wide Web is machine-understandable. This metadata can be used by the model to include data semantically similar to the query terms, allowing the inclusion of auxiliary information. The model proposed in the paper is completely compatible with Web 3.0 hence more relevant to the real-world scenario.

Motivation: The World Wide Web is extending into the Semantic Web (Web 3.0). Data on the semantic web is linked and exists in the form of a knowledge graph to facilitate machine understandability. This makes Web 3.0 powerful, robust, and agile. The existing recommendation model may not work competently on such an organized web of content, as they are based on older learning paradigms. There is a need for a recommendation model which includes semantic insights for making accurate predictions to improve user engagement. In this study, a video recommendation model is proposed that incorporates machine intelligence, which is a combination of machine learning and artificial intelligence paradigms. This model is designed to cater to the requirements of the Semantic Web. The video recommendation system named SemVidRec, is an ontology-based model to be able to process cohesive and huge data on the web. This improvisation helps.

Contribution: The proposed method introduces several novel contributions. Firstly, a semantic network is derived, which is based on enriched queries from the user history and insights extracted from the user profile. Shannon's entropy is utilized to construct the semantic network based on metadata instances classified by a strong deep-learning

model. Secondly, the method encompasses Structural topic modeling and harvests entities from the DBpedia knowledge store to ensure the relevance and diversity of query terms. Thirdly, a dataset is classified by extracting features from the semantic network by using strong bagging techniques. Fourthly, the method employs semantic similarity models with empirically decided differential thresholds to rank and make relevant recommendations. Finally, testing confirmed a higher percentage of precision, recall, F-Measure, and accuracy while ensuring a very low False Discovery Rate (FDR).

Organization: The paper is structured as follows: Sect. 2 explores Relevant Work, Sect. 3 provides an overview of the Planned System Architecture and details the implementation of the SemVidRec model. Section 4 highlights the Findings and Evaluation of the model's performance. This paper concludes in Sect. 5.

2 Related Work

Elahi et al. [1] proposes a model to solve the moderate cold-start problem as it extracts visual features from individual frames of a video using color histogram distance. Features extracted from the individual frames are aggregated using aggregation functions and recommendation.

Is done based on the K-Nearest Neighbor algorithm. Cosine similarity is used on the nearest neighbor set and the predicted preference score is calculated. Du et al. [2] proposes a model that uses both textual and non-textual features from the videos which are fused with priority-based late fusion. Collaborative embedded regression was used to overcome the drawback of the unavailability of one specific content. The proposed model is designed to operate in both in-matrix and out-of-matrix scenarios.

Zu et al. [3] Proposed a model to find suitable resources for online learning. The model constructs a subgraph on a seed set of recommendations. By constructing a cross-curriculum video-associated knowledge map and applying a random walk algorithm, the model recommends relevant subgraphs of course videos to learners. Vimala et al. [4] proposes a Kullback-Leibler divergence-based fuzzy C-means clustering method with improved square root cosine similarity to enhance the accuracy of collaborative filtering movie recommendation systems. To enhance the efficiency of the suggested approach, Support Vector Machines (SVM) are utilized to achieve more precise predictions. For comparison Fuzzy+ SVM+ Cosine model which is an improvisation of the model proposed by Vimala et al. [4], is used as one of the baseline models.

Yan et al. [5] proposes a hybrid collaborative filtering system based on a deep autoencoder model which intends to solve cold-start recommendation problems based on users' viewing behavior. The user/item information is processed by data processing and embedded layers, the auto-encoder layer is responsible for mining implicit features and the correlation between them. The final prediction is done by mapping the target rating vector with the feature matrix in the multi-layer perceptron layer. Cai et al. [6] presents an approach to active learning for video recommendation, which uses a multi-view strategy that leverages the visual characteristics of a video while requiring few annotations. A visual-to-text mapping function is utilized to map visual features and textual views to minimize classification loss. The model uses watching frequency and prediction inconsistency to select videos for metadata querying. Ma et al. [7] put forward a model for

suggesting micro-videos available on social networks. This model uses user-item interaction features, textual features, visual features extracted from videos. Using a deep neural network-based latent genre learning, it is possible to identify the concealed genres of micro-videos. This approach is proposed to enhance the recommendation quality by recognizing hidden patterns within the content, enabling more precise suggestions for the viewer.

Bhatt et al. [8] put forward a content-based recommendation model for online course content that uses sequential pattern mining of inter-topic relationships. This approach is proposed to create a personalized learning experience and to suggest courses that are relevant to the individual's interests based on their past interactions. The inter-topic relationships are mined from the instructor's syllabi as they tend to exist in educational corpora. The prediction is done based on content similarity between videos determined by Topic Similarity Score, Global, and Local sequence scores. Mei et al. [9] suggests an innovative recommendation system for online videos, which utilizes a collection of relevant videos based on their multimodal relevance (written, graphic, and audio components) and user clicks. A relevance mechanism is used to find and assign optimum weights by user clicks as videos have different intra-weight of relevance. An attention fusion function combines multimodal relevance and recommendations are made accordingly. Vellaichamy et al. [10] proposes a hybrid Collaborative Movie Recommender system that combines Fuzzy C Means clustering with Bat optimization to address the problem of handling huge volumes of data and to enhance clustering accuracy. The system clusters users into different groups and obtains the initial position of clusters using Bat Algorithm to generate relevant movie recommendations. In [11–15] several semantically inclined knowledge centric models in support of the proposed literature have been depicted.

3 Proposed System Architecture

To determine the user's personal preferences, insights must be drawn from the user input, which comprises user queries and personal profile information. The user profile contains the user's historical browsing data. It includes previous searches, previous liked videos, ratings given to videos, videos on the watch list, video channels subscribed by the user, etc. This data allows personalized cognizance of user preferences by analysis of past behavior, allowing the model to learn for a better recommendation. User queries are the present search instances or active inputs from the user regarding his/her preferences. Input data is subjected to preprocessing to provide query terms meaningful for prediction by eliminating unnecessary details. Tokenization, Lemmatization, Stop Word Removal, and Named Entity Recognition are some of the data preprocessing methods available in Python's Natural Language processing library NLTK, which can be used for data preprocessing. The output of this step is tokenized keywords belonging to categorical real-world objects (named entities) and free of redundant data. Preprocessing helps in curating the data for further analysis. The resulting query terms though containing keywords are not enough to describe user preferences, this calls for topic modeling and metadata generation on those query terms, as two steps follow this.

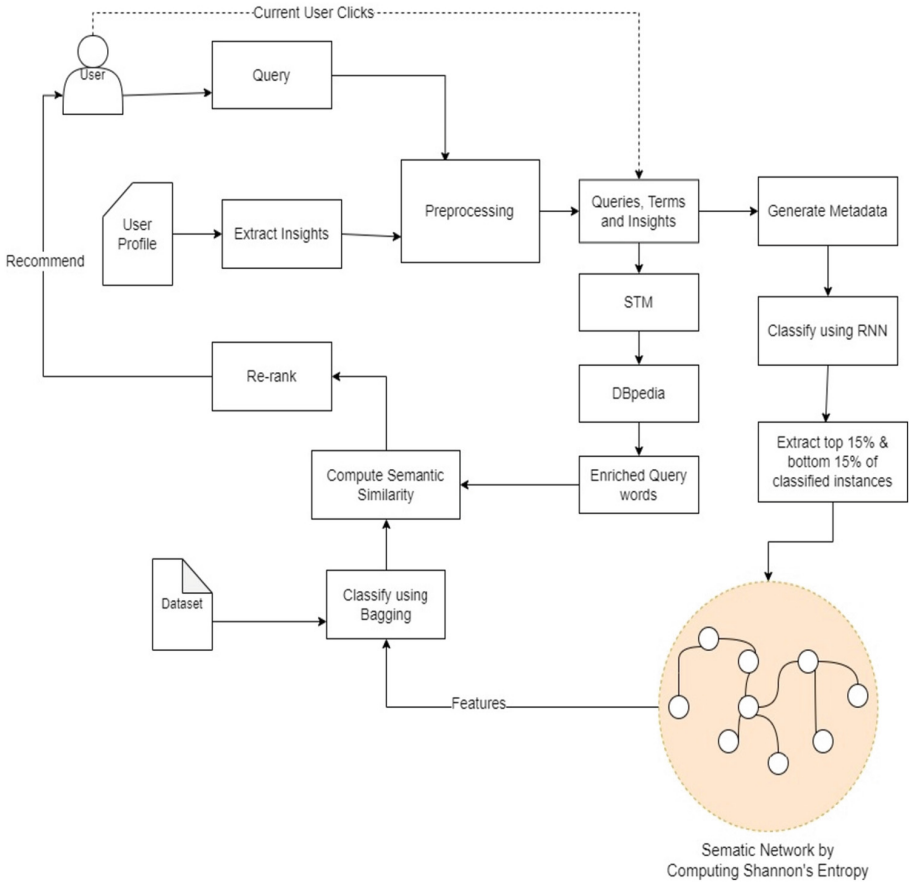


Fig. 1. System Architecture for Proposed SemVidRec

The generated query terms are used to extract metadata by web scraping. A tool is required that analyses data on the semantic web and yields metadata in machine-understandable form. RDF distiller is such a tool. RDF distiller is used to analyze HTML pages annotated by microdata and generate results in RDF specialized formats. This tool examines the content on the HTML5 pages with the help of microdata. This distiller can be used to search the web for annotated pages containing content, matching the query terms, and obtain metadata in the form of an RDF graph formed by multiple labeled RDF triples. Each triple is made of subject, predicate, and object as its components. The RDF graph obtained as the result of this step helps represent metadata in an unambiguous way. The RDF graph is examined to remove the predicate part of RDF triples, obtaining subjects and objects as separate entities. This is necessary because subjects and objects are to be classified in later steps without considering the connections between them describing the object or data property.

The metadata thus obtained is used exponentially large and there is a need to automatically discover classes from it. For feature identification a deep learning model, RNN

is used. Recurrent Neural networks can use internal states to process inputs of varying lengths and make decisions by considering current input along with previous inputs. The RNN discovers classes from among the metadata, by considering the labels associated with the subject and object and classifies them. RNN discovers classes and fits the metadata to the discovered classes. From each discovered class, 15% of the most fitting instances and 15% of the least fitting instances are picked and processed in further steps. This selection scheme is important because, though both categories of instances belong to the same class, there exists a level of diversity in them that can be used to glean insights. The rest 60% of classified instances either are aligned with the top or bottom 15% of the class, contributing least to semantic cognizance. Instances are large in volume and scale, hence it might not be suitable to use 100% of all instances in the recommendation system. The instances are extracted from metadata which is structured information from Web3.0. As the World Wide Web enlarges the metadata grows exponentially. To enhance the speed and efficiency of an algorithm, only a subset of instances is employed. It is possible to choose either the top 30% or the bottom 30% of instances, however this might not offer a diverse representation of the data. Choosing the top 15% and bottom 15% of classified instances ensures proper demarcation and diversity with heterogeneity among the instances selected.

The classified instances are used to create a semantic network. A semantic network is a form of hierarchical ontology map used for knowledge representation. The semantic network can be represented by nodes that encode concepts and edges which represent connections or semantic relations between concepts. The semantic network is cognitively based and the arrangement of nodes is on the basis of a taxonomical hierarchy. For the purpose of feature selection, a semantic network is needed where the selected class instances are organized based on their information measure. The information measure of each instance is realized by computing Shannon's entropy. In the context of instance-based feature selection, Shannon's entropy is used to measure the amount of information contained in each instance with respect to the selected class. Each instance can be thought of as a variable, with its possible values being the class labels. The entropy of an instance is computed by considering the distribution of class labels across the training data that contain that instance. Consider a dataset comprising different instances and their associated class labels. The goal is to identify a subset of instances that are highly informative for a specific class. To achieve this, the entropy value of each instance is calculated with respect to that class using Eq. (1). To compute this entropy value, probabilities that correspond to the frequency of each class label in the set of instances that contain the instance of interest is used.

$$H(X) = - \sum_{i=1}^n p(X_i) \log p(X_i) \quad (1)$$

An agent is modeled using AgentSpeak to create a semantic network based on Shannon's entropy. The state of the model is to calculate Shannon's entropy and the behavior is to create links between discovered class instances to form an information tree. This agent is run intra class to get vertical dependencies, later horizontal dependencies between classes are deduced. The agent is parallel processing hence, links are formed faster and the semantic network is formulated quicker. Topic modeling is a statistical

model aimed at studying the semantic structure of document collections. This unsupervised probabilistic model helps in identifying latent topics and extracting them from unstructured data. Latent Dirichlet Allocation (LDA) is a commonly used type of topic modeling. However, LDA doesn't allow the inclusion of metadata in modeling topics but only considers the text. Hence Structural Topic Modelling is used to include covariates in influencing the topic prevalence and content in query terms. STM is applied to query terms obtained after preprocessing, yielding relevant topics which might be of interest. These uncovered topics are insights drawn from the user data, useful in determining the interests and needs of the user. A python library version of Structural topic modeling will be used for topic discovery. STM will allow topics, relevant to the query terms to be extracted, however, this will not be enough to enrich the query terms. For the purpose of diversifying the query terms, a knowledge store named DBpedia is used. DBpedia dataset has entities classified as ontology, which are structured information from Wikipedia consisting of real-world facts and data, easily accessible on the web. The Virtuoso infrastructure on which the DBpedia dataset is based allows access to DBpedia RDF data through SPARQL endpoints. Using SPARQL endpoints can enrich the query terms with relevant data.

The enriched query terms are labeled categorical dataset which has to be classified based on the features yielded by the semantic network. The features from the formulated semantic network have to be selected to classify the dataset, for which a bagging classifier is used. In bagging, an ensemble of the outputs from the Random forest classifier and Decision trees is used to improve feature selection. Based on the information measure of each feature, indicated by Shannon's entropy features are selected. Within a decision tree, the features are represented by internal nodes, decision rules are represented by branches, and the classification output is stored in the leaf nodes. So here the decision is based on the level occupied by the feature in the semantic network. The random forest chooses the best prediction from among multiple random decisions previously made. Using features/class selection on each bootstrap will produce the best results. The feature which contains the most useful insights is selected first. The classes and their instances from the Semantic network are classified by the bagging classifier.

To produce suggestions, the enriched query terms are compared to the categorized class labels using measures of semantic similarity. These similarity metrics are obtained through the utilization of two functions, specifically the Modified Twitter Semantic Similarity and the Modified Simpson's Diversity Index. The Twitter Semantic Similarity (TSS) is a semantic similarity model that estimates similarity between words with high precision. The Modified Twitter Semantic Similarity is utilized to investigate the semantic structure of text by analyzing the frequency of word co-occurrence in a document corpus.

The frequency of co-occurrence of words w_1 and w_2 in tweets, irrespective of the order is calculated according to Eq. 2. $\Phi(w)$ is a measure of the frequency of a word on Twitter based on the velocity of occurrence. Timestamps of tweets are used to calculate the velocity of occurrence. However, in the case of the Semantic video recommendation model, there is no significance of timestamps, so the Modified Twitter Semantic Similarity measure is used which incorporates the deviational Probability distribution of the word in the web corpus. This is a measure distribution of the word across different web

pages on the video content platform. TSS is similar in performance to corresponding semantic similarity measures like cosine distance of Latent Similarity and the standard measure of WordNet. The result of Modified TSS is a matrix, where each entry represents the TSS measure between a class instance and the query term. The threshold of 0.75 is used to select the class instances similar to enriched query terms. In conventional and traditional practice, a preferred value of 0.75 is commonly used for various semantic similarity measures like Jaccard similarity and cosine similarity. The TSS measure also falls within this range, yielding values between 0 and 1. The specific value of TSS is influenced by factors such as the model’s strength, the number of instances, and the defined epoch set. TSS is computed only once, not iteratively, primarily due to the robustness of the semantic model and the extensive scale of instances involved. Therefore, TSS value of 0.75 is the most appropriate choice for the algorithm

$$TSS(w1, w2) = \left(\frac{\phi(w1, w2)}{\max(\phi(w1), \phi(w2))} \right)^\alpha \quad (2)$$

To quantitatively measure the diversity of recommendations, taking into account richness and divergence among the class instances selected, Simpson’s Diversity Index is used, with certain modifications. The existing index measures the probability of any two randomly chosen identities in a dataset, belonging to the same type. According to Eq. (2) The SDI is the weighted arithmetic mean of propositional abundances. The proportional abundances take values between 0 and 1 hence SDI also ranges between 0 and 1. To choose a diverse set of classified class instances from various sources, certain modifications must be applied to Simpson’s Diversity Index as shown in Eq. (3). Instead of using the square of propositional abundance, one-third of the total product of the square of APMI measure, self-information of enriched query terms, and self-information of specific class instances is used. Step deviation is used for standardizing the abundance values and give more weight to rare species, which can improve the sensitivity of this diversity index. Step deviation of 0.25 ensures diversity and variety in each class.

$$\lambda = \sum_{i=1}^R (s_i^2 * H(x) * H(y)) \quad (3)$$

TSS provides a measure of closeness between a query term and a set of class instances. Simpson’s Diversity Index is used to measure the diversity of recommendations, considering diversity among the class instances selected. The adjustments made to Simpson’s Diversity Index allow for the selection of the most varied group of classified class instances from varied sources. By using a combination of TSS and Simpson’s Diversity Index, the recommendation system can provide a set of class instances that are both relevant and diverse, thus providing a more comprehensive set of recommendations to the user. The process of reordering the recommendations is carried out by utilizing semantic similarity values computed by any measure of semantic similarities, such as the cosine similarity measure, to enhance the accuracy of the recommendations. This step aids in further refining the relevance of the suggestions.

If the user is dissatisfied with the recommendations, the system can use user clicks as feedback for improvisation. The user clicks are used as further input to the model to

improve the relevance and diversity of the recommendations, based on the user preferences. This method allows the algorithm to learn and adapt to the user's changing needs and preferences by generating a new set of recommendations based on the updated feedback. This process can continue until the user is satisfied with the recommendations, indicated by a lack of further clicks. The algorithm can then stop. Incorporating user feedback through clicks can help to improve the effectiveness of recommendation systems and provide a more personalized experience to users. Figure 1 illustrates the proposed System architecture for SemVidRec Video recommendation model.

4 Performance Evaluation and Results

The study utilizes two distinct datasets, the MMTF-14K, a Multifaceted Movie Trailer Dataset for Recommendation and Retrieval [17] and Video Recommendation System Dataset by GlobeIT Solutions, Pune [18]. Both datasets were strategically integrated into a single unit. The datasets were annotated individually using customized focus crawlers with the world wide web as the reference corpora. The annotations were used to categorize the entities in the datasets, and additional categories were added based on these annotations. The entities were then prioritized. These two datasets were combined into a single, larger dataset that was suitable for both movie and video recommendations, including movie trailers. The effectiveness of the SemVidRec model is evaluated by comparing its performance with other established video recommendation models such as VAVR, PVRRC, and CCVR. The results of the comparison are illustrated in Fig. 2. To assess and compare the performance of the system, a fixed set of queries (7156 queries) is used in the experiment. The baseline models and proposed SemVidRec model are implemented in the same environment and compared on the basis of metrics like Precision, Recall, Accuracy, F-measure, False Discovery Rate (FD), and Normalized Discounted Cumulative Gain (NDCG). Precision, Recall, Accuracy, and F-measure are measures of the relevance of predictions made by the model. nDCG quantifies the diversity yielded results. It is evident from Fig. 2, that for Visually-Aware Video recommendation (VAVR) with a cold start, the precision is 92.18%, Recall is 94.08%, Accuracy is 93.13%, F-measure is 93.12%, FDR is 0.08 and nDCG is 0.88. For Personalized Video Recommendation using Rich Contents (PVRRC), the precision is 92.44%, Recall is 95.07%, Accuracy is 93.76%, F-measure is 93.74%, FDR is 0.08 and nDCG is 0.89. For Cross-Curriculum Video Recommendation (CCVR) Algorithm based on a Video-Associated Knowledge Map the precision is 91.03%, Recall is 94.47%, Accuracy is 93.76%, F-measure is 92.72%, FDR is 0.09 and nDCG is 0.95. For SVM+ Fuzzy C-Means Clustering+ Cosine Similarity model the precision is 90.14%, Recall is 92.68%, Accuracy is 91.41%, F-measure is 91.39%, FDR is 0.1 and nDCG is 0.81.

The proposed SemVidRec model yields the highest precision of 95.43%, highest Recall of 97.47%, highest Accuracy of 96.45%, highest F-measure of 96.44%, lowest FDR of 0.05, and nDCG is 0.97. SemVidRec yields the most relevant yet diverse predictions from the dataset and performs better than the aforementioned baseline models. This is because it incorporates both Auxiliary data in the form of Metadata and supplementary knowledge in the form of enriched entities. Structural Topic modeling is used to uncover hidden topics relevant yet not easily recognizable in the queries. Then Dbpedia

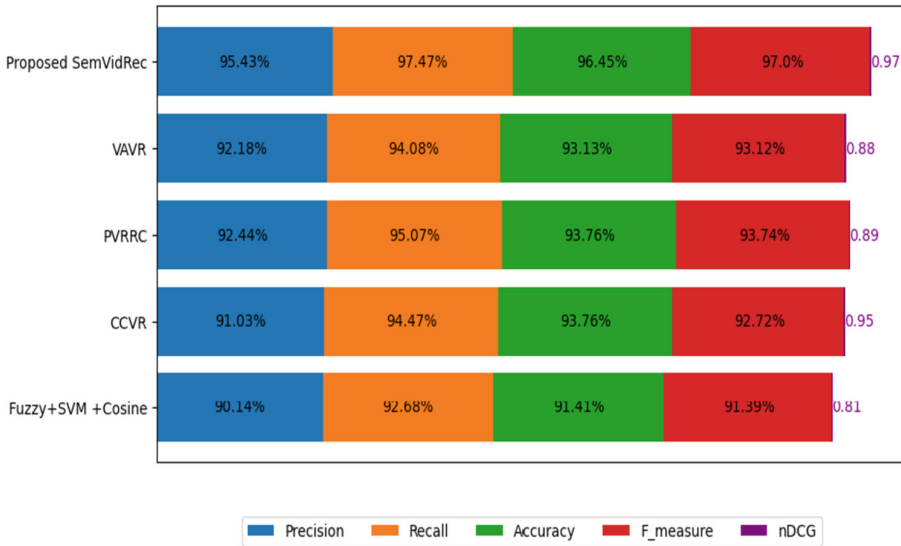


Fig. 2. Performance comparison of Proposed SemVidRec with referential models

knowledge store is used to enrich the query terms hence the nDCG value is very high for the proposed SemVidRec model. The classification of the dataset is achieved by utilizing the features extracted from the semantic network, followed by the implementation of a robust bagging classifier. Bagging ensures a comparative classification model with metadata as features.

The dataset is classified through a bagging classifier, which is an ensemble of decision trees and random forest classifiers. RNN classifies the metadata into categorical and relevant information to be used as features. By computing the Semantic network using Shannon's entropy of the classified metadata, the extraction of highly significant features can be ensured for selection by the bagging classifier. The relevance of prediction made with respect to the user profile and queries is high due to the hybridization of two semantic similarity measures namely Modified Twitter Semantic Similarity and Modified Simpson's diversity index. Both applied with varied threshold and step deviation amount to relevant terms from an ocean of enriched query terms.

VAVR solves the cold start problem, where little annotations are available on a new video, VAVR automatically annotates the videos with visual tags without human intervention. These visual descriptive tags extracted are utilized to generate predictions based on the correlation of visual tags on existing and newly added videos. For extraction of visual features, video is segmented into individual frames based on Color-Histogram distance. As the video is divided into frames, the overall essence of the video is lost and the annotated visual tags are not very relevant in the case of a diverse set of videos. The tags are automatically recognized by machine cognizance but there is a limited amount of information provided to the model. Though the model works in moderate cold start cases, it is highly complex and sometimes yields less relevant predictions based on the

density of visual tags discovered. Hence the model has a moderate nDCG value of 0.88 and an accuracy of 93.13%

PVRRC model ensures the personalization from the content in the videos. The model uses collaborative embedded regression to deal with the unavailability of one specific content by integrating a single content feature into collaborative filtering. The model uses rich content both textual and non-textual for the recommendation of videos in both in-matrix and out-of-matrix scenarios. PVRR uses Collaborative Embedded Regression to deal with the unavailability of single specific features. Priority Late Infusion method PRI is used to combine multiple heterogeneous content features both textual and non-textual. The incorporation of CEF and PRI contributes to improving the accuracy hence the model has above average accuracy of 93.76%. The model only considers titles, description, and reviews for textual features and a combination of normalized color histogram and aural tempos as non-textual features. This model fails to incorporate auxiliary information in the form of metadata and supplementary information in the form of enriched query terms, hence the nDCG value is 0.89, while that of SemVidRec is 0.97. This model depends on the knowledge of videos in the dataset which might not always be available and the relevance computation model is not very strong. However, personalization is an

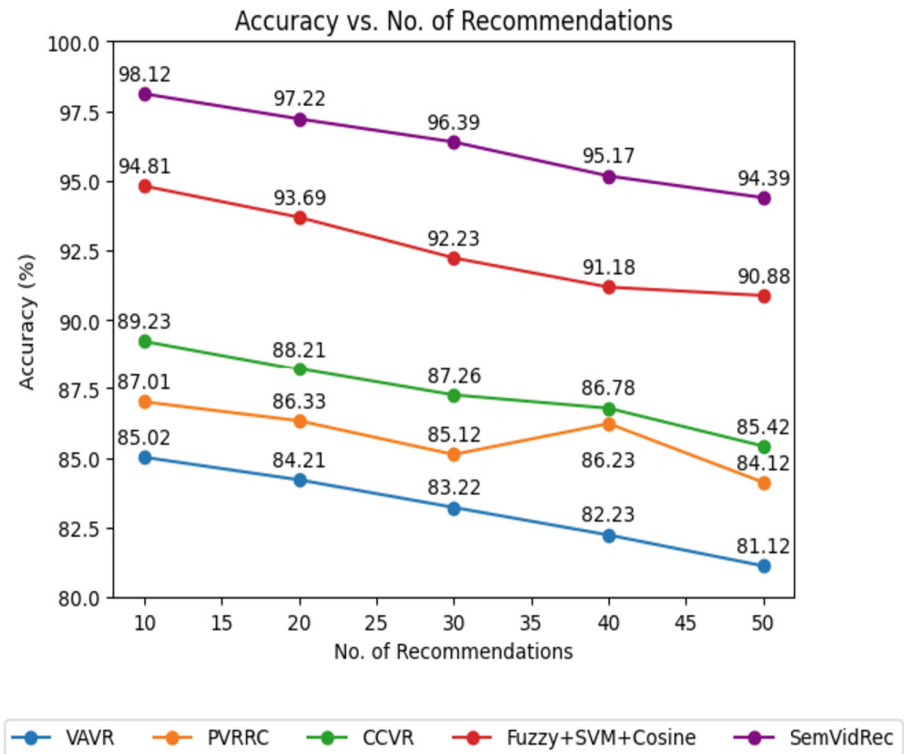


Fig. 3. Accuracy vs Number of Recommendations

added advantage. The relationship between accuracy and a number of recommendations made by SemVidRec and other baseline models is shown in Fig. 3.

The CCVR model aims to help online learners find appropriate learning resources through techniques like the creation of a seed video set, the calculation of correlations between course videos, and the generation of cross-curriculum video subgraphs. By using a video seed set to extract features from student learning platforms and creating a knowledge graph from relevant subgraphs, the CCVR model facilitates efficient resource recommendation for online learning. CCVR model has the highest nDCG value among the baseline models because it includes auxiliary knowledge on existing static knowledge in the form of a knowledge graph. Collaborative filtering is used to filter the generated seed set of videos based on ratings and other feedback. However, not all the videos on the video conglomerate platforms need to be rated. However, due to the inclusion of auxiliary information in the form of a knowledge graph, the model has a high nDCG value is 0.95. The combination of SVM+ Fuzzy C-Means Clustering+ Cosine Similarity model ensures a combination of the binary linear classifier with a strong clustering algorithm and semantic similarity measure. The model lacks auxiliary knowledge and hence has a low nDCG value of 0.81. Since a naive classifier is used and the regulatory mechanism is not strong the model doesn't perform very well. Fuzzy C means works on approximation hence the precision and recall of the model is not very high.

5 Conclusion

The proposed SemVidRec model utilizes semantic similarity to improve the accuracy of video recommendations using user inputs and past activities. The model incorporates metadata obtained from the web through semantic analysis and RDF distillation. This mode also employs sophisticated techniques like the RNN for classification and semantic network-based feature extraction. Additionally, hidden topics are discovered and query terms are enriched by topic modeling. Overall, the proposed model is a more accurate and personalized video recommendation system, which enhances the user experience on video streaming platforms. On a dataset constructed by strategically combining the Video Recommendation System Dataset by GlobeIT Solutions, Pune, and MMTF-14K, a Multifaceted Movie Trailer Dataset for Recommendation and Retrieval, the proposed SemVidRec model showed superior performance when compared to the VAVR, PVRR, and CCVR models. The proposed SemVidRec model outperforms other baseline models with the highest accuracy of 96.45%, precision of 95.43%, recall of 97.47% F-measure of 96.44%, FDR of 0.05, and nDCG of 0.97. This suggests that SemVidRec is a more effective and improved video recommendation system compared to the other models. As an extension of the proposed framework, hybridization with clustering algorithms can be used to reduce the load of classification and hence improve performance. As metadata can be large in volume and scope, only supervised domain-centric knowledge can be utilized in the model. The provision of using domain experts to add best-in-class knowledge will allow improvement with a human-in-the-middle strategy. Most importantly the overall algorithm can be improved by eliminating the learning algorithms and substituting them with theories like gamification.

References

1. Elahi, M., Hosseini, R., Rimaz, M.H., Moghaddam, F.B., Trattner, C.: Visually-aware video recommendation in the cold start. In: Proceedings of the 31st ACM Conference on Hypertext and Social Media, pp. 225–229 (2020)
2. Du, X., Yin, H., Chen, L., Wang, Y., Yang, Y., Zhou, X.: Personalized video recommendation using rich content from videos. *IEEE Trans. Knowl. Data Eng.* **32**(3), 492–505 (2018)
3. Zhu, H., et al.: A cross-curriculum video recommendation algorithm based on a video-associated knowledge map. *IEEE Access* **6**, 57562–57571 (2018)
4. Vimala, S.V., Vivekanandan, K.: A Kullback-Leibler divergence-based fuzzy C-means clustering for enhancing the potential of a movie recommendation system. *SN Appl. Sci.* **1**, 1–11 (2019)
5. Yan, W., Wang, D., Cao, M., Liu, J.: Deep auto-encoder model with convolutional text networks for video recommendation. *IEEE Access* **7**, 40333–40346 (2019)
6. Cai, J.J., Tang, J., Chen, Q.G., Hu, Y., Wang, X., Huang, S.J.: Multi-view active learning for video recommendation. In: *IJCAI*, pp. 2053–2059 (2019)
7. Ma, J., Li, G., Zhong, M., Zhao, X., Zhu, L., Li, X.: LGA: latent genre-aware micro-video recommendation on social media. *Multimedia Tools Appl.* **77**(3), 2991–3008 (2018)
8. Bhatt, C., Cooper, M., Zhao, J.: SeqSense: Video recommendation using topic sequence mining. In: Schoeffmann, K., et al. (eds.) *MMM 2018*. LNCS, vol. 10705, pp. 252–263. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-73600-6_22
9. Mei, T., Yang, B., Hua, X.S., Yang, L., Yang, S.Q., Li, S.: Videoreach: an online video recommendation system. In: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 767–768 (2007)
10. Vellaichamy, V., Kalimuthu, V.: Hybrid collaborative movie recommender system using clustering and bat optimization. *Int. J. Intell. Eng. Syst* **10**(1), 38–47 (2017)
11. Santhanavijayan, A., Naresh Kumar, D., Deepak, Gerard: A semantic-aware strategy for automatic speech recognition incorporating deep learning models. In: Satapathy, S.C., Bhateja, V., Janakiramaiah, B., Chen, Y.-W. (eds.) *Intelligent System Design*. AISC, vol. 1171, pp. 247–254. Springer, Singapore (2021). https://doi.org/10.1007/978-981-15-5400-1_25
12. Praveen, S.V., Ittamalla, R., Deepak, G.: Analyzing the attitude of Indian citizens towards COVID-19 vaccine—a text analytics study. *Diabetes Metab. Syndr.* **15**(2), 595–599 (2021)
13. Deepak, G., Gulzar, Z., Leema, A.A.: An intelligent system for modeling and evaluation of domain ontologies for Crystallography as a prospective domain with a focus on their retrieval. *Comput. Electr. Eng.* **96**, 107604 (2021)
14. Deepak, G., Rooban, S., Santhanavijayan, A.: A knowledge centric hybridized approach for crime classification incorporating deep bi-LSTM neural network. *Multimed Tools Appl* **80**, 28061–28085 (2021)
15. Kumar, A., Deepak, G., Santhanavijayan, A.: HeTOnto: a novel approach for conceptualization, modeling, visualization, and formalization of domain centric ontologies for heat transfer. In: 2020 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT), pp. 1–6. IEEE (2020)
16. Umaa Mageswari, S., Mala, C., Santhanavijayan, A., Deepak, G.: A non-collaborative approach for modeling ontologies for a generic IoT lab architecture. *J. Inf. Optim. Sci.* **41**(2), 395–402 (2020)
17. Deldjoo, Y., Constantin, M.G., Ionescu, B., Schedl, M., Cremonesi, P.: MMTF-14K: a multifaceted movie trailer feature dataset for recommendation and retrieval. In: Proceedings of the 9th ACM Multimedia Systems Conference, pp. 450–455 (2018)
18. GlobeIT Solutions, Pune, India. Video Recommendation System Dataset (2015). <https://www.datahub.io/dataset/video-recommendation-system-dataset>