CHAPTER 4

LATENT SEMANTIC ANALYSIS AND LAPLACIAN
SCORE FEATURE SELECTION FOR CLASSIFYING
OPINIONS

4.1 INTRODUCTION

Feature selection and extraction is an important task in OM and Sentimental Analysis (OSMA). An entity is a hierarchical component and its subcomponents representations are each being associated with a set of attributes, while voluminous documents are processed for sentiment with different features like n-grams, location based features, structural or discourse features, part-of-speech, lexicon based features and syntactic features (Kumar & Abirami 2015).

Another OM approach is using machine learning which uses text classification activities. Using machine learning has another effect. A good feature selection technique reduces features in the OM process. Feature selection chooses relevant features based on a specific measurement. Its main purpose is to simplify training and reduce training time. Performance of classifiers like k Nearest Neighbor is poor when there are many features. But, it is important to choose a feature selection technique that reduces features without reducing the OM’s performance. Many common feature selection techniques like Part-of-speech (POS), Document Frequency, Information Gain (IG), and Chi Square were incorporated in OM.
Feature selection gained importance due to its tendency of saving classification cost regarding time and computation load. Searching for features is through decision trees. The latter is an intermediate feature space inducer to choose essential features (Jeevanandam, Jotheeswaran & Kumaraswamy 2013). There are three feature selection technique groups i.e. filter, wrapper and embedded. In filter category, a features group is chosen based on a specific mathematical equation for use with any classifier. In contrast, features chosen by a wrapper and embedded techniques are bound to a specific classifier.

Other than being very rigid, wrapper and embedded techniques need high resource allocation and longer execution time (Samsudin et al 2013). There are varied techniques for feature selection like those based on minimum, Stemmed Terms, Dependence Y-relation, Term Frequency–Inverse Document Frequency (TF-IDF), Graph Distance, Opinion Words, Document Frequency, Mutual Information (MI), IG, CHI, N-gram of which some are famous (Kothari & Patel 2015).

4.2 FEATURE SELECTION METHODS

Feature Selection Methods are (Jeevanandam, Jotheeswaran & Koteeswaran 2015),

- Correlation based feature selector (CFS)
- Information Gain
- Support Vector Machine (SVM)
- Principal component analysis (PCA)
4.2.1 Correlation based Feature Selector (CFS)

“Correlation based Feature Selector (CFS)” is a filter algorithm ranking feature subsets according to correlation based heuristic evaluation. Its bias is to subsets with “features highly correlated to class and uncorrelated to others”. Irrelevant features are ignored as it will have low correlation with a class. Redundant features are screened out will have high correlation with one and many features. Feature acceptance is based on “the extent to which it predicts classes where space is not predicted by other features”. CFS’s feature subset evaluation function is given by:

\[ M_s = \frac{k \bar{r}_{cf}}{\sqrt{k + k(k-1)\bar{r}_{ff}}} \]

where \( M_s \) is heuristic merit of a feature subset.

\( S \) is the feature subset

\( \bar{r}_{cf} \) is the mean feature-class correlation \((f \in S)\)

\( \bar{r}_{ff} \) is the average feature-feature inter correlation”

4.2.2 Information Gain

Information Gain calculates an instance’s probability as it is a segment border comparing it to a segment border probability where a feature has a specific value. The higher a probability change, the more useful the feature (Isabella & Suresh 2012).

4.2.3 Support Vector Machine (SVM)

Support Vector Machines (SVMs) were highly effective at conventional text categorization, outperforming Naive Bayes. In a two-
category case, the idea behind a training procedure is finding a maximum margin hyperplane, represented by a vector $\mathbf{w}$, that separates document vectors in one class from those in another, but for which separation or margin, is large. This corresponds to a “constrained optimization problem; letting $c_j \in \{1, -1\}$ (corresponding to positive or negative) be the correct class of document $d_j$,” solution is written as

$$\mathbf{w} := \sum_j \alpha_j c_j \mathbf{d}_j, \quad \alpha_j \geq 0$$

where, “$\alpha_j$’s (Lagrangian multipliers) are obtained by solving a dual optimization problem”.

Those $\mathbf{d}_j$ so that $\alpha_j$ is greater than zero are support vectors, as they are the only document vectors contributing to $\mathbf{w}$.

Test instances classification comprises determining which side of $\mathbf{w}$’s hyperplane they fall on (Chandrakala & Sindhu 2012).

### 4.2.4 Principal Component Analysis (PCA)

Principal Components Analysis (PCA) reduces inputs dimensions when their dimensions are large and components are highly correlated. PCA determines smaller artificial variables set which represents variance of a set of observed variables. Artificial variables thus calculated are called principal components which are used as a predictor or criterion variable in other analysis. The variables are orthogonalized by PCA and the principal components with largest variation are chosen and those with least variations are eliminated from a dataset. PCA is applied on data as follows:
• “A dataset with a mean of zero is formed by subtracting mean of data from each data dimension.

• Covariance matrix is calculated.

• Eigenvectors and Eigenvalues of covariance matrix are calculated.

• Principal components of a dataset are represented by highest Eigenvalues and those of less significance are removed forming a feature vector.

• A new dataset is derived”

Feature set dimension is reduced using PCA and Learning Vector Quantization classifies the opinions (Jotheeswaran et al 2012).

4.3 METHODOLOGY

In this work Latent Semantic Analysis (LSA) and Laplacian Score (LS) feature selection are evaluated to classify opinions.

4.3.1 Latent Semantic Analysis (LSA)

Latent Semantic Analysis (LSA) maps text objects, usually documents and words, to a latent semantic space. Vectors proximity in space means that the original text objects are semantically related. But, LSA’s limitation is its inability to differentiate fine-grained relations. Synonyms and antonyms may be assigned high similarity scores when applied to lexical semantics. Asymmetric relations like hyponyms or hypernyms are not differentiated.

LSA has two steps. First, taking a collection of d documents with words from a vocabulary list of size n, it builds a d×n document-term matrix
W to encode a word’s occurrence information in that document. Simply, element $W_{i,j}$ can be a term frequency of $j^{th}$ word in $i^{th}$ document. A weighting scheme that better captures a word’s importance in a document like TF×IDF (Salton et al 1975) is used. LSA computes similarity between two documents or two words in latent space.

In a term-by-document matrix representing a documents collection, LSA applies “singular value decomposition (SVD)” to matrix to statistically estimate latent dimensions (or factors) and term-term associations of a collection”. Specifically, a term is built by a document matrix $X$, in a review documents corpus. Applying SVD decomposes term-by-document matrix into a product of three matrices (Hai et al 2012):

$$X = LVR$$

Where “L and R are left and right singular matrices, and V is a diagonal matrix of singular values”.

Let “$r$ be the rank of a raw matrix $X$. A value $k<r$ is selected. Let $V_k$ denote diagonal matrix generated by choosing top $k$ singular values from matrix $V$, and let $L_k$ and $R_k$ be matrices generated by selecting corresponding columns from matrices L and R, respectively. Thus we obtain a reduced matrix $X_k$ by multiplying the three new matrices”:

$$X_k = L_k V_k R_k$$

Matrix “$X_k$ is a best low rank (k) approximation of raw matrix $X$, that reduces the Frobenius norm or reconstruction error in form”:

$$\|E\|_F = \sqrt{\sum_{j=1}^{r} \sum_{d=1}^{D} (e_{jd})^2}$$
where “$E = X - X_k$, $e_{id}$: element of matrix E, $T$: term set size, and $D$: corpus size”.

In new latent space, measure pair-wise term associations via cosine similarity of corresponding row vectors of “smoothed” matrix $X_k$.

4.3.2 Laplacian Score (LS)

Laplacian Score (LS) (Wang 2014) is a feature selection “filter” method. LS assumes that data from the same class are close to each other in real world classification problems. A feature’s importance is evaluated by its locality preserving power. Locality preserving power is reflected by its Laplacian score for a feature. LS, based on distance is inspired by the possibility of identifying examples with affinity when relatively next to each other (Spolaôr et al 2011).

In a set of “samples $X = [x_1, x_2, \ldots, x_n]$”. Let $x_{ir}$ denote $r^{th}$ feature in $i^{th}$ sample. LS first builds a nearest neighbor graph $G$ with $n$ nodes. If nodes $i$ and $j$ are connected”, put

$$S_{ij} = \exp\left(-\frac{|x_i - x_j|^2}{t}\right)$$

Where “$t$ is a suitable constant”.

Otherwise, put “$S_{ij} = 0$. Weight matrix $S$ of a graph models the local data space structure”.

Then, define graph Laplacian matrix for the $r^{th}$ feature:

$$L = D - S$$
where \( D_u = \sum_{j=1}^{n} S_{ij} \)

Laplacian score of \( r \)th feature is expressed as:

\[
L_r = \frac{\tilde{f}_r^T L \tilde{f}_r}{\tilde{f}_r^T D \tilde{f}_r}
\]

\[
\tilde{f}_r = f_r - \frac{\sum_{i=1}^{n} x_i D_{ui}}{\sum_{i=1}^{n} D_{ui}}
\]

4.4 RESULTS AND DISCUSSION

Experiments were conducted for sentiment classification using online movie review data and medical service data set. From IMDb dataset 640 positive instances and 200 negative instances and from Medical Service Dataset 230 positive instances and 185 negative instances were used for evaluation.

Figures give the classification accuracy, precision and recall used for classifying the opinion into positive or negative. Table 4.1 & 4.2 and Figure 4.1 to 4.14 shows the results with the various classifiers for IMDb and medical service dataset respectively.
a. Using IMDb Dataset

Table 4.1 Summary of Results using IMDb Dataset

<table>
<thead>
<tr>
<th></th>
<th>LSA-NB</th>
<th>LSA-FLRC</th>
<th>LSA-AdaBoost</th>
<th>Laplacian-NB</th>
<th>Laplacian-FLRC</th>
<th>Laplacian-AdaBoost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classification Accuracy</td>
<td>0.8261</td>
<td>0.8318</td>
<td>0.8408</td>
<td>0.8592</td>
<td>0.8637</td>
<td>0.8892</td>
</tr>
<tr>
<td>Precision for Negative Opinion</td>
<td>0.8425</td>
<td>0.8469</td>
<td>0.8556</td>
<td>0.8712</td>
<td>0.8768</td>
<td>0.8889</td>
</tr>
<tr>
<td>Precision for Positive Opinion</td>
<td>0.8003</td>
<td>0.8082</td>
<td>0.8176</td>
<td>0.8407</td>
<td>0.8435</td>
<td>0.8896</td>
</tr>
<tr>
<td>Recall for Negative Opinion</td>
<td>0.8688</td>
<td>0.8742</td>
<td>0.8796</td>
<td>0.8946</td>
<td>0.8957</td>
<td>0.929</td>
</tr>
<tr>
<td>Recall for Positive Opinion</td>
<td>0.7641</td>
<td>0.7703</td>
<td>0.7844</td>
<td>0.8078</td>
<td>0.8172</td>
<td>0.8313</td>
</tr>
<tr>
<td>F measure for Negative Opinion</td>
<td>0.7818</td>
<td>0.7888</td>
<td>0.8007</td>
<td>0.8239</td>
<td>0.8301</td>
<td>0.8595</td>
</tr>
<tr>
<td>F measure for Positive Opinion</td>
<td>0.8554</td>
<td>0.8603</td>
<td>0.8674</td>
<td>0.8827</td>
<td>0.8861</td>
<td>0.9085</td>
</tr>
</tbody>
</table>

Figure 4.1 Classification Accuracy achieved by LSA-AdaBoost using IMDb Dataset
From Figure 4.1 it is seen that the classification accuracy achieved by LSA-AdaBoost is much better than that of LSA-Naïve Bayes and LSA-FLRC. LSA-AdaBoost achieves better classification accuracy by 1.7638% and 1.0762% than the LSA-Naïve Bayes and LSA-FLRC classifiers respectively. The classification accuracy achieved by Laplacian-AdaBoost is much better than that of Laplacian-Naïve Bayes and Laplacian-FLRC. Laplacian-AdaBoost achieves better classification accuracy by 3.4317% and 2.9095% than the Laplacian-Naïve Bayes and Laplacian-FLRC classifiers respectively.

![Figure 4.2 Precision for Negative Opinion achieved by LSA-AdaBoost using IMDb Dataset](image)

**Figure 4.2** Precision for Negative Opinion achieved by LSA-AdaBoost using IMDb Dataset

From Figure 4.2 it is seen that the precision for negative opinion achieved by LSA-AdaBoost is much better than that of LSA-Naïve Bayes and LSA-FLRC. LSA-AdaBoost achieves better precision by 1.5429% and 1.022% than the LSA-Naïve Bayes and LSA-FLRC classifiers respectively. The precision for negative opinion achieved by Laplacian-AdaBoost is much better than that of Laplacian-Naïve Bayes and Laplacian-FLRC. Laplacian-AdaBoost achieves better precision by 2.0112% and 1.3706% than the Laplacian-Naïve Bayes and Laplacian-FLRC classifiers respectively.
From Figure 4.3 it is seen that the precision for positive opinion achieved by LSA-AdaBoost is much better than that of LSA-Naïve Bayes and LSA-FLRC. LSA-AdaBoost achieves better precision by 2.1386% and 1.1564% than the LSA-Naïve Bayes and LSA-FLRC classifiers respectively.

The precision for positive opinion achieved by Laplacian-AdaBoost is much better than that of Laplacian-Naïve Bayes and Laplacian-FLRC. Laplacian-AdaBoost achieves better precision by 5.6522% and 5.3199% than the Laplacian-Naïve Bayes and Laplacian-FLRC classifiers respectively.
From Figure 4.4 it is seen that the recall for negative opinion achieved by LSA-AdaBoost is much better than that of LSA-Naïve Bayes and LSA-FLRC. LSA-AdaBoost achieves better recall by 1.2354% and 0.6158% than the LSA-Naïve Bayes and LSA-FLRC classifiers respectively. The recall for negative opinion achieved by Laplacian-AdaBoost is much better than that of Laplacian-Naïve Bayes and Laplacian-FLRC. Laplacian-AdaBoost achieves better recall by 3.7728% and 3.6499% than the Laplacian-Naïve Bayes and Laplacian-FLRC classifiers respectively.

![Recall for Positive Opinion](image)

**Figure 4.5 Recall for Positive Opinion achieved by LSA-AdaBoost using IMDb Dataset**

From Figure 4.5 it is seen that the recall for positive opinion achieved by LSA-AdaBoost is much better than that of LSA-Naïve Bayes and LSA-FLRC. LSA-AdaBoost achieves better recall by 2.6219% and 1.8139% than the LSA-Naïve Bayes and LSA-FLRC classifiers respectively. The recall for positive opinion achieved by Laplacian-AdaBoost is much better than that of Laplacian-Naïve Bayes and Laplacian-FLRC. Laplacian-AdaBoost achieves better recall by 2.8674% and 1.7106% than the Laplacian-Naïve Bayes and Laplacian-FLRC classifiers respectively.
Figure 4.6  F measure for Negative Opinion achieved by LSA-AdaBoost using IMDb Dataset

From Figure 4.6 it is seen that the F measure for negative opinion achieved by LSA-AdaBoost is much better than that of LSA-Naïve Bayes and LSA-FLRC. LSA-AdaBoost achieves better F measure by 2.3886% and 1.4973% than the LSA-Naïve Bayes and LSA-FLRC classifiers respectively. The F measure for negative opinion achieved by Laplacian-AdaBoost is much better than that of Laplacian-Naïve Bayes and Laplacian-FLRC. Laplacian-AdaBoost achieves better F measure by 4.2295% and 3.4801% than the Laplacian-Naïve Bayes and Laplacian-FLRC classifiers respectively.

Figure 4.7  F measure for Positive Opinion achieved by LSA-AdaBoost using IMDb Dataset
From Figure 4.7 it is seen that the F measure for positive opinion achieved by LSA-AdaBoost is much better than that of LSA-Naïve Bayes and LSA-FLRC. LSA-AdaBoost achieves better F measure by 2.3886% and 1.4973% than the LSA-Naïve Bayes and LSA-FLRC classifiers respectively. The F measure for positive opinion achieved by Laplacian-AdaBoost is much better than that of Laplacian-Naïve Bayes and Laplacian-FLRC. Laplacian-AdaBoost achieves better F measure by 4.2295% and 3.4801% than the Laplacian-Naïve Bayes and Laplacian-FLRC classifiers respectively.

b. Using Medical Service Dataset

Table 4.2 Summary of Results using Medical Service Dataset

<table>
<thead>
<tr>
<th></th>
<th>LSA-NB</th>
<th>LSA-FLRC</th>
<th>LSA-AdaBoost</th>
<th>Laplacian-NB</th>
<th>Laplacian-FLRC</th>
<th>Laplacian-AdaBoost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classification</td>
<td>0.8458</td>
<td>0.8602</td>
<td>0.8651</td>
<td>0.8668</td>
<td>0.8699</td>
<td>0.894</td>
</tr>
<tr>
<td>Accuracy</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Precision for</td>
<td>0.8103</td>
<td>0.829</td>
<td>0.8377</td>
<td>0.8368</td>
<td>0.8466</td>
<td>0.877</td>
</tr>
<tr>
<td>Negative Opinion</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Precision for</td>
<td>0.8773</td>
<td>0.8874</td>
<td>0.8884</td>
<td>0.8924</td>
<td>0.8894</td>
<td>0.9079</td>
</tr>
<tr>
<td>Positive Opinion</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Recall for</td>
<td>0.8541</td>
<td>0.8649</td>
<td>0.8649</td>
<td>0.8689</td>
<td>0.8649</td>
<td>0.8865</td>
</tr>
<tr>
<td>Negative Opinion</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Recall for</td>
<td>0.8391</td>
<td>0.8565</td>
<td>0.8652</td>
<td>0.8652</td>
<td>0.8739</td>
<td>0.9</td>
</tr>
<tr>
<td>Positive Opinion</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F measure for</td>
<td>0.8578</td>
<td>0.8717</td>
<td>0.8766</td>
<td>0.8786</td>
<td>0.8816</td>
<td>0.9039</td>
</tr>
<tr>
<td>Negative Opinion</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F measure for</td>
<td>0.8316</td>
<td>0.8466</td>
<td>0.8511</td>
<td>0.8525</td>
<td>0.8557</td>
<td>0.8817</td>
</tr>
<tr>
<td>Positive Opinion</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Figure 4.8 Classification Accuracy achieved by LSA-AdaBoost using Medical Service Dataset

From Figure 4.8 it is seen that the classification accuracy achieved by LSA-AdaBoost is much better than that of LSA-Naïve Bayes and LSA-FLRC. LSA-AdaBoost achieves better classification accuracy by 2.2561% and 0.568% than the LSA-Naïve Bayes and LSA-FLRC classifiers respectively. The classification accuracy achieved by Laplacian-AdaBoost is much better than that of Laplacian-Naïve Bayes and Laplacian-FLRC. Laplacian-AdaBoost achieves better classification accuracy by 3.0895% and 2.7326% than the Laplacian-Naïve Bayes and Laplacian-FLRC classifiers respectively.

Figure 4.9 Precision for Negative Opinion achieved by LSA-AdaBoost using Medical Service Dataset
From Figure 4.9 it is seen that the precision for negative opinion achieved by LSA-AdaBoost is much better than that of LSA-Naïve Bayes and LSA-FLRC. LSA-AdaBoost achieves better precision by 3.3252% and 1.044% than the LSA-Naïve Bayes and LSA-FLRC classifiers respectively. The precision for negative opinion achieved by Laplacian-AdaBoost is much better than that of Laplacian-Naïve Bayes and Laplacian-FLRC. Laplacian-AdaBoost achieves better precision by 4.6913% and 3.5275% than the Laplacian-Naïve Bayes and Laplacian-FLRC classifiers respectively.

![Precision for Positive Opinion](image)

**Figure 4.10** Precision for Positive Opinion achieved by LSA-AdaBoost using Medical Service Dataset

From Figure 4.10 it is seen that the precision for positive opinion achieved by LSA-AdaBoost is much better than that of LSA-Naïve Bayes and LSA-FLRC. LSA-AdaBoost achieves better precision by 1.2573% and 0.1126% than the LSA-Naïve Bayes and LSA-FLRC classifiers respectively. The precision for positive opinion achieved by Laplacian-AdaBoost is much better than that of Laplacian-Naïve Bayes and Laplacian-FLRC. Laplacian-AdaBoost achieves better precision by 1.7219% and 2.0586% than the Laplacian-Naïve Bayes and Laplacian-FLRC classifiers respectively.
From Figure 4.11 it is seen that the recall for negative opinion achieved by LSA-AdaBoost is much better than that of LSA-Naïve Bayes and LSA-FLRC. LSA-AdaBoost achieves better recall by 1.2565% and 0% than the LSA-Naïve Bayes and LSA-FLRC classifiers respectively. The recall for negative opinion achieved by Laplacian-AdaBoost is much better than that of Laplacian-Naïve Bayes and Laplacian-FLRC. Laplacian-AdaBoost achieves better recall by 2.0052% and 2.4666% than the Laplacian-Naïve Bayes and Laplacian-FLRC classifiers respectively.

From Figure 4.12 it is seen that the recall for positive opinion achieved by LSA-AdaBoost is much better than that of LSA-Naïve Bayes and LSA-FLRC. LSA-AdaBoost achieves better recall by 0.83% and 0% than the LSA-Naïve Bayes and LSA-FLRC classifiers respectively. The recall for positive opinion achieved by Laplacian-AdaBoost is much better than that of Laplacian-Naïve Bayes and Laplacian-FLRC. Laplacian-AdaBoost achieves better recall by 0.83% and 0% than the Laplacian-Naïve Bayes and Laplacian-FLRC classifiers respectively.
From Figure 4.12 it is seen that the recall for positive opinion achieved by LSA-AdaBoost is much better than that of LSA-Naïve Bayes and LSA-FLRC. LSA-AdaBoost achieves better recall by 3.0628% and 1.0106% than the LSA-Naïve Bayes and LSA-FLRC classifiers respectively. The recall for positive opinion achieved by Laplacian-AdaBoost is much better than that of Laplacian-Naïve Bayes and Laplacian-FLRC. Laplacian-AdaBoost achieves better recall by 3.9429% and 2.9427% than the Laplacian-Naïve Bayes and Laplacian-FLRC classifiers respectively.

**Figure 4.13  F measure for Negative Opinion achieved by LSA-AdaBoost using Medical Service Dataset**

From Figure 4.13 it is seen that the F measure for negative opinion achieved by LSA-AdaBoost is much better than that of LSA-Naïve Bayes and LSA-FLRC. LSA-AdaBoost achieves better F measure by 2.1679% and 0.5605% than the LSA-Naïve Bayes and LSA-FLRC classifiers respectively. The F measure for negative opinion achieved by Laplacian-AdaBoost is much better than that of Laplacian-Naïve Bayes and Laplacian-FLRC. Laplacian-AdaBoost achieves better F measure by 2.8387% and 2.4979% than the Laplacian-Naïve Bayes and Laplacian-FLRC classifiers respectively.
From Figure 4.14 it is seen that the F measure for positive opinion achieved by LSA-AdaBoost is much better than that of LSA-Naïve Bayes and LSA-FLRC. LSA-AdaBoost achieves better F measure by 2.3177% and 0.5301% than the LSA-Naïve Bayes and LSA-FLRC classifiers respectively. The F measure for positive opinion achieved by Laplacian-AdaBoost is much better than that of Laplacian-Naïve Bayes and Laplacian-FLRC. Laplacian-AdaBoost achieves better F measure by 3.3675% and 2.993% than the Laplacian-Naïve Bayes and Laplacian-FLRC classifiers respectively.

4.5 CONCLUSION

Feature selection is required for OM applications, as they lower data dimensionality by removing irrelevant features. This chapter evaluates a feature selection for OM using Latent Semantic Analysis (LSA) and Laplacian Score (LS) to classify opinions. Experiments were undertaken with Naïve Bayes and AdaBoost classifiers and the results are compared to judge the various feature selection methods. Classification accuracy achieved by LSA-AdaBoost is much better than that of LSA-Naïve Bayes and LSA-FLRC.
when using IMDb dataset. LSA-AdaBoost achieved improved classification accuracy by 1.7638% and 1.0762% compared to LSA-Naïve Bayes and LSA-FLRC classifiers respectively. Classification accuracy by Laplacian-AdaBoost is better than that of Laplacian-Naïve Bayes and Laplacian-FLRC. Laplacian-AdaBoost improved classification accuracy by 3.4317% and 2.9095% compared to Laplacian-Naïve Bayes and Laplacian-FLRC classifiers respectively.