Movie Classification Using k-Means and Hierarchical Clustering

An analysis of clustering algorithms on movie scripts

Dharak Shah
DA-IICT, Gandhinagar
Gujarat, India
dharak_shah@daiict.ac.in

Saheb Motiani
DA-IICT, Gandhinagar
Gujarat, India
motiani_saheb@daiict.ac.in

Vishrut Patel
DA-IICT, Gandhinagar
Gujarat, India
patel_vishrut@daiict.ac.in

***Abstract*— Usually, movies are associated with at least one genre. Assigning a genre to a movie gives the observer an idea about the generic theme of the movie. The trend of associating movies with genres has developed with the continuously increasing number of movies produced over the years and the similarities observed among these movies. It is difficult to exactly determine the factors that qualify a movie to be classified into a specific genre and there are many underlying subtleties required to do so. This study tries to assign movies a corresponding genre based on the movie's script using k-means and hierarchical clustering algorithms.**

Keywords—document clustering; k-means; hierarchihcal clustering; movie genres; movie scripts; IMDB

I. INTRODUCTION

Movies are a significant part of the entertainment industry. Every year, hundreds of movies are produced and released on national or international scale. With such a large number of movies, a huge amount of data is also associated. Some sites, such as the International Movie Database (from here on referred to as IMDB) have organized this data to an extent and have made it visible to the general user.

However, a large chunk of this data has to be curated manually. For example, the genres are assigned to any movie by experienced reviewers and critics. This study tries to determine whether this process can be automated.

Some interesting research already exists which try to classify movies based on various factors. [1] provides the methodology for movie genre classification based on the scene classification of movie trailers. [2] and [3] presents the method to classify movies into Comedies, Action, Drama or Horror based on computable visual cues. [4] categorizes movies into Comedies, Action, Drama or Horror based on the measurable qualities of the musical score of the movie.

Movie scripts provide a lot of critical information about the movie, such as the full dialogue and the scene settings. We try to leverage this information existing in the movie scripts to determine the genre of the movie. We use some of the existing document clustering techniques to determine the cluster in which a particular script should belong. Here, each cluster represents a genre. Movies in the same cluster should belong to the same genre and movies in different clusters should not.

We have used k-means and hierarchical clustering algorithms for clustering the movie scripts, once they have been pre-processed.

II. IMDB AND IMSDB

A. IMDB

This study has used some of the genres identified by IMDB for classification. The genres assigned to

each movie by IMDB have been used to evaluate the results of this study.

IMDB also provides a set of keywords to describe each movie. These keywords aim to provide a slightly better understanding of the theme of the movie. [5] presents an interesting research wherein similar genres have been clustered together based on these keywords. The results of [5] have also been used in this study and are explained in a greater detail in the next section.

B. IMSDB

The International Movie Script Database (IMSDB) has been the source of movie scripts for this study. IMSDB provides a large number of movie scripts across various genres. However, the scripts are often using different formatting and require some amount of pre-processing. The pre-processing is discussed in Section V.

A total of 260 movie scripts were finally selected. These scripts ranged across various genres. The given number is small because of the limitation in the availability of computing resources.

III. DETERMINING MOVIE GENRES TO BE USED FOR CLUSTERING

IMDB recognizes a total of 27 different genres. However, it has been observed by [5] that some of these genres show a high correlation. For example, movies which have been tagged as Mystery are very likely to have the genre Thriller associated with it as well. This was also verified by our clustering methods.

This helps us in reducing the number of clusters. On performing clustering algorithms on closely related genres such as Mystery and Thriller, it has been observed by [1],[2],[3],[4] and [5] that there is a very thin line of difference between two such very closely related genres and currently, no research describes decisive quantitative factors using which the two genres can be distinguished. Of course, Mystery/Thriller combination is just one example.

[1],[2],[3] and [4] have classified the movies into Comedies, Horror, Drama and Action. All the other genres, such as Thriller or Crime, would fall in either one of these categories.

[5] has clubbed the genres into five groups using hierarchical clustering as follows:

Table 1

Cluster	Genres Included
Cluster 1	Short, Drama Comedy, Romance, Family, Music, Fantasy, Sport, Musical
Cluster 2	Thriller, Horror, Action, Crime, Adventure, Sci-Fi, Mystery, Animation, Western
Cluster 3	Documentary, History Biography, War, News
Cluster 4	Reality-TV, Game Show, Talk Show
Cluster 5	Adult

The above classification includes some genres such as Reality-TV, Game Show, Talk Show, Adult, News etc. Our corpus did not contain scripts belonging to these genres. Hence, based on the linkage matrices provided in [5], we created a classification more suitable for our study.

Table 2

Cluster	Genres Included
Cluster 1	Drama, Comedy, Romance, Family, Sport, Musical
Cluster 2	Action, Western, War
Cluster 3	Sci-Fi, Adventure, Fantasy, Animation
Cluster 4	Crime, Mystery, Thriller
Cluster 5	Horror

The above classification was derived by grouping together movie genres which show high similarity in terms of co-occurrence and repetition of movie keywords as defined by IMDB. For more details regarding how the keywords were used to determine the group of genres, please refer [5].

IV. Determining the dominant movie genre for evaluation

Generally, a large number of movies belong to more than one cluster. For example, *Star Wars: A New Hope* belongs to Sci-Fi as well as Adventure. Since, in this study, we are focusing on hard clustering algorithms such as k-means and hierarchical clustering, it becomes necessary, for the sake of simplicity in evaluation, to assign exactly one genre to each movie.

For evaluation, each movie was finally assigned a single genre based on the keywords assigned to it by IMDB. If the majority of keywords for a movie belong to the genre G1, then the dominant genre assigned to the movie would be G1. [5] explains how the keywords corresponding to a genre can be extracted.

If the genre G1 belongs to cluster C1 as shown in *Table 2*, then the cluster for the said movie would be C1.

The clustering algorithms classified each movie into genres G1 through G17. And each genre was further classified into clusters C1 through C5. Hence, finally each movie belonged to one of the five clusters. These results were tested against each movie's dominant genre as observed on IMDB.

V. Pre-processing movie scripts

Firstly we removed extra escape sequences and markup tags from the movie scripts. After doing so we used NLTK for further steps.

NLTK is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing and semantic reasoning.

We performed tokenization using NLTK tokenizer package. We used wordpuct_tokenize method on the sentences of scripts.

After tokenizing we used Porter stemmer for stemming using NLTK stemmer package. The Porter stemming algorithm (or 'Porter stemmer') is a process for removing the commoner morphological and inflexional endings from words in English. Its main use is as part of a term normalization process that is usually done when setting up Information Retrieval systems.

We then removed stop words using WordNet standard stop word list. Once we are done with above steps we create TF-IDF matrix for further processing and results. TF-IDF, term frequency–inverse document frequency, is a numerical statistic which reflects how important a word is to a document in a collection or corpus. We apply k-means on this TF-IDF matrix, but we process it further to apply hierarchical clustering.

VI. Clustering using k-means

In Information Retrieval, k-means clustering is a method of cluster analysis which aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean.

Once the TF-IDF matrix has been created, we use k-means to classify each of the vectors into one of the k clusters.

Actually, k-means assumes that all of its instances are real valued vectors. Here, the instances are the movie scripts' vectors from the TF-IDF matrix. Each row represents a movie script and each column denotes a dimension along which the script's magnitude is written.

In k-means the clusters are based on the centroids, center of gravity, or mean of points in a cluster, c:

$$\vec{\mu}(c) = \frac{1}{|c|} \sum_{\vec{x} \in c} \vec{x}$$

The reassignment of instances to clusters is based on distance to the current cluster centroids.

There are multiple distance metrics which can be used while clustering using k-means.

All the k-means algorithms have been made to run for 500 turns.

We used Euclidean distance as well as the cosine distance of the vectors as the metric for clustering.

The Euclidean distance metric defines the distance between the two centroids of two clusters as follows:

$$L\ (\vec{x}, \vec{y}) = \sqrt{\sum_{i=1}^{m} (x_i - y_i)^2}$$

The results obtained using k-means and the Euclidean distance metric are as follows:

Table 3
(The parentheses indicate genre names, refer Table 2)

Cluster	Precision	Recall
Cluster 1 (DCRFSM)	0.69	0.43
Cluster 2 (AWW)	0.31	0.36
Cluster 3 (SAFA)	1.00	0.20
Cluster 4 (CMT)	0.88	0.20
Cluster 5 (H)	0.16	0.64
Average	0.61	0.37
F1 Measure	0.45	

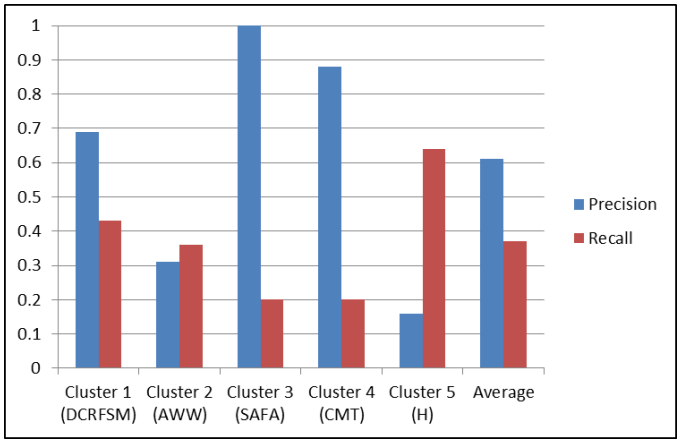

Figure 1: k-means with Euclidean metric

The cosine distance metric defines the distance between the two centroids of two clusters as follows:

$$L\ (\vec{x}, \vec{y}) = 1 - \frac{\vec{x} \bullet \vec{y}}{|\vec{x}| \cdot |\vec{y}|}$$

This is the cosine of the angle between the two vectors.

The results obtained using k-means and the cosine distance metric are as follows:

Table 4
(The parentheses indicate genre names, refer Table 2)

Cluster	Precision	Recall
Cluster 1 (DCRFSM)	0.62	0.52
Cluster 2 (AWW)	0.41	0.40
Cluster 3 (SAFA)	0.57	0.43
Cluster 4 (CMT)	0.36	0.47
Cluster 5 (H)	0.20	0.37
Average	0.43	0.44
F1 Measure	0.44	

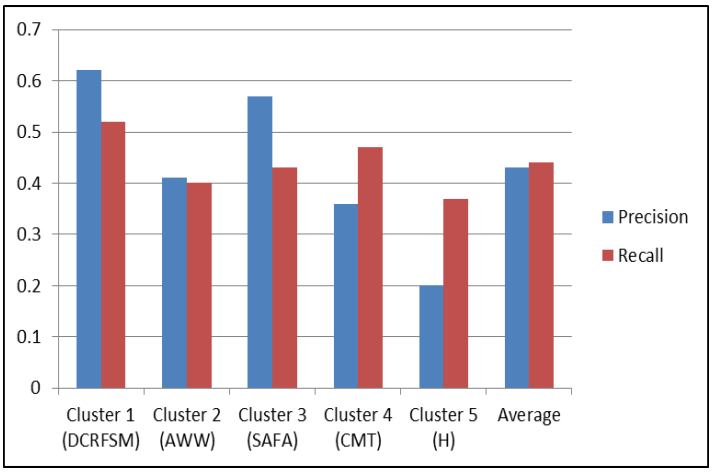

Figure 2: k-means with cosine distance metric

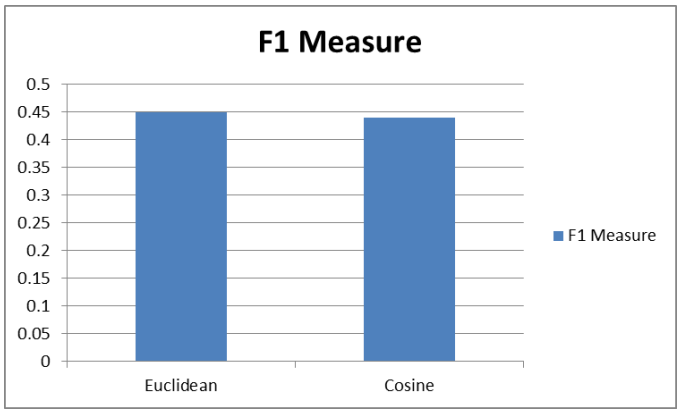

Figure 3: Comparison of F1 Measures for different metrics on k-means

VII. HIERARCHICAL CLUSTERING

Hierarchical clustering is a method of cluster analysis which seeks to build a hierarchy of clusters. Strategies for hierarchical clustering generally fall into two types:

Agglomerative: This is a "bottom up" approach: each observation starts in its own cluster, and pairs of clusters are merged as one moves up the hierarchy.

Divisive: This is a "top down" approach: all observations start in one cluster, and splits are performed recursively as one moves down the hierarchy.

In order to decide which clusters should be combined (for agglomerative), or where a cluster should be split (for divisive), a measure of dissimilarity between sets of observations is required. In most methods of hierarchical clustering, this is achieved by use of an appropriate metric (a measure of distance between pairs of observations), and a linkage criterion which specifies the dissimilarity of sets as a function of the pairwise distances of observations in the sets.

The various linkage criteria are as follows:

- Single/Nearest Point Algorithm
- Complete/Farthest Point Algorithm
- Average/UPGMA
- Centroid/UPGMC
- Other methods include WPGMA (uses Centroid),WPGMC (Average of centroids gives the new centroid) and
- Ward algorithm.

In this study we have used some of the linkage criteria used above. We have used the Nearest Point Algorithm as well as the Ward algorithm as the linkage criteria.

For all points *i* in cluster *u* and *j* in cluster *v*, the Nearest Point Algorithm assigns the distance between the clusters *u* and *v* as:

$$d(u, v) = \min(dist(u[i], v[j]))$$

The results for Hierarchical clustering using the Nearest Point Algorithm are as follows:

Table 5
(The parentheses indicate genre names, refer Table 2)

Cluster	Precision	Recall
Cluster 1 (DCRFSM)	0.35	0.27
Cluster 2 (AWW)	0.22	0.36
Cluster 3 (SAFA)	0.30	0.20
Cluster 4 (CMT)	0.34	0.44
Cluster 5 (H)	0.32	0.48
Average	0.30	0.35
F1 Measure	0.32	

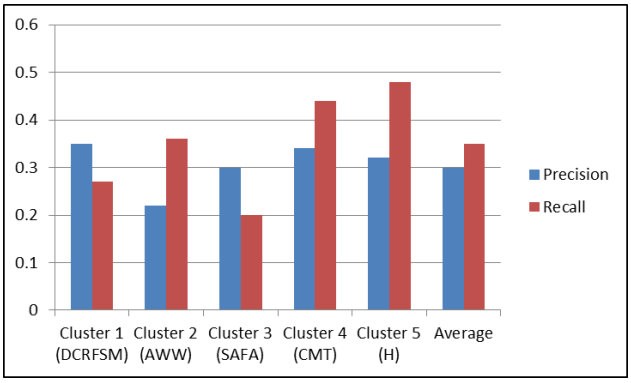

Figure 4: Hierarchical clustering using the NPA linkage criteria

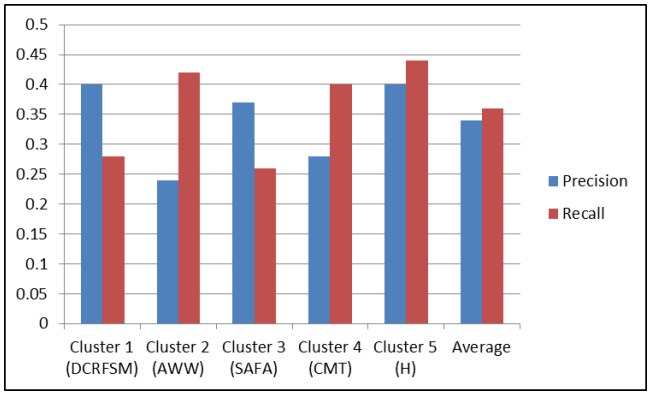

Figure 5: Hierarchical clustering using the Ward linkage criteria

For the Ward variance minimization algorithm, the new entry d(u,v) is computed as follows,

$$d(u,v) = \sqrt{\frac{|v|+|s|}{T}d(v,s)^2 + \frac{|v|+|t|}{T}d(v,t)^2 + \frac{|v|}{T}d(s,t)^2}$$

The results for Hierarchical clustering using the Ward Algorithm are as follows:

Table 6

(The parentheses indicate genre names, refer Table 2)

Cluster	Precision	Recall
Cluster 1 (DCRFSM)	0.40	0.28
Cluster 2 (AWW)	0.24	0.42
Cluster 3 (SAFA)	0.37	0.26
Cluster 4 (CMT)	0.28	0.40
Cluster 5 (H)	0.40	0.44
Average	0.34	0.36
F1 Measure	0.35	

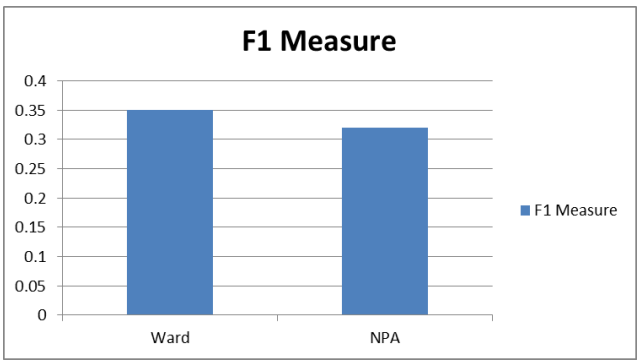

Figure 6: Comparison of F1 Measures for different linkage criteria on hierarchical clustering.

VIII. CONCLUSION

It was observed for this study that the overall results using hard clustering methods aren't very satisfactory. One of the reasons for this might be that assigning each movie just the dominant genre isn't a very good idea. A movie, when assigned a dominant genre and stripped of its other genres loses very important part of its identity.

REFERENCES

[1] H. Zhou, T. Hermans, A. V. Karandikar, J. M. Rehg, "Movie Genre Classification via Scene Categorization", Proc. 10th international conference on Multimedia, pp. 747-750, 2010.

[2] Z. Rasheed and M. Shah, "Movie genre classification by exploiting audio-visual features of previews", Proc. the 16th International Conference on Pattern Recognition vol.2, no., pp. 1086- 1089 vol.2, 2002.

[3] Z. Rasheed, Y. Sheikh, and M. Shah, "On the use of computable features for film classification," IEEE

Transactions on Circuits and Systems for Video Technology, vol.15, no.1, pp. 52- 64, Jan. 2005.

[4] A. Austin, E. Moore, U. Gupta, and P. Chordia, "Characterization of movie genre based on music score," IEEE International Conference on Acoustics Speech and Signal Processing, pp.421-424, 2010

[5] H. Bulut and S. Korukoglu, "Analysis and Clustering of Movie Genres", Journal Of Computing, Volume 3, Issue 10, pp.16-23, October 2011