Textual Analysis of Movies Dataset in SAS Enterprise Miner

By

Jigar Mehta, Taj Pirzada, Preethi Bojja, Myanka Batra

Movies dataset contained synopsis of reviews on each movie along with movie genre dataset that classified movie according to genres.Textual analysis was performed on the synopsis of the reviews to find few search results and to verify the genre classifiction to the topic modelling using SAS Enterpise Miner.

❖ Finding a Text String in a Dataset

The first part of the exercise was to do preliminary searching of text based on movie knowledge and to find out the results. Movie Data dataset was considered for this purpose and the following questions were answered.

a. Identify the name of an actor or actress of interest.

The actor that we identified for this question is "Hugh Jackman". In the following parts we will use the name in a few queries.

b. Find all of the movies in the data set that have a synopsis that mentions the selected name.

We start by importing the **MOVIEDATA** to SAS and create a file import node. Further, to setup for filtering through the text we add the text parsing node which decomposes textual data and generates a quantitative representation suitable for data mining purposes. Lastly, we add the text filter node which transforms the quantitative representation into a compact format suitable for filtering easily.

After this we run our diagram and once we receive results we filter the data via "Filter View", wherein we can easily search for words. In this case we search the name selected in part a. i.e. "Hugh Jackman"

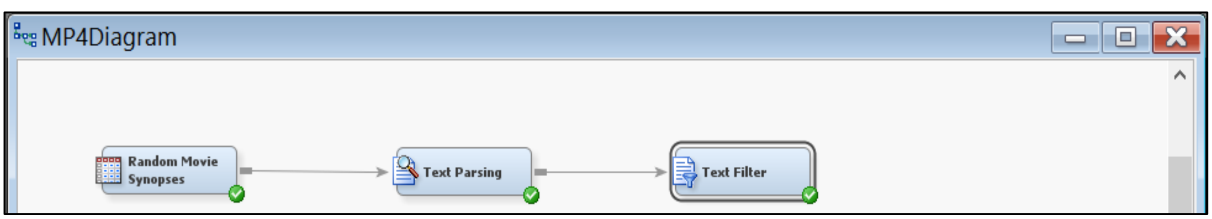
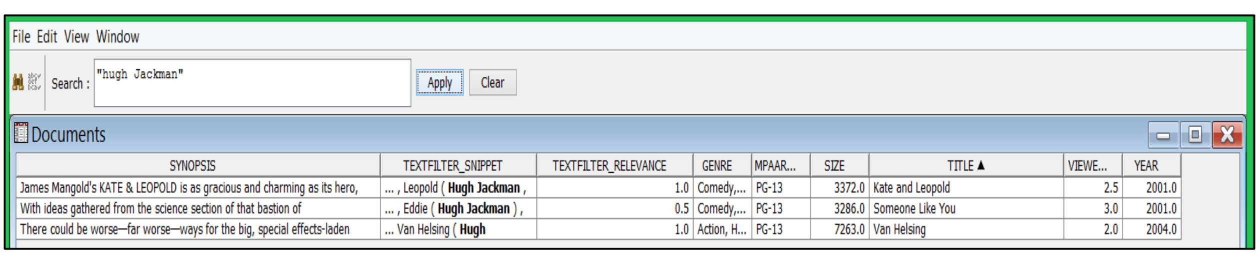


Diagram of Movie Data text filtering

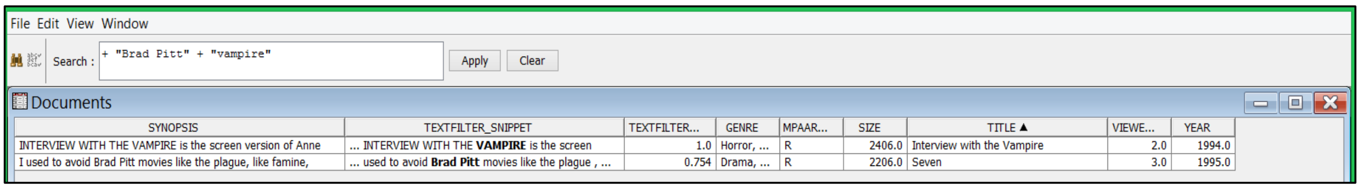


SYNOPSIS	TEXTFILTER_SNIPPET	TEXTFILTER_RELEVANCE	GENRE	MPAAR...	SIZE	TITLE ▲	VIEWE...	YEAR
James Mangold's KATE & LEOPOLD is as gracious and charming as its hero,	... , Leopold (**Hugh Jackman** ,	1.0	Comedy,...	PG-13	3372.0	Kate and Leopold	2.5	2001.0
With ideas gathered from the science section of that bastion of	... , Eddie (**Hugh Jackman**) ,	0.5	Comedy,...	PG-13	3286.0	Someone Like You	3.0	2001.0
There could be worse—far worse—ways for the big, special effects-laden	... Van Helsing (**Hugh**	1.0	Action, H...	PG-13	7263.0	Van Helsing	2.0	2004.0

Search results of Hugh Jackman

c. Has Brad Pitt ever portrayed a vampire in a movie?

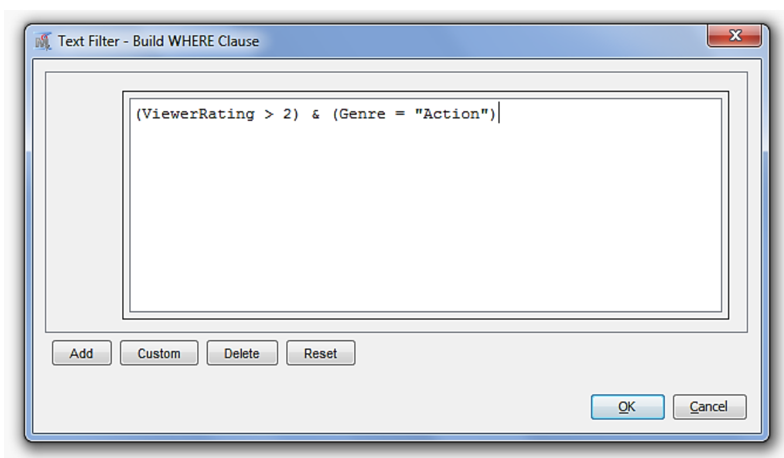
As we already set our base in the previous part of the question, in this case we simply search in the filter view for the text “Bard Pitt” and “Vampire”. Also it can be noticed that we have added a “+” sign before the text “Brad Pitt”, it is to give equal weightage to both. For example, if we just put + between the two strings the latter string gets more weightage and the results we get have the text “vampire” in them but may or may not have the word “Brad Pitt”. Hence putting equal weightage to both words gives us the result where both the strings appear.



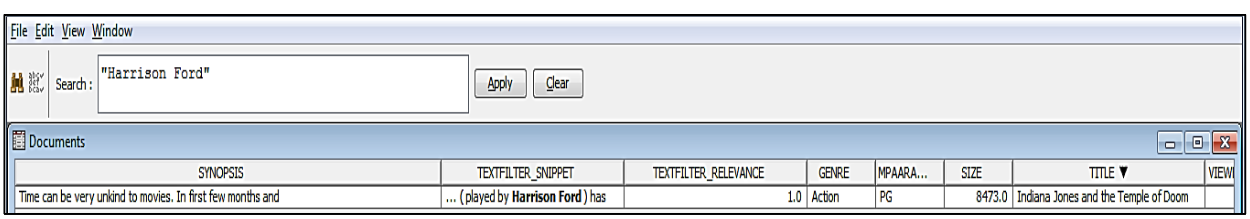
Brad Pitt and Vampire search result

d. Formulate a complex question that this dataset can answer. Show your work to answer this question.

The query that we formulated is a case where we search for all the action movies in which Harrison Ford acted and which have ranking greater than 2. To implement this query along with the steps that we followed in the earlier part we add the “where” clause to check for the rankings and genre. For adding the clause, we use the “subset document” field in the properties of the text filter. Herein we can add multiple where conditions as shown in the screen shot.



Thereafter we can simply search for the actor, as the data is filtered according to the where clause conditions we get the answer to our query.



❖ Text Analysis on Genres and Topic Modelling

Movies dataset contained synopsis of reviews on each movie and the movies were assigned to maximum of 5 genres based on relevancy of movie. Textual analysis is performed on the synopsis of the reviews to verify the genre classifiction to the topic modelling using SAS Enterpise Miner. The results obtained using various models and the inferences along with variation in parametrs in the models are documented as below:

Q2.B)

Results of running topic model on only Synopsis:

After we got the results of the topic model (10 topics) using default settings as mentioned, we analyzed the top 10-15 keywords for each topic and tagged the topics as following genres (highlighted in blue):

Topic Id	Document Cut off	Term Cutoff	Topic	# Terms	# Docs	Our Interpretation
1	0.142	0.013	+show,+movie,+rate,+recommend,acting	862	179	Suspense
2	0.133	0.013	hollywood,+protagonist,+play,+film,+plot	970	115	Drama
3	0.106	0.013	+motion,+viewer,+picture,+moment,+character	1232	175	Romance
4	0.086	0.013	+man,+woman,+american,+people,+country	1251	142	Documentary
5	0.078	0.013	+comedy,+funny,+joke,humor,+laugh	1201	181	Comedy
6	0.058	0.013	+action,+effect,+war,earth,science	1136	129	SciFi
7	0.062	0.013	+cop,+crime,+thriller,police,+action	1146	168	Mystery
8	0.077	0.013	granger,gauge,movie,+mother,+child	1060	177	SciFi
9	0.081	0.013	best,+oscar,+win,actor,picture	870	78	Drama
10	0.096	0.013	+bond,bond,connery,james,jeffrey	863	92	Action

The 10 genres given to us was **ACTION, COMEDY, DOCUMENTARY, DRAMA, HORROR, KIDSFAMILY, MYSTERY, ROMANCE, SCIFI,** and **SUSPENSE.**

After comparing the 10 topics generated by the topic model to the above 10 topics selected for analysis, we have following inferences:

1. Tagging Kids and Family into 1 genre does make sense as top words under this topic is a combination of kids, student and family members.
2. In future, Suspense and Mystery can be combined into 1 genre (1 topic) as the words in both the topics are very similar.
3. There is no possibility of Horror in the 10 topics what SAS generates.
4. Making more than 10 topics is useful as SAS does a better job in generating more distinct and meaningful topics.

Q2.C) Devise and implement a plan to compare the performance of the classifier when manipulating the following parameters:

a. Frequency Weighting
b. Term Weight
c. Minimum number of documents
d. Number of topics

Process Flow Diagram:

We varied number of parameters in the text filter and text topic nodes and compared 5 different text topic models to see which classifier performs better in terms of root mean square error.

Text Topic Model	Frequency Weight	Term Weight	Min # docs	Min # topics
1	Binary	IDF	4	10
2	Log	Entropy	4	10
3	Log	IDF	30	10
4	Log	IDF	4	10
5	Log	IDF	4	25

Model selection based on various parameters

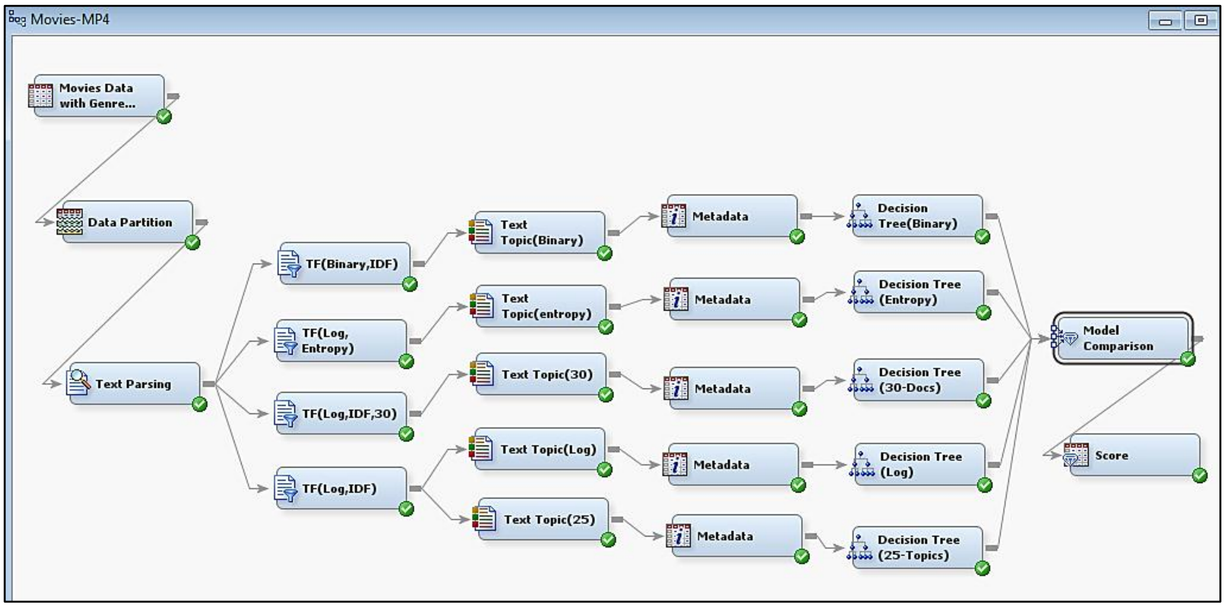


Diagram of the complete text analysis

1. **The file import node** creates a single SAS data set from your document collection. The SAS data set is used as input for the Text Parsing node, and contains the actual text. We use only the synopsis field for developing a text topic model.

2. **The Data partition node** is used to split the data into training and validation. Here, we have used 75:25. The training data is used for model fitting. The validation data is used to assess the accuracy of the classifier in the Model Comparison node. The validation data set is also used for model fine-tuning in the Decision Tree model node to create the best subtree.

3. **The text parsing node** decomposes textual data and generates a quantitative representation suitable for data mining purposes. It enables you to parse a document collection in order to

quantify information about the terms that are contained therein. We used movie start list as the start list. And SASHELP.ENGSTOP as the stop list. These are list of many terms that are excluded from further computations. The selections indicated in the Parts of Speech option ensure that the analysis ignores low-content words such as prepositions and determiners. We have also changed the default option of ‘Check Spelling’ from “No” to “Yes to check and correct the spelling of terms in the input data set.

4. **The text filter node** is used for transformation (dimension reduction). It transforms the quantitative representation into a compact and informative format. It can be used to reduce the total number of parsed terms or documents that are analyzed. Therefore, you can eliminate extraneous information so that only the most valuable and relevant information is considered. Here, we tried experimenting various parameters like term weight, frequency weight and minimum number of documents to create different topic models.

   The term frequency and inverse document frequency is used to produce a composite weight for each term in each document. The *tf-idf* weighting scheme assigns to term t a weight in document d given by:

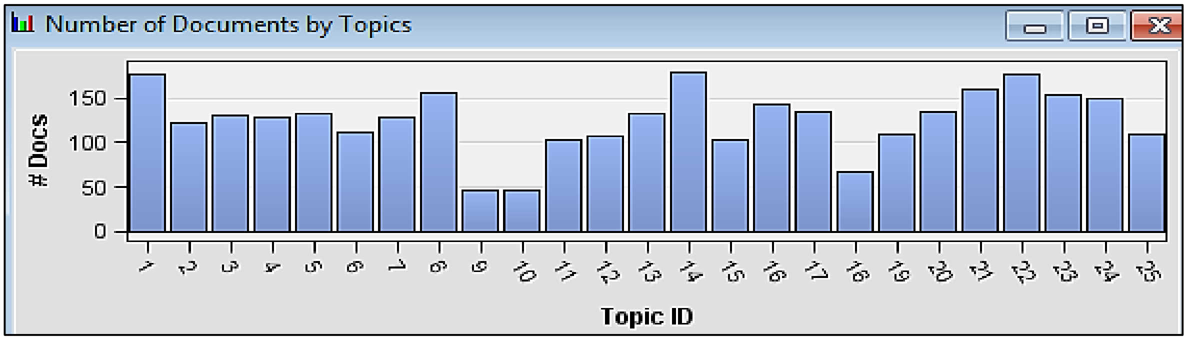
$$\text{tf-idf}_{t,d} = \text{tf}_{t,d} \times \text{idf}_t.$$

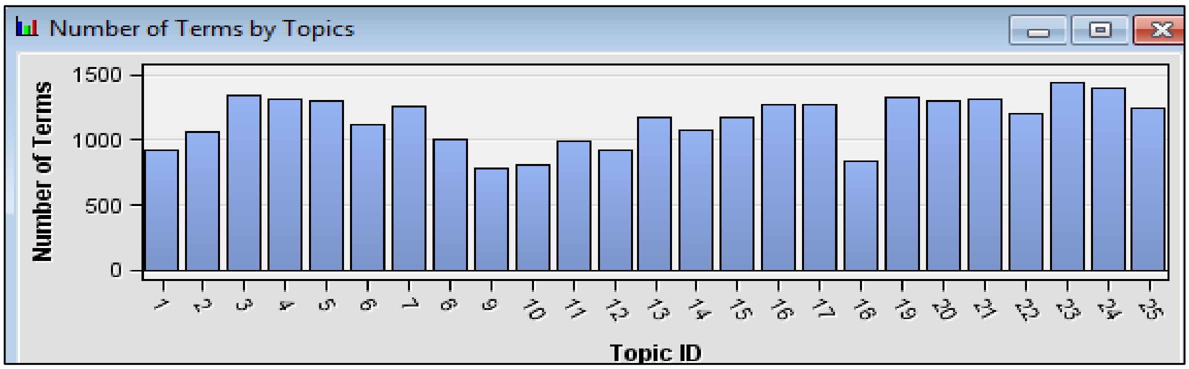
   In other words, tf-idf $_{t,d}$ assigns to term t a weight in document d that is:

   - highest when occurs many times within a small number of documents (thus lending high discriminating power to those documents);
   - lower when the term occurs fewer times in a document, or occurs in many documents (thus offering a less pronounced relevance signal);
   - lowest when the term occurs in virtually all documents.

5. The Text Topic node performs cluster analysis to group the documents and summarizes the collection by identifying “topics”. The node uses singular-value-decomposition (SVD) in the background to capture information from a sparse term-by-document matrix. The node can be configured to identify single-term topics or multi-term topics in the data. The properties of the node has been decided carefully based on the size of the document collection.

Output of Text Model:





Terms by topics

Topic ID	Document Cutoff	Term Cutoff	Topic	Number of Terms	# Docs
1	0.129	0.013	+show,+recommend,acting,+movie,nudity	915	176
2	0.112	0.013	hollywood,+protagonist,+order,+play,+cinema	1065	122
3	0.084	0.013	+motion,+picture,+viewer,+love,+adaptation	1335	131
4	0.075	0.013	+woman,+man,+people,+american,york	1315	128
5	0.061	0.013	+comedy,+joke,humor,+funny,+laugh	1301	132
6	0.073	0.013	+war,+soldier,+battle,war,war	1118	112
7	0.056	0.013	+president,+thriller,+murder,political,+government	1258	128
8	0.064	0.013	granger,gauge,movie,+revolve,+writer	1006	155
9	0.079	0.012	best,actor,+win,picture,+nominate	782	45
10	0.080	0.013	+bond,bond,connery,spectre,james	801	46
11	0.075	0.013	science,fiction,cameron,+alien,earth	990	103
12	0.082	0.013	+kid,+age,jeffrey,+dog,+voice	923	106
13	0.067	0.013	+school,+girl,+student,+high,+high school	1175	132
14	0.079	0.013	acceptable,+language,+rate,+teenager,sexual	1075	180
15	0.067	0.013	chan,+action,martial,+stunt,jackie	1172	102
16	0.066	0.013	+crime,+heist,+criminal,+cop,joe	1274	144
17	0.068	0.013	horror,+horror,+thriller,+killer,+victim	1263	134
18	0.069	0.013	harry,+harry,dirty,dvd,san	829	67
19	0.047	0.013	+town,western,costner,texas,west	1331	110
20	0.053	0.013	+song,+musical,music,+sing,+singer	1297	135
21	0.062	0.013	+child,+family,+mother,family,+home	1314	160
22	0.070	0.013	+woman,+love,+relationship,+romance,sexual	1199	177
23	0.055	0.013	+character,roberts,+help,+play,julia	1440	153
24	0.060	0.013	dr,sandler,+comedy,adam,+doctor	1391	149
25	0.062	0.013	+team,+coach,football,+player,+sport	1247	110

25 topics modelling from synopsis data

Topics extracted from the topic node are groups of terms that define a compact representation of the document collection. For example, Topic ID 5 shows as “+comedy, humor,

+jokes, laugh, hilarious" which seems very relevant for this analysis as we identify as this genre as Comedy. We can identify the important grouped terms to analyze the movie synopsis using these text topics.

Text Topic node also assigns a score for each document and term cutoff for each topic. Then, these thresholds are used to determine if the association is strong enough to consider that a document or a term belongs to the topic. In this analysis we used only multi term topics.

In the Topics window, there is a column labeled **Term-Cutoff**. For each created topic, the algorithm computes a topic weight for every term in the corpus. This measures how strongly the term represents or is associated with the given topic. Terms that are above a certain value, called the Term Cutoff, however, that all terms have a topic weight for each topic, although it might be a very small value.

Every document receives a topic weight for each topic. The documents with topic weight values above the **document-cutoff** for this topic are included in that particular topic.

TextTopic_raw1 - TextTopic_raw10 – These are numeric variables that indicate the strength a particular topic has within a given document. Three topics were generated because this was specified on the Property Sheet. These variables are the same as the topic weight values for the documents that were previously looked at in the Documents window of the interactive Topic Viewer. Each of these variables (topics) has a label (the five most descriptive terms) to identify it and help the user interpret the topic.

TextTopic_1 - TextTopic_10 – These are binary variables defined for each document and constructed from the TextTopic_raw1 - TextTopic_raw3 values based on the document cutoff values described earlier. For example, TextTopic_1 is set to 1 if a document has a TextTopic_raw1 value greater than the cutoff value for this particular topic. Otherwise, it is set to 0.

6. **Metadata:**

   The Metadata node is used to modify attributes to the decision tree model. Initially while creating, the data course, we used only synopsis as input variable and all other variables were set as rejected. The metadata node allows to assign new role to the variables. We assign 'target' to the 10 Genres and

7. **Decision Tree:** (Output for Decision Tree Model 5 shown here)

   The topics extracted is used as inputs in a predictive model. Each topic represents an input variable. In our case, we have 10 input variables because the topic node extracted 10 topics. We used 75:25 split in creating training and validation data sets for building predictive models. The target variable used for modeling is the movie genre based on the movie synopsis, which is determined by text topic variables. We used Decision Tree, which have been suggested by prior researchers as better for textual data.

In all the models, we used Average Square Error and misclassification rate as the model selection criteria. We used default options for all other properties in the decision tree model.

Fit Statistics

Target	Target Label	Fit St	Statistics Label	Train	Validation
Action		_...	Sum of Frequencies	1145	382
Action		_...	Misclassification Rate	0.143231	0.159686
Action		_...	Maximum Absolute Error	0.98	1
Action		_...	Sum of Squared Errors	247.9972	95.99368
Action		_...	Average Squared Error	0.108296	0.125646
Action		_...	Root Average Squared Error	0.329083	0.354466
Action		_...	Divisor for ASE	2290	764
Action		_...	Total Degrees of Freedom	1145	.
Comedy		_...	Sum of Frequencies	1145	382
Comedy		_...	Misclassification Rate	0.396507	0.376963
Comedy		_...	Maximum Absolute Error	0.9	0.9
Comedy		_...	Sum of Squared Errors	509.594	173.6162
Comedy		_...	Average Squared Error	0.22253	0.227246
Comedy		_...	Root Average Squared Error	0.471731	0.476704
Comedy		_...	Divisor for ASE	2290	764
Comedy		_...	Total Degrees of Freedom	1145	.
Documentary		_...	Sum of Frequencies	1145	382
Documentary		_...	Misclassification Rate	0.006114	0.010471
Documentary		_...	Maximum Absolute Error	0.990826	1
Documentary		_...	Sum of Squared Errors	13.6749	8.05359
Documentary		_...	Average Squared Error	0.005972	0.010541
Documentary		_...	Root Average Squared Error	0.077276	0.102671
Documentary		_...	Divisor for ASE	2290	764
Documentary		_...	Total Degrees of Freedom	1145	.
Drama		_...	Sum of Frequencies	1145	382
Drama		_...	Misclassification Rate	0.388646	0.397906
Drama		_...	Maximum Absolute Error	0.890244	1
Drama		_...	Sum of Squared Errors	504.5528	175.4487
Drama		_...	Average Squared Error	0.220329	0.229645
Drama		_...	Root Average Squared Error	0.469392	0.479213
Drama		_...	Divisor for ASE	2290	764
Drama		_...	Total Degrees of Freedom	1145	.
Horror		_...	Sum of Frequencies	1145	382
Horror		_...	Misclassification Rate	0.054148	0.057592

Decision Tree 5 – Tree diagram and Fit statistics

Similary the result shows for all the 10 genres

8. Model comparison:

This node is one of the most powerful feature of SAS E-miner. We used the Model Comparison node to benchmark model performance and find a champion model among the different Decision Tree nodes in our process flow diagram. The Model Comparison node enables you to judge the generalization properties of each predictive model based on their predictive power, lift, sensitivity, profit or loss, and so on. Here, we used the average squared error and misclassification rate to compare the 5 fitted different decision tree models. These models are based on the different parameters of the text filter and text topic nodes.

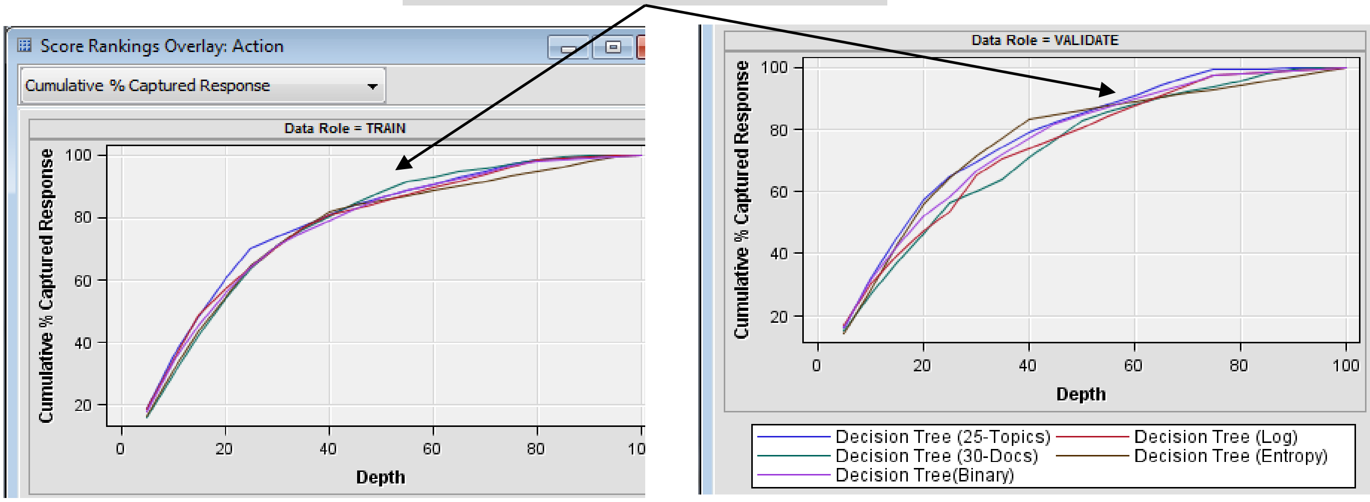
Averaged square error – Train (0.10830), Validation (0.12565)
Misclassification rate – Train (0.14323), Validation (0.15969)

Fit Statistics
Model Selection based on Valid: Misclassification Rate (_VMISC_)

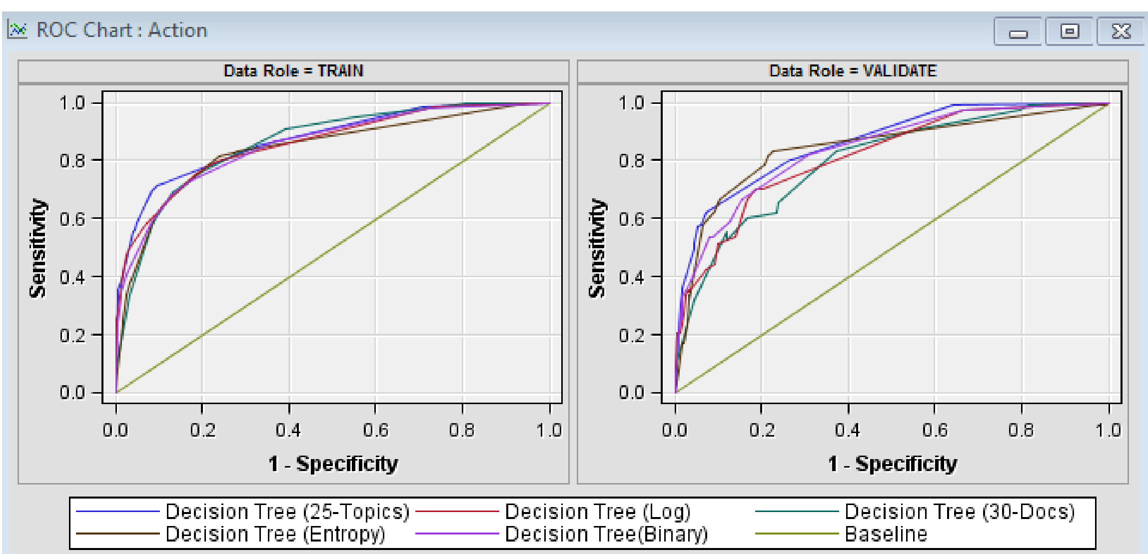
Selected Model	Model Node	Model Description	Valid: Misclassification Rate	Train: Average Squared Error	Train: Misclassification Rate	Valid: Average Squared Error
Y	Tree5	Decision Tree (25-Topics)	0.15969	0.10830	0.14323	0.12565
	Tree2	Decision Tree (Entropy)	0.16492	0.12585	0.17205	0.12909
	Tree	Decision Tree(Binary)	0.18848	0.12172	0.17031	0.13825
	Tree4	Decision Tree (Log)	0.20419	0.11673	0.15721	0.14558
	Tree3	Decision Tree (30-Docs)	0.21990	0.12556	0.17467	0.15486

Comparison of Models 1 to 5 on training and validation datasets

Model 5 performs better than others
- both on training and validation



Cumulative % of captured responses of Training & Validation test



ROC curve of Training & validation set

The 2nd graph shows cumulative captured response % by depth of the tree. Decision Tree with 25 topic models performs better in terms of capturing responses and classification on both training and validation datasets.

The 3rd graph shows the ROC curve for both training and validation datasets for all the 5 models. We observe that DT with 25 topic model have highest lift among all the others.

Sensitivity ~ 85% at 40% of dataset which means that the model is able to capture 85% of correct genres at 40% of target dataset.

Changing parameters and measuring effect on accuracy of the classifier:

1) Minimum number of topics:

We examined the effect of increasing the number of topics only in the text topic model node from 10 to 25, while keeping other parameters constant. We observed that increasing the numbers of topics from 10 to 25 increased the accuracy of the model – the rms error reduced on both training and validation datasets. This is because the synopsis has more categories than the 10 genres (topics) selected for analysis. Increasing the number of topics led to better topics being extracted which were different from each other and were characterized by top words by term and frequency weight respectively.

2) Minimum number of documents:

By minimum number of documents, classifier eliminates those terms for topic consolidation which are not present in the document number specified. Here, we changed the min. no. of documents from 4 to 30. The results were observed in the **Document cutoff value and Term cutoff value** with increase in these values when the count is increased.

Topic ID ▲	Document Cutoff	Term Cutoff	Topic	Number of Terms	# Docs
1	0.169	0.034	+show,+recommend,acting,+rate,+movie	180	183
2	0.161	0.035	hollywood,+protagonist,+play,+film,+cinema	182	120
3	0.083	0.035	+effect,+action,earth,+war,science	204	172
4	0.127	0.035	+motion,+moment,+viewer,+picture,+character	208	192
5	0.103	0.034	movie,granger,gauge,+comedy,+writer	111	148
6	0.098	0.035	+comedy,+funny,humor,+joke,+laugh	153	195
7	0.117	0.035	+woman,+man,+people,+american,+job	198	154
8	0.104	0.034	+bond,+car,bond,+action,+cop	163	158
9	0.108	0.034	best,+win,+oscar,+nominate,+nomination	126	92
10	0.114	0.034	+school,+mother,+kid,+girl,+son	145	184

Term cutoff, document cutoff of classifier with min. 30 documents

Topic ID	Document Cutoff	Term Cutoff	Topic	Number of Terms	# Docs
1	0.142	0.013	+show,+movie,+rate,+recommend,acti...	862	179
2	0.133	0.013	hollywood,+protagonist,+play,+film,+plot	970	115
3	0.106	0.013	+motion,+viewer,+picture,+moment,+c...	1232	175
4	0.086	0.013	+man,+woman,+american,+people,+c...	1251	142
5	0.078	0.013	+comedy,+funny,+joke,humor,+laugh	1201	181
6	0.058	0.013	+action,+effect,+war,earth,science	1136	129
7	0.062	0.013	+cop,+crime,+thriller,police,+action	1146	168
8	0.077	0.013	granger,gauge,movie,+mother,+child	1060	177
9	0.081	0.013	best,+oscar,+win,actor,picture	870	78
10	0.096	0.013	+bond,bond,connery,james,jeffrey	863	92

Term cutoff, document cutoff of classifier with min. 4 documents

3) Frequency Weight

Frequency weight is measure of calculating the frequency of a term in a document. Binary method assigns 1 for presence of term in a document and 0 for absence of a term in a document. This removes repetitive terms in a document. Log based frequency weighting removes the effect of terms that occurs multiple times in a document. Here, we used Log weight as default and changed to Binary based weighting. The results were observed in document cut off values.

Topics

Category	Topic ID	Document Cutoff	Term Cutoff	Topic	Number of Terms	# Docs
Multiple	1	0.132	0.013	+show,+reco...	1006	176
Multiple	2	0.126	0.013	hollywood,+pr...	1093	117
Multiple	3	0.105	0.013	+motion,+relat...	1323	161
Multiple	4	0.080	0.013	granger,gaug...	976	149
Multiple	5	0.065	0.013	+effect,+action...	1240	168
Multiple	6	0.076	0.013	+comedy,+fun...	1329	152
Multiple	7	0.085	0.013	+fellow,princip...	1377	130
Multiple	8	0.064	0.013	+crime,+cop,+...	1260	181
Multiple	9	0.072	0.013	best,+oscar,pi...	1212	132

Document cutoff value for Binary Frequency weighting based Classifier

4) Term Weight:

Term weights are useful for distinguishing important terms from others. The value helps in categorizing documents in which the terms exists many times. Here, we changed the term weight from Inverse Document Frequency (IDF) to Entropy. In this implementation the document cut off value differed from Entropy based classifier to IDF based classifier with IDF giving more weightage to topic segmentation based on documents.

Topics

Category	Topic ID	Document Cutoff	Term Cutoff	Topic	Number of Terms	# Docs
Multiple	1	0.142	0.013	+show,+movie...	862	179
Multiple	2	0.133	0.013	hollywood,+pr...	970	115
Multiple	3	0.106	0.013	+motion,+view...	1232	175
Multiple	4	0.086	0.013	+man,+woma...	1251	142
Multiple	5	0.078	0.013	+comedy,+fun...	1201	181
Multiple	6	0.058	0.013	+action,+effect...	1136	129
Multiple	7	0.062	0.013	+cop,+crime,+...	1146	168
Multiple	8	0.077	0.013	granger,gaug...	1060	177
Multiple	9	0.081	0.013	best,+oscar,+...	870	78

Document cutoff value for IDF based Classifier

Category	Topic ID	Document Cutoff	Term Cutoff	Topic	Number of Terms	# Docs
Multiple	1	0.124	0.013	+show,+rate,+...	950	176
Multiple	2	0.119	0.013	hollywood,+pr...	1042	115
Multiple	3	0.097	0.013	+motion,+view...	1339	181
Multiple	4	0.077	0.013	+woman,+mot...	1282	169
Multiple	5	0.074	0.013	+comedy,+jok...	1247	177
Multiple	6	0.071	0.013	science,movie...	1140	152
Multiple	7	0.068	0.013	+cop,+crime,+...	1182	169
Multiple	8	0.086	0.013	+bond,bond,c...	831	48
Multiple	9	0.079	0.013	best,actor,+wi...	786	52

Document cutoff value for Entropy based Classifier

By changing the parameters in the above methods, the term cut off and document cut off values are changing which varies the topics available for classification thus changing the errors and effecting the efficiency of model.