Genre Specific Aspect Based Sentiment Analysis of Movie Reviews

Viraj Parkhe
Department of Computer Science and Engineering
IIT-(BHU), Varanasi
Varanasi, India
viraj.parkhe.cse10@iitbhu.ac.in

Bhaskar Biswas
Department of Computer Science and Engineering
IIT-(BHU), Varanasi
Varanasi, India
bhaskar.cse@iitbhu.ac.in

Abstract— *The opinion conveyed by the user towards a movie can be understood by doing Sentiment Analysis on the movie review. In the current work we focus on Genre Specific Aspect Based Sentiment Analysis of Movie Reviews. Using the aforementioned dataset and considering movie genres like action, comedy, crime, drama and horror, we develop a fine grained unsupervised analysis model using lexicons that are context specific to each genre under consideration.*

Keywords— Sentiment Analysis; Movie genres; Movie Aspects; Lexicon; Context specific

I. Introduction

Sentiment analysis [1] is a methodology by which we find out the sentimental orientation of a piece of text. Using it, we can infer whether a particular person has conveyed a positive or negative sentiment in the said text under consideration. We tackled the issue of aspect based sentiment analysis of movie reviews in our previous publication [2]. In it we use the concept of "driving factors", which enhanced the overall classification accuracy by amplifying the effect of certain movie aspects with respect to others. In the current work, we tend to use the same concept, but for reviews with different genre. Many researchers have done work on Aspect based analysis of review, be it movie or customer review. Also many algorithms have been developed for the same. But not much work has been done on genre specific aspect based analysis. Genre specific reviews demand special techniques while analysing as such reviews contain sentences or words that have unique meaning based on the context i.e. genre in which they are used. Thus in this paper we try to develop an unsupervised aspect based analysis model that uses context i.e. genre specific lexicons. We made use of separate lexicons for each genre, and using this we try to inculcate context sensitivity into the model. Also using aspect based analysis, we try to develop fine grained aspect level analysis model.

Bo Pang et al [3] suggested many analysis methods including supervised models like Naïve Bayes, Support Vector Machines and Maximum Entropy classifiers. Using different features like unigrams, unigrams and bigrams, top unigrams, adjectives, they carried out experiments and compared their results. Their work gave researchers great insight into the field of sentiment analysis. Research conducted by Abd. Samad Hasan Basari et al [4] increased the overall accuracy of analysis models like Support Vector Machines by coupling them with Particle Swarm Optimization algorithm. Their work increased the overall accuracy obtained by them from 71.87% to 77%. Recent years have shown an increase in aspect based sentiment analysis as better analysis of review can be done if polarities of individual aspects are considered. Tun Thura Thet et al [5] developed a model for fine-grained sentiment orientation and sentiment strength analysis of various aspects of the movie. They formulated domain and generic opinion lexicons to score words in the review. The individual word score obtained via these lexicons was propagated over the entire document by using the inter-word dependencies obtained through the usage of dependency trees. A feature based heuristic model for aspect level sentiment analysis was developed by V.K. Singh et al [6]. In their scheme they identified various movie aspects in the review text and assigned labels to them. Each aspect text is then scored using SentiWordNet [7] with feature selection comprising of adverbs, adjectives, n-grams and verb features. They obtained the overall document score by averaging individual aspect score. Maite Taboada et al [8] in their work Genre-Based Paragraph Classification for Sentiment Analysis distinguished between different types of paragraphs of movie reviews using a taxonomy and classification system. They identified two types of paragraphs: formal vs. functional type and within the latter identified two more types: description and comment. Using this methodology of classification, they achieved an improvement over the baseline algorithm without paragraph classification. Alistair Kennedy et al [9] used the idea of valence shifter in their paper. They considered three types of shifters: negation, intensifiers and diminishers. They studied various methods including term count based methods and machine learning based methods and applied the concept of valence shifters over these methods. Jurgen Broß et al [10] proposed an unsupervised method to generate a context aware sentiment lexicon for product reviews, by utilising the semistructuredness of user generated review. They used statistical approach to find out the orientation of words in the lexicon, thus inducing context sensitivity. The rest part of the paper is organized as follows: Section II describes the proposed method, Section III describes the dataset, results and performance and Section IV gives the conclusion and future scope of work and in the end are references.

II. Proposed Method

The method aims at developing a lexicon based aspect oriented analysis approach for genre specific reviews. Fig. 1 describes the flow of the proposed method.

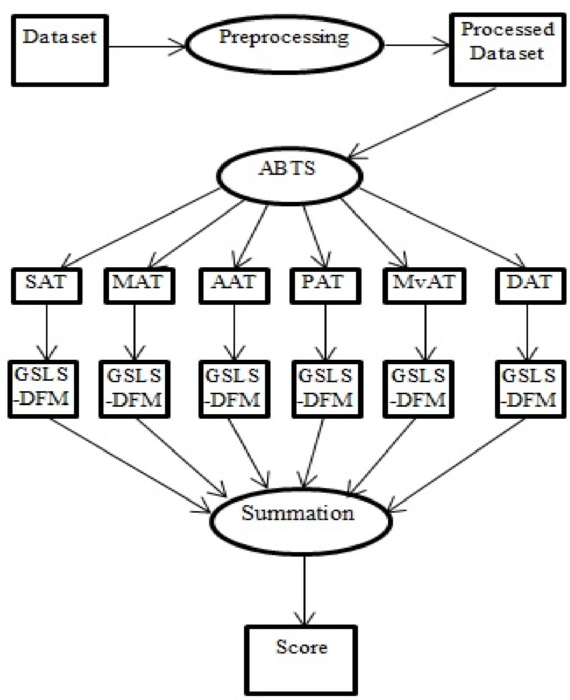

Fig. 1 Diagram for the proposed method

ABTS= Aspect Based Text Separator, SAT = Screenplay Aspect Text, MAT = Music Aspect Text, AAT = Acting Aspect Text, PAT = Plot Aspect Text, MvAT = Movie Aspect Text, DAT = Direction Aspect Text, GSLS-DFM = Genre Specific Lexicon based Scoring and Driving Factor Multiplication.

We used the dataset that Mahesh Joshi, Dipanjan Das, Kevin Gimpel, and Noah A. Smith [11] used in their experiments. For further details on the dataset please refer Section III. The dataset was in XML format and each file contained movie details like name of the movie, genre of the movie, date of release and also sited web links for obtaining full reviews. Pre-processing was required for the dataset as it was not in accordance with our requirements. We extracted the links and the genre of each movie from the dataset. Then from the corresponding links we acquired the full movie review manually. However the dataset didn't contain ratings for the movies. We needed the movie ratings so that we may compare the classification obtained using our methodology with the actual movie rating and determine the accuracy of our classification. We used movielens dataset [12] for determining the movie rating. Having done so, we prepared a pre-processed dataset containing information like the name of the movie, its genre, its rating and the review text. This dataset was then used further for experimental purposes.

Now the next step was that of separating the review text into aspect specific text. As mentioned in our previous publication [2] we tend to use Aspect Based Text Separator (ABTS) for this purpose. ABTS separates the review text into different groups based on the movie aspects. It does this using an aspect lexicon. The process of ABTS separation and solution to problems like ambiguity are explained thoroughly in the previous publication. The next step was that of the classification of these separated aspect texts. As mentioned previously, sometimes certain words tend to have different meaning based on the context in which they are used. For example take the word 'funny'. It tends to denote a positive orientation if used in the review of a comedy movie, as one would expect a comedy movie to be funny. But if used in a movie of horror genre, then the orientation of the word becomes somewhat negative as horror movies tend not to be funny. People, in general, tend to enjoy horror movies that scare them and which are thrilling. If someone uses the word funny to describe a horror movie, then there is a very good chance that the word was used in a negative sense. Thus to account for all these context sensitive words, we tend to develop a genre specific lexicon. This lexicon would contain certain words whose orientation would depend on the genre in which they are used. We formulated a list of top 500 frequently used adjectives in everyday life and formed a lexicon out of these words. . Now we wanted to assign orientation to these words based on the movie genre. Thus we used the methodology of Semantic orientation to do so.

Before we define what Semantic Orientation (SO) [13] is, let us first define Pointwise Mutual Information (PMI). PMI between two words is the amount of information that we acquire about the presence of one word when we observe the other [14]. The formula for PMI is:

$$\mathrm{PMI}(\mathrm{word}_1,\mathrm{word}_2) = \log_2\left(\frac{p(\mathrm{word}_1 \& \mathrm{word}_2)}{p(\mathrm{word}_1)p(\mathrm{word}_2)}\right)$$

Here $p(word_1 \& word_2)$ is the probability of $word_1$ and $word_2$ occurring together. If the words are statistically independent, then the probability of their co-occurrence is given by $p(word_1)p(word_2)$. The degree of statistical dependence between words is given by the ratio of $p(word_1 \& word_2)$ and $p(word_1)p(word_2)$. The semantic orientation of a word, *word* is calculated by:

$$\mathrm{SO}(\mathrm{word}) = \mathrm{PMI}(\mathrm{word},\mathrm{X}) - \mathrm{PMI}(\mathrm{word},\mathrm{Y})$$

Here X denotes a positively oriented word or string of words and Y denotes a negatively oriented word or a string of words. Thus we find the co-occurrence of *word* with a positive word and with a negative word. Then we subtract the PMI obtained with the positive and negative word to get the overall orientation of the word. Thus if we obtain a negative value, the overall SO is negative, and it means that the word under consideration occurs more closely with the negative word string and similarly if the result the positive, the word is closely associated with the positive word string. Thus the SO methodology can be easily used for preparing genre specific lexicon. Now the question that remained was that of finding the co-occurrence of words. Previously researchers used to use AltaVista search engine and it's NEAR operator to do so [14]. But since AltaVista is no longer operational, we decided to query a movie dataset instead of the internet to find out the co-occurrence of words. The query which we issued over the dataset returned us with the hitcount of our search query. This hitcount was then used to calculate the probability for finding the co-occurrence. We used the Large Movie Review Dataset [15] for the same. The NEAR operator functionality was programmatically recreated using Boolean operators to work on our dataset. Two find co-occurrence of two words, *word1* and *word2* we issued a Boolean query over our dataset as: "word1 word2" OR "word2 word1" OR "word1 * word2" OR "word2 * word1" [16]. The above query considers all the cases related to the positioning of the words. Here '*' represents a wildcard, which means there can be single or multiple words between the two words. As mentioned earlier we calculated the proximity of the adjective towards both positive and negative words or strings, via acquired hit count. We prepared a positive and negative string for each genre under consideration. Consider the genre 'action'. For this genre we prepared a positive string as follows "action AND good" and negative string as "action AND bad". When the adjective was queried along with this string, the results returned the documents in which the words 'action' and 'good' co-occurred along with the adjective. Similarly strings were created for other genres like comedy, crime, drama and horror. We worked on the assumption that in the reviews, the user might have mentioned to which genre the movie belonged to. It was a fair assumption according to us as most of the reviews we encountered had some indication contained in them regarding the genre of the movie.

After hitcount collection, the SO of the adjective was calculated according to the given formula. The SO values were normalised and value was brought between -1 and 1 before storing the values in the lexicon. This process was carried on for all 500 adjectives and for each adjective, we calculated the SO for each genre. The SO values for each genre corresponding to each adjective were stored in a file and a genre specific lexicon was formed. After the review is passed through the ABTS, the separated aspect texts are forwarded to be scored using the Genre Specific Lexicon Scorer (GSLS). Here we extract the adjectives from the text and score it using the lexicon. If the adjective is present in the lexicon, we give it a score corresponding to the genre of the review which is under consideration. If the adjective is not present in the lexicon, then the adjective is scored using the SentiWordnet [7] dictionary. SentiWordnet contains sentimental scores, which are not context specific, for huge collection of adjectives, adverbs and nouns.

We also consider the effect of negation and intensifiers [9] on the adjective that is being scored. Negations are the words that change the polarity of the adjective. Consider that we have a statement such as "The film was not good" in the review. If we don't consider the effect of negation, then by the above methodology, we will only score the word "good" in the sentence and compute the overall score of the sentence. Since "good" denotes a positive adjective across all the genres, the overall score of the sentence will become positive, which is not the case as we have a word "not" in front of "good". This is called negation effect. To handle this we always keep track of previous word while considering a certain word. If the previous word is a negation, and the current word is an adjective, then the score of the adjective is reversed i.e. if the score of adjective is positive we make it negative and vice versa. Intensifiers also have an impact over the adjective. Intensifiers are the words that enhance the score of the adjective. Suppose we have word "good", then a word like "very" is called an intensifier as the score of string "very good" is better than "good". Thus if we encounter an intensifier before an adjective, we extract the score of the intensifier from SentiWordnet, add that score with 1 and multiply the net score with the score of the adjective. Thus the overall score of the adjective is intensified. Here in this methodology we have only considered bi-grams (two words) while analysing the impact of negations and intensifiers. For each aspect text, we score all the adjectives present in it using the above method. After all the adjectives have been scored, we average the value of all the scores and assign this averaged value as the score for the aspect text. We follow this process for all the aspect texts. Thus what we get are single scores for all the aspect texts. The next step is the application of driving factors on these scores (DFM). As mentioned in [2] driving factors are used for amplifying the importance of certain aspects in the overall classification process and also in fine grained analysis of the review under consideration. After application of driving factors, we follow the summation process in which a proper threshold was set and review scores were compared with this threshold, and review classification based on this comparison was done.

III. Dataset, results and performance

The dataset we used consists of reviews of 1718 movies, released during the year 2005 to 2009 from the following sources [11]:

- Austin Chronicle (www.austinchronicle.com) -- 462 reviews
- Boston Globe (www.boston.com) -- 731 reviews
- LA Times (www.calendarlive.com) -- 625 reviews

- Entertainment Weekly (www.ew.com) -- 1039 reviews
- New York Times (www.nytimes.com) -- 1375 reviews
- Variety (www.variety.com) -- 1454 reviews
- Village Voice (www.villagevoice.com) -- 1396 reviews

Each file in the dataset consisted of the following information: the name of the movie, the genre of the movie, some financial information regarding the movie, and a list of links to various sites, from which we obtained the full movie review. Using the links we scraped the internet and prepared a dataset of plain text reviews for experimental purposes. TABLE I shows the results obtained by applying the above methodology across all the considered genres. It shows the highest and the lowest accuracy obtained for each genre corresponding to the driving factors considered. We used the following performance measures for our experiment:

$$Accuracy = \frac{Total\ correctly\ classified\ documents}{Total\ number\ of\ documents}$$

$$Precision = \frac{tp}{tp + fp}$$

$$Specificity = \frac{tn}{\mathrm{N}}$$

$$Recall = \frac{tp}{\mathrm{P}}$$

$$F - 1\ score = \frac{2 * precision * recall}{precision + recall}$$

Here *tp*, *fp* and *tn* are the true positives, false positives and true negatives obtained during the classification. *P* refers to total positively oriented reviews in the dataset. *N* refers to total negatively oriented documents in the dataset. TABLE II through TABLE VI shows the performance across all the genres. Here only top 5 results w.r.t. accuracy are displayed.

As we can see from the results we have the following highest accuracies for different genres: 0.7148 for action genre, 0.70464 for comedy genre, 0.802120 for crime genre, 0.812196 for drama genre and 0.638418 for horror genre. All the genres have got pretty good accuracy except horror genre. This might be for the fact that the lexicon which we used might not have covered all the relevant adjectives that might have been used in the horror genre review. Sarcastic comments too might have resulted in a low accuracy. In the above methodology, we have used bi-grams to determine the effect of negation and intensifiers. Using tri-gram and other n-gram features can result in a better accuracy. For ex. the sentence “not very good” will have a positive score according to our algorithms as it will only score the words “very” and “good” and not the word “not”. Using tri-grams can solve this problem as it will also consider the word “not”.

TABLE II. PERFORMANCE MEASURES FOR ACTION GENRE

Accuracy	Precision	Specificity	Recall	F-Score
0.7148	0.6150870	0.022113	0.9906542	0.7588
0.7139	0.6133971	0.007371	0.9984423	0.7598
0.7053	0.6177685	0.090909	0.9314641	0.7427
0.6910	0.6141479	0.1154791	0.8925233	0.7275
0.6815	0.6124031	0.14004914	0.8613707	0.7158

TABLE III. PERFORMANCE MEASURES FOR COMEDY GENRE

Accuracy	Precision	Specificity	Recall	F-Score
0.70464	0.7044307	0.0330275	0.9944576	0.8248
0.70188	0.70133481	0.0146788	0.9984164	0.8239
0.70077	0.70122630	0.0165137	0.9960411	0.8229
0.69856	0.70111482	0.01284403	0.9944576	0.8216
0.67754	0.70000	0.06422018	0.9422013	0.8032

TABLE IV. PERFORMANCE MEASURES FOR CRIME GENRE

Accuracy	Precision	Specificity	Recall	F-Score
0.802120	0.80071	0.034482	1.000	0.8880
0.798586	0.79787	0.017241	1.000	0.8875
0.787985	0.80000	0.051724	0.977	0.8796
0.777385	0.79779	0.051724	0.964	0.8730
0.766784	0.79120	0.017241	0.960	0.8674

TABLE V. PERFORMANCE MEASURES FOR DRAMA GENRE

Accuracy	Precision	Specificity	Recall	F-Score
0.812196	0.811993	0.005714	1.000000	0.8961
0.811656	0.812229	0.008571	0.998664	0.8958
0.806799	0.813355	0.025714	0.988689	0.8924
0.785752	0.812429	0.051428	0.956753	0.8786
0.778197	0.810580	0.048571	0.948103	0.8739

TABLE VI. PERFORMANCE MEASURES FOR HORROR GENRE

Accuracy	Precision	Specificity	Recall	F-Score
0.638418	0.644171	0.121212	0.945945	0.7663
0.632768	0.659722	0.257575	0.855855	0.7450
0.627118	0.653061	0.227272	0.864864	0.7441
0.621468	0.626436	0.015151	0.981981	0.7648
0.610169	0.632911	0.121212	0.900900	0.7431

IV. CONCLUSION AND FUTURE SCOPE OF WORK

Using the driving factors in the above mentioned methodology we got the following results: for action genre we got plot, movie and direction as the most important factors as these aspects had the highest values, for comedy we got acting, plot and movie, for crime we got screenplay, music and plot, for drama we got music, movie and direction and for horror we got music, direction and movie. These results obtained are only for the particular dataset under consideration. Other datasets may give different results, but by plain observation of the results one can say that they are somewhat correct. Take the example of comedy genre. In this genre people like the way actor act and make them laugh. Also the storyline and hilarious plot of the movie might make someone crack a smile. Thus these things are evident from the results we got for the comedy genre. Using the driving factors we were able to extract the most importance aspects for a particular dataset under consideration. Thus by using this methodology, we can identify importance aspects across various datasets and across various genres. Using this knowledge, we might try to develop a fine grained recommendation system which recommends the user with movies not only on ratings, but also on the aspects about the movies he likes. For ex. say a user may like acting and background score rather than the plot of the movie. Thus using the factors obtained from various dataset, we can recommend him/her a movie from the genre which has high driving factors for acting and music. This is just a hypothetical system we are trying to present here. We are merely trying to suggest in what types of domains the driving factors can be used. Also instead of using it on movie review, we can use them on customer reviews for business analytics and marketing of the product. The customers give their opinion about various product aspects like its performance, usability, cost, build quality etc. and thus creating a domain for driving factor usage. The research can have a considerable application on reviews in Indian languages too. However we have to identify the aspect words, and also the adjectives used for generating genre specific lexicon, in the language under consideration carefully. Also SentiWordnet will have to be developed in the considered language. Having said so, further study is needed for feasible application of this model on Indian languages. We have only used the concept of negation, intensifiers and genre specific lexicon to induce some knowledge about inter-word dependencies in the algorithm. Various other techniques like dependency tree, clause based scoring [5] can be used for further detailed analysis.

REFERENCES

[1] Sentiment Analysis, Wikipedia – The Free Encyclopedia (http://en.wikipedia.org/wiki/Sentiment_analysis)

[2] Viraj Parkhe, Bhaskar Biswas, "Aspect Based Sentiment Analysis of Movie Reviews: Finding the polarity directing aspects", International Conference on Soft Computing and Machine Intelligence, New Delhi, India, 2014.

[3] Bo Pang and Lillian Lee, Shivakumar Vaithyanathan, "Thumbs up? Sentiment Classification using Machine LearningTechniques", Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), Philadelphia, July 2002, pp. 79-86.

[4] Abd. Samad Hasan Basaria, Burairah Hussina, I. Gede Pramudya Anantaa, Junta Zeniarjab, "Opinion Mining of Movie Review using Hybrid Method of Support Vector Machine and Particle Swarm Optimization", in Malaysian Technical Universities Conference on Engineering & Technology 2012, MUCET 2012 Part 4 - Information And Communication Technology.

[5] Tun Thura Thet, Jin-Cheon Na and Christopher S.G. Khoo, "Aspect-based sentiment analysis of movie reviews on discussion boards"., in Journal of Information Science,2010 36:823.

[6] V.K. Singh, R. Piryani, A. Uddin and P. Waila, "Sentiment Analysis of Movie Reviews A new Feature-based Heuristic for Aspect-level Sentiment Classification", Proceedings of the 2013 International Muli-Conference on Automation, Communication, Computing, Control and Compressed Sensing, Kerala-India, March 2013, IEEE Xplore, pp. 712-717.M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[7] SentiWordNet, lexical resource for opinion mining,(http://sentiwordnet.isti.cnr.it/).

[8] Maite Taboada, Julian Brook, and Manfred Stede ,"Genre-Based Paragraph Classification for Sentiment Analysis" ,Proceedings of SIGDIAL 2009: the 10th Annual Meeting of the Special Interest Group in Discourse and Dialogue, pages 62–70,Queen Mary University of London, September 2009.

[9] Alistair Kennedy and Diana Inkpen, Sentiment Classification of Movie and Product Reviews Using Contextual Valence Shifters, University of Ottawa, Ottawa, Canada.

[10] J¨urgen Broß and Heiko Ehrig", Generating a Context-Aware Sentiment Lexicon for Aspect-Based Product Review Mining", IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, 2010.

[11] Mahesh Joshi, Dipanjan Das, Kevin Gimpel, and Noah A. Smith. Movie Reviews and Revenues: An Experiment in Text Regression. In Proceedings of NAACL-HLT (Short paper track), Los Angeles, CA, June 2010.

[12] Movielens dataset from GroupLens research group, http://www.grouplens.org; IMDb: http://www.imdb.com; Rotten Tomatoes : http://www.rottentomatoes.com.

[13] Peter D. Turney, Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews, Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL), Philadelphia, July 2002, pp. 417-424.

[14] P.D. Turney, M.L. Littman, Unsupervised Learning of Semantic Orientation from a Hundred-Billion-Word Corpus, National Research Council of Canada.

[15] Large Movie Review Dataset, Acquired from Stanford AI Lab: (http://ai.stanford.edu/~amaas/data/sentiment).

[16] Proximity Search, Wikipedia- The free Encyclopedia (http://wikipedia/wiki/proximity_search)

TABLE I EXPERIMENTAL RESULTS OBTAINED ACROSS ALL THE GENRES CONSIDERED

Genres		Accuracy	S-DF	M-DF	A-DF	P-DF	Mv-DF	D-DF
Action	Highest	0.7148	0.005134	0.008173	0.00232	0.19835	0.59100	0.1898
	Lowest	0.6090	0.554716	0.0010773	0.02132	0.227974	0.19255	4.21E-4
Comedy	Highest	0.70464	0.01589	0.05926	0.21036	0.58062	0.105890	0.0269
	Lowest	0.60785	0.110389	0.322646	0.081760	0.38561	0.092092	0.0058
Crime	Highest	0.8021201	0.649658	0.2793	0.001218	0.035368	1.77E-4	0.0251
	Lowest	0.7561830	0.980045	0.0042	0.006968	7.768E-4	1.941E-4	0.0077
Drama	Highest	0.81219	0.00788	0.11319	0.22646	1.0180E-	0.29045	0.361213
	Lowest	0.72369	0.48733	0.013263	0.001070	0.022910	0.362478	0.045260
Horror	Highest	0.6384180	0.025329	0.50345	0.00181	0.03021	0.0270	0.40506
	Lowest	0.5480225	0.975004	1.533E-4	2.88E-5	1.133E-5	0.02477	2.854E-5

Here *S-DF = Screenplay Driving Factor, M-DF=Music Driving Factor, A-DF = Acting Driving Factor, P-DF = Plot Driving Factor, Mv-DF =Movie Driving Factor, and D-DF =Direction Driving Factor*