

# ToViS

*by* Tushaar Gangarapu

---

**Submission date:** 03-May-2018 11:50AM (UTC+0530)

**Submission ID:** 958119906

**File name:** tovis-topic-based.pdf (443.83K)

**Word count:** 4316

**Character count:** 22030

*Project Report On*

# ToViS: Topic Based Video Suggestion System

*Submitted by*

15IT117      Tushaar Gangarapu  
15IT119      Himadri Pal  
15IT130      Pratyush Prakash  
15IT146      Suraj Hegde

VI Semester B.Tech (IT)

*Under the guidance of*

Dr. Sowmya Kamath S  
Department of IT, NITK Surathkal

<sup>1</sup>  
*in partial fulfillment for the award of the degree*

*of*

**Bachelor of Technology**

*in*

**Information Technology**

*at*



<sup>2</sup>  
Department of Information Technology  
NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA, SURATHKAL  
March 2018

# Department of Information Technology

NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA, SURATHKAL

## *Certificate*

This is to certify that this is a bonafide record of the project entitled “**ToViS: Topic Based Video Suggestion System**” presented by the students whose names are given below during VI semester 2018 in partial fulfilment of the requirements of the degree of Bachelor of Technology in Information Technology.

Register No.	Name of Student
15IT117	Tushaar Gangarapu
15IT119	Himadri Pal
15IT130	Pratyush Prakash
15IT146	Suraj Hegde

Date: May 3, 2018

Dr. Sowmya Kamath S  
(Project Guide)

## Department of Information Technology

NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA, SURATHKAL

### *Declaration*

We hereby declare that the Project Work Report entitled “**ToViS: Topic Based Video Suggestion System**” which is being submitted to **National Institute of Technology Karnataka, Surathkal** in accordance with the completion of the minor project for the degree of **Bachelor of Technology in Information Technology** is a bonafide report of the work carried out by us.

Register No.	Name of Student
15IT117	Tushaar Gangarapu
15IT119	Himadri Pal
15IT130	Pratyush Prakash
15IT146	Suraj Hegde

Department of Information Technology

Place: NITK, Surathkal

Date: May 3, 2018

## Acknowledgments

We would like to express our appreciation to all those who have helped us in the successful completion of this project, directly and indirectly.

Firstly, we would like to express our sincere gratitude to our guide *Dr. Sowmya Kamath S*, Department of Information Technology, NITK Surathkal for her insightful advice and guidance throughout the course of our project work. Without her continued support and interest, this project would not have been the same as presented here.

This project would have been impossible without the help of our hostel mates, who took their valuable time to provide inputs to our model development process. Also, we express our thanks to all the teaching and non-teaching staff in the Department of Information Technology for their invaluable support.

May 3, 2018  
National Institute of Technology Karnataka

Tushaar Gangarapu  
Himadri Pal  
Pratyush Prakash  
Suraj Hegde

## Abstract

The exponential growth of web content over the last couple of years has been a promoter and boost for advances in movie recommendation and retrieval. This report focuses on the methodology used for suggesting similar movies to the user based on the view they are currently watching. Collaborative filtering has been the predominant and the current state of the art technique for recommendation of movies. In this report, we show that using a topicality measure for grouping related movies outperforms the collaborative filtering model which uses suffers from the low sparsity and presentation and position bias. Our model also performs better in terms of tackling popularity bias. The topicality based recommendation uses a combination of two approaches to rank the movies. The first approach involves using the applying the traditional inverse document frequency (*idf*) heuristic to the movies. The second approach takes into account user feedback in the form of co view information and various heuristics defined in the methodology are applied. The experiment is conducted on the *IMDb dataset*. In addition, we show that employing user feedback improves user rating and engagement when compared with the collaborative filtering approach.

# Contents

## Acknowledgements

<b>3</b>		
<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Literature Review</b>	<b>4</b>
2.1	Background . . . . .	4
2.2	Identified Gaps . . . . .	5
2.3	Problem Statement . . . . .	6
2.4	Objectives . . . . .	6
<b>3</b>	<b>Methodology</b>	<b>7</b>
3.1	Experimental Setup for Data Collection . . . . .	7
3.2	Tagging the Dataset . . . . .	7
3.3	Topical Analysis . . . . .	7
3.3.1	Topic Count Computation . . . . .	7
3.3.2	Inverse Document Frequency ( <i>idf</i> ) of Topics . . . . .	8
3.3.3	Quality Factor and Stop Words . . . . .	8
3.4	Scoring . . . . .	8
3.5	ToViS: The Video Suggestion Model . . . . .	9
<b>4</b>	<b>Implementation</b>	<b>10</b>
4.1	Work Done . . . . .	10
4.2	Results and Analysis . . . . .	10
4.2.1	Evaluation Methodology . . . . .	10
4.2.2	Evaluation Metrics and Evaluation . . . . .	11
4.3	Innovative Work . . . . .	11
4.4	Details of each individual's work w.r.t. project tasks . . . . .	14
<b>5</b>	<b>Conclusion and Future Work</b>	<b>17</b>
	<b>References</b>	<b>18</b>

## List of Figures

3.1	<i>ToViS</i> Process Model Diagram . . . . .	9
4.1	Watch Time for 20 Users (Based on Experimental Setup) . . . . .	12
4.2	Completion Rate for 20 Users (Based on Experimental Setup) . . . . .	12
4.3	Completion Rate for 20 Users (Based on Experimental Setup) . . . . .	13
4.4	Averaged Evaluation Metrics Used for User Relevance Evaluation . . . . .	13
4.5	User Interface for ToViS (I) . . . . .	14
4.6	User Interface for ToViS (II) . . . . .	14
4.7	User Interface for ToViS (III) . . . . .	15
4.8	User Interface for ToViS (IV) . . . . .	15
4.9	Gantt Chart Representing the Work Flow and Individual Contributions .	16



# List of Tables

2.1	Summary of Existing Works . . . . .	5
4.1	Summary of Likert Evaluation (20 users) . . . . .	11

# Chapter 1

## Introduction

In the past few years, the video content on the Internet increased tremendously. Advancements in social media (Facebook, Twitter, Instagram etc.) have resulted in an abundance of content supply and demand. It has been observed that the number of people willing to spend more time on a website are relatively very less. Now-a-days related video recommendation is *ubiquitous* and *pervasive* on the Internet. From YouTube statistics it can be inferred that over six billion hours of movies have been watched every month. Furthermore, video upload rate is 100 movies per minute. All the above mentioned reasons led to the necessity of development of sophisticated technology for movie retrieval, discovery and recommendations. Movie suggestions are the movies the user is presented with, in relevance to the current movie being watched ( $V_W$ ).

The movies being recommended can be purely content based (tags, genres etc.) or can be collaborative filtering based (used by most of the modern recommender systems). Recommendations via collaborative filtering approach is suitable for movies with good views while the same fails for fresh content and tail videos which have very noisy and sparse co-view data. To overcome sparsity and noise related issues, researchers have combined collaborative filtering with content based models to develop hybrid models. CNN, Netflix, Hulu etc. all use hybrid recommender systems to improve *user's browsing experience* through click-through analytics and co-view data.

Usually movie (video) is modeled as a set of content data which includes a set of *data topics* extracted from many sources such as metadata, Wikipedia articles, uploader keywords (frequently used), freebase entities, common queries used to search, names of play-lists. Then, these assigned (weighted) topics are used to obtain a set of related movies (videos) for recommendations and suggestions. Furthermore, to achieve an *effective topic based* movie representation, weights are assigned to each topic tagged with a movie. Two techniques have been presented to associate weights to topics. The first technique involves a well known retrieval method which uses inverse document frequencies (*idf*)

and topic frequencies ( $tf$ ). The second technique involves using the implicit (explicit) user feedback. Now, this feedback can be utilized to learn the optimal weights of topics (supervised learning).

The topic based representation of the movie being watched ( $V_W$ ) is a *query* which is addressed to the inverted index, the index saves video representations based on topics as *documents*. This way, the movie suggestion problem can be seen as a *retrieval of movies-over-inverted index of documents*. To find the most optimal movie recommendations (even in large volumes of videos), usage of query optimization algorithms can improve efficiency. The top ranked videos (documents) retrieved as a result to the query ( $V_W$ ) are suggested to the viewer as a set of **content based** video recommendations which (recommendations) can either be presented to the user without any modifications or are appended to the results of co-view based movie (video) recommender (hybrid modeling) system.

# Chapter 2

## Literature Review

### 2.1 Background

There has been immense progress in the field of video retrieval due to the increasing user base and user demands on video sharing sites like YouTube, dailymotion, etc. This has led to addressing the various drawbacks and challenges to this video recommendation system.

One of the common challenges includes the position bias which is the case with most retrieval systems. Position bias is the positive bias which is introduced towards the first few retrieved results. Sometimes users even neglect the last couple of videos retrieved by the IR system. This leads to those videos getting a bad feedback recursively and finally the system starts ignoring them. Hence both the system and user relevance is affected.

However it was seen that position bias is not the only bias affecting the system. Yisong Yue et. al [2] introduced a new concept of presentation bias. This sort of bias is predominantly seen in all on-line video sharing platforms. Presentation bias is introduced when the one set of videos have title bolding and the other set doesnt. This prompts the user to select the video from the former set even though their quality is low or they are not related. Title Bolded videos are perceived by the users to be more attractive.

The existing video recommendation systems heavily rely on collaborative filtering as a method to suggest videos. These systems are based on co-view data. Both Davidson et. al. [3] and Baluja et. al. [4] describe such a system. The reason for the success of co view based filtering is because the metadata could be noisy. This is seen in the case when user closes the browser before the system receives the long-watch notification. So, there is inconsistency in the various metrics like abandonment rate. Whereas co-view graphs are nothing but directed graphs depicting a particular user session. Baluja et. al. [4] introduces a concept of an adsorption algorithm. Its a learning and classification framework which uses a rich graph structure and labelled objects.

Collaborative filtering as a method performs well for popular and frequently watched videos but in the case of fresher and tail content videos there is a massive degradation in performance. The reason for this is that the co-view data is noisy and sparse. Hence this filtering algorithm would not be effective if its only based on co-view data.

To tackle this some hybrid approaches have been experimented and implemented. Bo Yang [5] presents a hybrid approach based on relevance feedback and multi-modal fusion. Their model is capable of working without any user preference information. This is very different from the traditional approaches. The multi-modal relevance is depicted by aural, video and textual relevance which is used to recommend videos. However their system was only tested for a small portion of TRECVID dataset having preset sample videos. Hence the system would not be favorable for a large collection of videos.

The approach given below is a hybrid approach tested for a larger audience and a large movie database. Table 2.1 summarizes the existing works.

Table 2.1: Summary of Existing Works

Author	Approach	Merits	Limitations
Davidson J. et al. [3]	Collaborative filtering	Recommendations are more accurate	Presentation bias, only co-viewed videos
Shumeet Baluja et al. [4]	Co-view graph	Recommendations are more accurate	Presentation bias, only co-viewed videos
Bo Yang et al. [5]	Hybrid model	Considers both video info and co-views (multimodal)	Small scale

## 2.2 Identified Gaps

This session briefs about the gaps in the existing recommender systems based on the literature survey. Firstly, the existing recommender (suggestion) systems base their existence on *collaborative filtering* methods, which might work for popular, more viewed, more liked, dense data; while the same fails for sparse data and tail (fresh) videos. It can also be seen that there is a lack of *content based* video retrieval systems. Also, *topic based retrieval* has never been done on a large scale.

## 2.3 Problem Statement

Given a large corpus of movies (here, movie trailers) and appropriate metadata (genre and user submitted tags); propose a methodology to suggest a list of movies the user would like to watch next based on the current movie.

## 2.4 Objectives

The main objective of the recommendation system is to suggest to the user what she would enjoy watching next based on the current movie. This is done by representing user feedback in the form of a co-view based graph. In the representation of this co-view based graph, every movie is represented by a *node*. The current node is linked to any other nodes, if they are often watched in the same session. The nodes in the proximity of the query node and related to the topic is suggested to the user. To find the topical relation, a hybrid dataset composed of the *IMDb movie dataset* comprising of last 35 years of movies and their respective genres along with a set of user defined tags is used.

## Chapter 3

### Methodology

#### 3.1 Experimental Setup for Data Collection

To form the co-view graph, **twenty** people had been chosen (not randomly to not induce bias), and their preferences were noted. The students selected were of varied departments including both circuit and non-circuit branches of different age and linguistic groups with varied tastes and preferences. This was done so as to eliminate any bias that might be introduced as a result of “similar” set of people. They were given a start movie and asked to navigate through to 10 other movies. They however were not asked to watch the movies due to their length, but instead movie *trailers* were used. In this manner, a list of **ten** co-viewed movies was obtained for each movie as the start movie. There were a total of **five** sessions per participant that spanned at different times (not immediately) for over a month. The participants were also asked to tag the watched movies/trailers (user defined tags).

#### 3.2 Tagging the Dataset

To tag the movies, a database of user submitted tags was collected. The movie tags are a combination of the set of genres of a movie and the user submitted tags. These tags were ultimately appended along with the coviews to a common csv file `coview_movie.csv`.

#### 3.3 Topical Analysis

##### 3.3.1 Topic Count Computation

Next, the topic count function was implemented. The topic count function for a particular movie,  $V_W$  and a given topic  $\tau$  returns the number of movies in the co-view graph having

the same topic as the target movie. Here the random unrelated movies which might be seen by the user get a count ( $c(\cdot)$  in equation 3.1) value of **zero** hence they get a score value of zero. Hence, they are omitted.

### 3.3.2 Inverse Document Frequency (*idf*) of Topics

We apply the traditional information retrieval heuristic – inverse document frequency (*idf*) on the topics. This is calculated for all the topics. This is a normalization technique which ensures that rare topics get equal priority as the common ones. This list is stored as a dictionary and the computation is *done only once* so that the prediction-time overheads decrease. This component is attributed by  $\log(1 + df(\cdot))$  in equation 3.1, where  $df(\cdot)$  is the document frequency.

### 3.3.3 Quality Factor and Stop Words

Since the topic terms obtained from the user submitted tags and the genres *do not* contain any stop-words, we need not explicitly normalize the terms during indexing and querying. So, the factor that indicates stop word removal ( $\mathcal{I}_s(\cdot)$ ) is considered to be **one** (equation 3.1). We assume the Quality of the video ( $q(\cdot)$ ) to be **one** (equation 3.1). The quality factor is based on factors like “thumbs up”, “thumbs down”, video popularity etc. But since these factors introduce a popularity bias, these are not added.

$$sc(V_W, V_R) = q(V_R) \sum_{\tau \in V_W \cap V_R} \mathcal{I}_s(\tau) \cdot \frac{c(\tau, V_W)}{\log(1 + df(\tau))} \cdot c(\tau, V_R) \quad (3.1)$$

## 3.4 Scoring

Our final function is the *scoring function* derived from equation 3.1 is defined as in equation 3.2.

$$sc(V_W, V_R) = \sum_{\tau \in V_W \cap V_R} \frac{c(\tau, V_W)}{\log(1 + df(\tau))} \cdot c(\tau, V_R) \quad (3.2)$$

This function computes the similarity score ( $sc(V_W, V_R)$ ) between two videos  $V_W$ ,  $V_R$ . This similarity measure is calculated for all the videos compared with the current video or movie being watched. These similarity scores are then sorted in order and the videos corresponding to top **ten** similarity scores are presented to the user as watch movie recommendations.

We define a *session*, to be the time until which the user chooses one of the videos recommended to the user by the system. The session starts with the user getting a



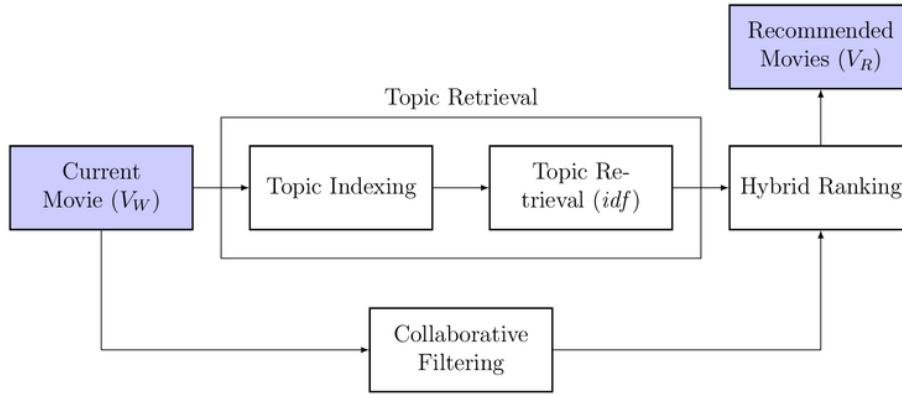


Figure 3.1: *ToViS* Process Model Diagram

random video or movie (during experiment) or the user selecting some movie. Then the above scores are generated and the videos are presented. The user can accordingly select the next movie or choose to end the session.

### 3.5 ToViS: The Video Suggestion Model

**ToViS** is a video suggestion system that basically is a hybrid model. The model mainly focuses on topic based movie retrieval (see section 3.1 for experimental setup) combined with collaborative filtering methods to obtain near optimal recommendations. The block diagram representing the same is depicted as in Figure 3.1. Topic retrieval and Topic indexing are the main areas of focus in this research.

## Chapter 4

# Implementation

### 4.1 Work Done

The system built uses two retrieval techniques, suggestions based on topical and collaborative filtering. The users were given the liberty to select the first movie. Based on the above described methodology the co-view graph was used to calculate the score of all the other movies with respect to the current movie. The top ten movies were ranked and suggested to the user.

Our work incorporated both the user feedback and topic based on inverted index. This allowed our model to be robust. The user feedback also made room for unrelated movie recommendations. This was because the during the process of collecting co views users tend to deviate from the genres and teleport to some random genre. Hence those changes were visible in the co-view graph.

### 4.2 Results and Analysis

The following section describes the evaluation methodology, metrics used and the evaluation outcomes. We have used, user-centered evaluation metrics – watch time, completion rate, abandonment rate.

#### 4.2.1 Evaluation Methodology

There are many ways of evaluating systems like the one presented here (ToViS) – online studies, user studies and historical data based user simulation. We must note that the sample test subjects chosen must represent the population subjects as much as possible. This is necessary to avoid any bias in decision and perception resulting from personal judgments. In addition to selecting the ‘perfect sample’, explicit feedback collection from

the selected users might result in intrusion and irritation.

If a user is to evaluate the given system would result in varied opinions based on a lot of ‘bias’ factors – culture, emotion, location etc. Table 4.1 represents the **Likert Evaluation** of the proposed system with respect to the experimental setup (section 3.1). The following section presents a much more apt evaluation methodology.

Table 4.1: Summary of Likert Evaluation (20 users)

Question <sup>4</sup> Asked	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
Did the recommender system recommend relevant videos?	3	8	6	1	2
Did your session time increase as result of suggestions?	1	3	10	4	2
Did you have Overall Satisfaction?	4	6	7	2	1

#### 4.2.2 Evaluation Metrics and Evaluation

We have used much more apt evaluation metrics that were user centered. **Watch time** is the total time a user spends watching videos (movies) on ToViS. **Completion rate** is the number of videos that were presented to the user by ToViS and were watched completely. *Abandonment rate* is the number of movies (videos) that the user watches up-until the point where the user *does not* continue the session (watch any of the recommendations).

The evaluation of the system with the above mentioned metrics has been carried out in the experimental setup (section 3.1). Watch time (see Figure 4.2.2), Completion rate (see Figure 4.2.2) and Abandonment rate (see Figure 4.2.2) were captured for all 20 users and then these were averaged for overall evaluation (see Figure 4.2.2). Figures 4.5 to 4.8 represent the User Interface for ToViS.

### 4.3 Innovative Work

We added a novel concept of building the co-view graph by approaching a wide range of users and constructing a feedback loop. These metrics formed the basis for our model which was later coupled with the topic based recommendations.

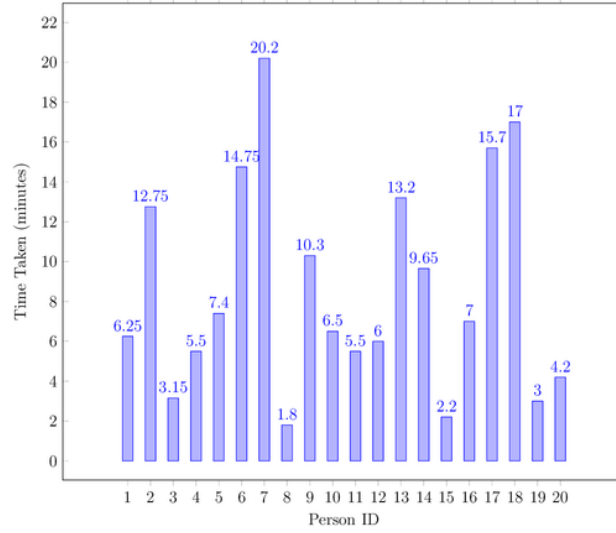


Figure 4.1: Watch Time for 20 Users (Based on Experimental Setup)

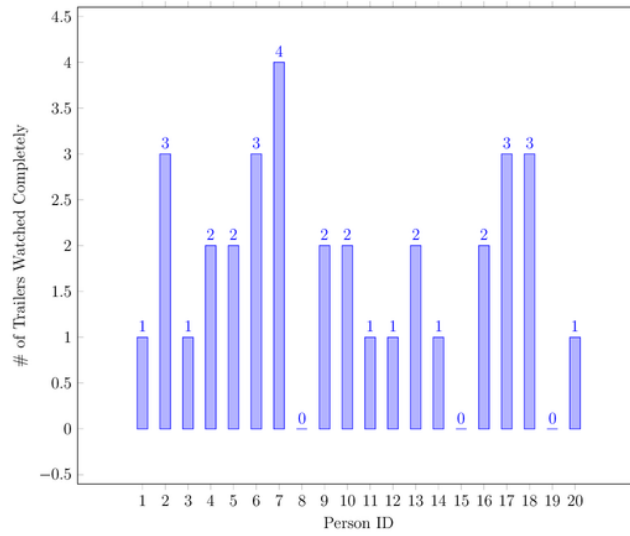


Figure 4.2: Completion Rate for 20 Users (Based on Experimental Setup)

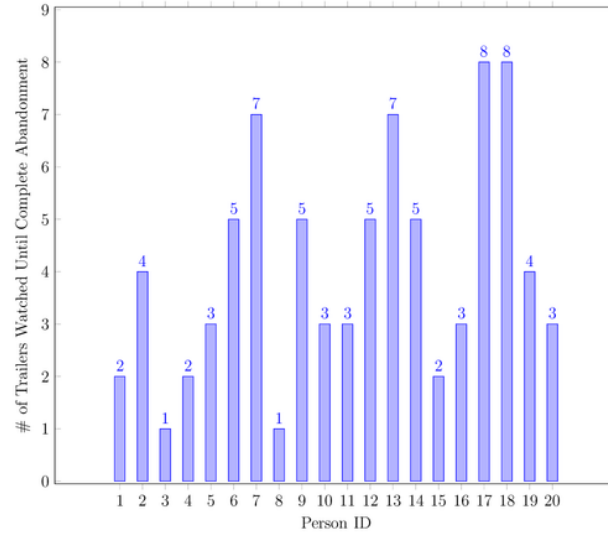


Figure 4.3: Completion Rate for 20 Users (Based on Experimental Setup)

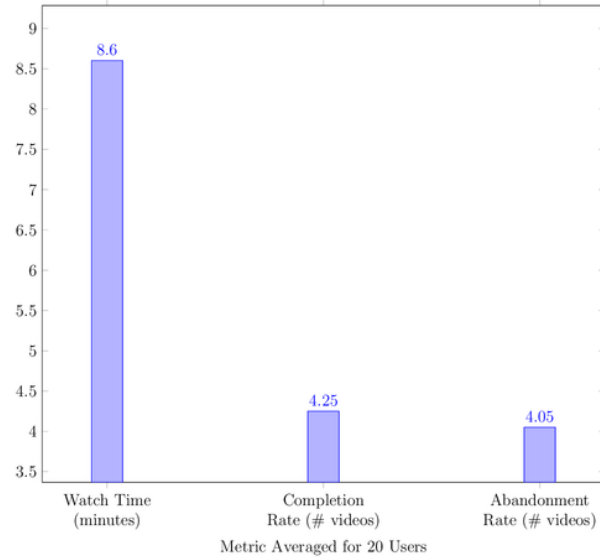


Figure 4.4: Averaged Evaluation Metrics Used for User Relevance Evaluation

## Dunston Checks In (1996)

### Watch Next:

- [It Takes Two \(1995\)](#)
- [Toy Story \(1995\)](#)
- [Dracula: Dead and Loving It \(1995\)](#)
- [Big Green, The \(1995\)](#)
- [Mr. Wrong \(1996\)](#)
- [Don't Be a Menace to South Central While Drinking Your Juice in the Hood \(1996\)](#)
- [Four Rooms \(1995\)](#)
- [Muppet Treasure Island \(1996\)](#)
- [Happy Gilmore \(1996\)](#)
- [Black Sheep \(1996\)](#)

Figure 4.5: User Interface for ToViS (I)

## Ace Ventura: When Nature Calls (1995)

### Watch Next:

- [Dracula: Dead and Loving It \(1995\)](#)
- [Mr. Wrong \(1996\)](#)
- [Don't Be a Menace to South Central While Drinking Your Juice in the Hood \(1996\)](#)
- [Four Rooms \(1995\)](#)
- [Happy Gilmore \(1996\)](#)
- [Black Sheep \(1996\)](#)
- [Friday \(1995\)](#)
- [Bio-Dome \(1996\)](#)
- [Two if by Sea \(1996\)](#)
- [It Takes Two \(1995\)](#)

Figure 4.6: User Interface for ToViS (II)

## 4.4 Details of each individual's work w.r.t. project tasks

The individual contributions and the project time-line has been progressively represented in the form of a Gantt chart as in Figure 4.9.

## Mr. Wrong (1996)

### Watch Next:

- [Dracula: Dead and Loving It \(1995\)](#)
- [Don't Be a Menace to South Central While Drinking Your Juice in the Hood \(1996\)](#)
- [Four Rooms \(1995\)](#)
- [Happy Gilmore \(1996\)](#)
- [Black Sheep \(1996\)](#)
- [Friday \(1995\)](#)
- [Bio-Dome \(1996\)](#)
- [Two if by Sea \(1996\)](#)
- [It Takes Two \(1995\)](#)
- [Ace Ventura: When Nature Calls \(1995\)](#)

Figure 4.7: User Interface for ToViS (III)

## Pocahontas (1995)

### Watch Next:

- [How to Make an American Quilt \(1995\)](#)
- [Things to Do in Denver When You're Dead \(1995\)](#)
- [Persuasion \(1995\)](#)
- [Leaving Las Vegas \(1995\)](#)
- [Bridges of Madison County, The \(1995\)](#)
- [Angels and Insects \(1995\)](#)
- [Carrington \(1995\)](#)
- [Bed of Roses \(1996\)](#)
- [Sense and Sensibility \(1995\)](#)
- [Once Upon a Time... When We Were Colored \(1995\)](#)

Figure 4.8: User Interface for ToViS (IV)

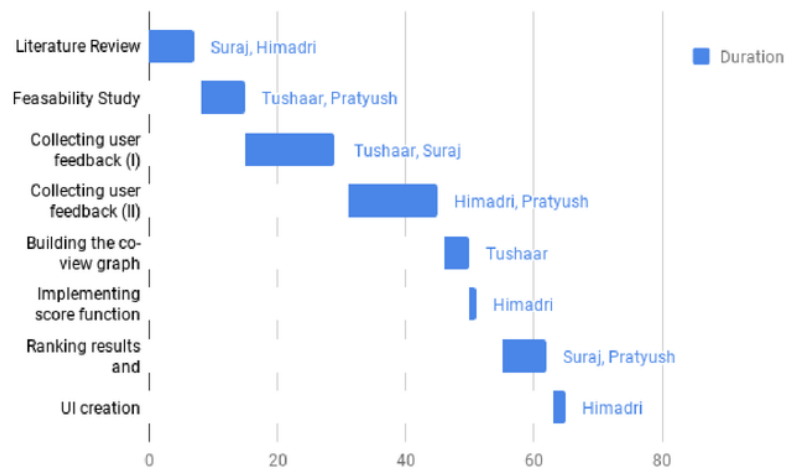


Figure 4.9: Gantt Chart Representing the Work Flow and Individual Contributions



## Chapter 5

### Conclusion and Future Work

In our work, we address the challenges faced by the current most famous model of collaborative filtering. While considering popular and regularly watched videos the user based filtering model gives an accurate recommendation. However for fresh and tail content videos, the collaborative model fails horribly. Our model takes into account the topics of individual videos. Hence this problem is tackled. User engagement increases by using our model of user feedback.

Topicality based retrieval also outperforms collaborative filtering in terms of avoiding the various biases which affect the retrieval system in the form of noise. The hybrid model expressed in the last section would include the best of both worlds.

As the size of the movie database increases it wouldnt be possible to perform computation for the entire database especially in the case of multiple clients. Hence some optimization techniques like WeakAnd needs to be implemented which would consider only a subset of the dataset. This random sample collected from the dataset should accurately describe the dataset. WeakAnd query optimization scores a subset of the entire document so that it crosses a given threshold.

The performance of the model can be further improved by merging it with the collaborative filtering model. However, in this case there would be a need to rerank all the results obtained from both the topic based model and the collaborative model. This reranking function should not be biased towards any one model. Hence the parameters used must either be common for both the models and must have equal weightage in the models or must not overlap with any of the models. The latter one is preferred as it introduces a new metric which might further improve the recall and precision. This hybrid model could outperform both the models.

Dynamic updation of the topic weights could make our model an evolutionary one. Given enough time, our model would keep improving and performing better.

## References

- [1] Bendersky, Michael, et al. (2014). “Up next: retrieval methods for large scale related video suggestion.” Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM.
- [2] Yue, Yisong, Rajan Patel, and Hein Roehrig. (2010). “Beyond position bias: Examining result attractiveness as a source of presentation bias in clickthrough data.” Proceedings of the 19th international conference on World wide web, ACM.
- [3] Davidson, James, et al. (2010). “The YouTube video recommendation system.” Proceedings of the fourth ACM conference on Recommender systems, ACM.
- [4] Baluja, Shumeet, et al. (2008). “Video suggestion and discovery for youtube: taking random walks through the view graph.” Proceedings of the 17th international conference on World Wide Web, ACM.
- [5] Yang, Bo, et al. (2007). “Online video recommendation based on multimodal fusion and relevance feedback.” Proceedings of the 6th ACM international conference on Image and video retrieval, ACM.

## ORIGINALITY REPORT

2%

SIMILARITY INDEX

2%

INTERNET SOURCES

1%

PUBLICATIONS

%

STUDENT PAPERS

## PRIMARY SOURCES

1

[www.scribd.com](http://www.scribd.com)

Internet Source

1%

2

[www.inderscience.com](http://www.inderscience.com)

Internet Source

<1%

3

[www.geo.ed.ac.uk](http://www.geo.ed.ac.uk)

Internet Source

<1%

4

[openresearch-repository.anu.edu.au](http://openresearch-repository.anu.edu.au)

Internet Source

<1%

5

Tao Mei, Bo Yang, Xian-Sheng Hua, Shipeng Li. "Contextual Video Recommendation by Multimodal Relevance and User Feedback", ACM Transactions on Information Systems, 2011

Publication

<1%

6

[zenodo.org](http://zenodo.org)

Internet Source

<1%

Exclude quotes      On

Exclude bibliography      On

Exclude matches      < 10 words