Robust Ensemble Co-Clustering Algorithm (RECCA) For Enzyme Clustering

3Ms.R.Rajeswari and 3Dr. G.GunaSekaran

3PhD Research scholar, St. Peter’s University, Chennai, India.
3Principal, Meenakshi College of Engineering, Chennai, India.

ABSTRACT

This research work aims to propose a mechanism with improved preprocessing technique for enzyme clustering. A robust ensemble mechanism is proposed in this research work initially deals with the enhanced principal component analysis. Then the objective function for the co-clustering ensemble towards application to enzyme clustering is presented. A spectral co-clustering ensemble algorithm is described with constructive mathematical modeling followed by the brief algorithm description. The proposed algorithm is capable enough to perform co-clustering with the objective function as the primary component. Simulation results proved that the proposed mechanism RECCA performs better in terms of accuracy and computation time.

INTRODUCTION

Recently, there has been a common augment in the amount of data publicly obtainable in wide-reaching manner predominantly in the field of Bioinformatics, where massive amounts of data have been collected in the form of DNA sequences, protein sequences and structures, information on biological pathways, etc. This has shown the way to varied and scattered sources of biological data. Protein function prediction, and especially enzyme function prediction is on the go Bioinformatics research arena due to the exponential augment in the number of proteins being discovered. This is due to the sequenced genomes, to the difficulties in experimentally characterizing enzyme function and mechanisms, and to the potential biotechnological use of newly discovered enzyme functions. With the above mentioned aspects, prediction of protein’s function is a firm job typically carried out by labor-intensive experimental work or in a semi-automatic manner by making use of sequence homology. This research dimension is capable enough to profit from clustering techniques, since they permit the creation of groups of similar proteins that can be jointly studied. The style in which biological information is collected in using loads of dissimilar datasets pretenses a research challenge for incorporating clustering algorithms.

As an example, the Protein Data Bank (PDB) is a repository of 3D structural data, has dozens or even hundreds of entries for the same molecule. Inconsistencies and redundancies are probably arise due to the attributes representing a given concept may have different names in different databases. Conflicts between data values will also stay ahead, as diverse sources may have unusual attribute values for the same real-world object, due to different representations, scaling or encoding. In this research work robust ensemble co-clustering is introduced in order to analyze how the integration of various data sources in the form of constraints affects the success of enzyme clustering, which might lead to important information about the functions and structures of the enzymes, as well as functional diversification acquired throughout family evolution and to improve the performance for the same.

The remarkable contributions of this paper are:

✓ The knowledge of whether or not adding information from external sources to the database is able to improve the clustering quality for this application;
✓ The lateral way for the collected information to be transformed into constraint sets for the meticulous biological problem;
✓ To perform co-clustering in order to improve the performance by reducing the computation time and increasing average accuracy value.

Corresponding Author: Ms.R.Rajeswari, PhD Research scholar, St. Peter’s University, Chennai, India.
Literature Review:

Sugato Basu et al., 2004 proposed a probabilistic model for semi-supervised clustering based on Hidden Markov Random Fields (HMRFs) that provides a principled framework for incorporating supervision into prototype-based clustering. Bilenko et al., 2004 proposed new methods for the two approaches as well as presents a new semi-supervised clustering algorithm that integrates both of these techniques in a uniform, principled framework. Klein et al., 2002 modified a constrained clustering algorithm to perform exploratory analysis on gene expression data using prior knowledge presented in the form of constraints. Also authors studied the effectiveness of various constraints sets. Wagstan et al., 2001 demonstrated how the popular k-means clustering algorithm can be profitably modified to make use of this information. In experiments with artificial constraints on six data sets, authors observed some improvements in clustering accuracy.

Erlang Zeng et al., 2007 modified a constrained clustering algorithm to perform exploratory analysis on gene expression data using prior knowledge presented in the form of constraints. Authors have studied the effectiveness of various constraints sets. To address the problem of automatically generating constraints from biological text literature, authors considered cluster-based and method similarity-based method. Shahreen Kasim et al., 2013 presented a novel computational framework called the “multi-stage filtering-Clustering Functional Annotation” (msf-CluFA) for clustering gene expression data. The framework consists of fuzzy c-means clustering, achieving dominant cluster, improving confidence level components. In protein databases there was a substantial number of proteins structurally determined but without function annotation. Understanding the relationship between function and structure can be useful to predict function on a large scale. Marcelo Boareto et al., 2012 have analyzed the similarities in global physicochemical parameters for a set of enzymes which were classified according to the four Enzyme Commission (EC) hierarchical levels. Also by using relevance theory authors have introduced a distance between proteins in the space of physicochemical characteristics.

Due to inspiration by the principle of gene transposon proposed by Barbara McClintock, a new immune computing algorithm for automatic clustering named as Gene Transposon based Clone Selection Algorithm (GTCSA) proposed in this Ruochen Liu et al., 2012. It does not require a prior knowledge of the number of clusters; an improved variant of the clonal selection algorithm used to determine the satisfied number of clusters and the appropriate partitioning of the data set as well. Clara Higuera et al., 2013 proposed an expert system (ES), making the main contribution, to cluster a complex data set of 365 prokaryotic species by 114 metabolic features, information which may be incomplete for some species. Inspired on the human expert reasoning and based on hierarchical clustering strategies, Clara Higuera et al., 2013 proposed ES estimates the optimal number of clusters adequate to divide the dataset and afterwards it starts an iterative process of clustering, based on the Self-organizing Maps (SOM) approach, where it finds relevant clusters at different steps by means of a new validity index inspired on the well-known Davies Bouldin (DB) index.

Rosfuzah Roslan et al., 2010 aimed at enhancing the overlap between computational predictions. Guoren Wang et al., 2010 explored a novel concept of local conserved gene cluster (LC-Cluster). To avoid the exponential growth in subspace search, we further authors have presented two efficient algorithms, namely falconer and e-falconer, to mine the complete set of maximal LC-Clusters from gene expression data sets based on enumeration tree. Thanh-Phuong Nguyen and Tu-Bao Ho., 2012 have presented a novel method to effectively predict disease genes by exploiting, in the semi-supervised learning (SSL) scheme, data regarding both disease genes and disease gene neighbours via protein–protein interaction network. Multiple proteomic and genomic data were integrated from six biological databases, including Universal Protein Resource, Interologous Interaction Database, Reactome, Gene Ontology, Pfam, and InterDom, and a gene expression dataset.

Banerjee et al., 2004 introduce a partitional co-clustering formulation that was driven by the search for a good matrix approximation-every co-clustering was associated with an approximation of the original data matrix and the quality of co-clustering was determined by the approximation error. Dhillon et al., 2003 presented an innovative co-clustering algorithm that monotonically increases the preserved mutual information by intertwining both the row and column clusterings at all stages. Bin Gao et al., 2006 proposed a consistent information theory which generates an effective algorithm to obtain the co-clusters of different types of objects. Inderjit Dhillon et al., 2001 presented the novel idea of modeling the document collection as a bipartite graph between documents and words, using which the simultaneous clustering problem can be posed as a bipartite graph partitioning problem. To solve the partitioning problem, authors used a new spectral co-clustering algorithm that uses the second left and right singular vectors of an appropriately scaled word-document matrix to yield good bipartitionings.

Proposed Work:

The proposed research work initially deals with the enhanced principal component analysis. Then the
objective function for the co-clustering ensemble towards application to enzyme clustering analysis is presented. A spectral co-clustering ensemble algorithm is described with constructive mathematical modeling followed by the brief algorithm description. The proposed algorithm is capable enough to perform co-clustering with the objective function as the primary component.

**Enhanced Principal Component Analysis:**

An enhanced weighted version of PCA (EPICA) is introduced where more importance is given to observations whose values are more important. The higher the absolute expression value the more probable is that the meeting minutes are related to the particular topic. To that end, this enhanced PCA uses a new correlation coefficient that gives higher weights to observations that are considered to be more important. Also, the correlation coefficient is sensitive to the presence of outliers and noise in the data. The ranks of the observations are used. In the meeting dataset ranking the observations for each conversation from 1 (highest rank) to n (lowest rank) is taken. The Pearson’s correlation coefficient of the ranked data is thus obtained using the Spearman’s rank correlation coefficient $r_s$, which is given by the expression

$$r_s = \frac{\sum_{i=1}^{n}(R_i - \bar{R})(Q_i - \bar{Q})}{\sqrt{\sum_{i=1}^{n}(R_i - \bar{R})^2 \sum_{i=1}^{n}(Q_i - \bar{Q})^2}}$$

where $\bar{R}$ and $\bar{Q}$ are the average ranks. However, for computational purposes, a more convenient expression which assumes there are no ties is

$$r_s = 1 - \frac{6 \sum_{i=1}^{n}(R_i - Q_i)^2}{n^3 - n}$$

It is clear from this rewritten form of $r_s$ that the calculation of the distance between two ranks in Spearman’s coefficient is given by

$$D_s^2 = (R_i - Q_i)^2$$

which does not take rank importance into account, because if $(R_i - Q_i)$ is, for instance, $(1, 3)$ or $(n-2, n)$, the contribution is the same. The following alternative distance measure is proposed:

$$WD_i^2 = (R_i - Q_i)^2\left((n - R_i + 1) + (n - Q_i + 1)\right)$$

The first term of this product is $D_s^2$, exactly as in Spearman’s coefficient, and represents the distance between $R_i$ and $Q_i$; the second term is a linear weighting function which represents both the importance of $R_i$ and $Q_i$. Hence the weighted rank measure of correlation is obtained using

$$\tau_w = 1 - \frac{6 \sum_{i=1}^{n}(R_i - Q_i)^2(2n + 2 - R_i - Q_i)}{n^4 + n^3 - n^2 - n}$$

which yields values between -1 and +1. The calculation of the distance between two ranks $R_i$ and $Q_i$ is given by

$$WD_i^2 = (R_i - Q_i)^2(2n + 2 - R_i - Q_i)$$

where the second term of the product is a linear weighting function which represents the importance of $R_i$ and $Q_i$. Hence, the distance measure is

$$W_2D_i^2 = (R_i - Q_i)^2(2n + 2 - R_i - Q_i)^2$$

which reflects more than $WD_i^2$ the higher importance of agreement on top ranks. It is common to define rank correlation coefficients, such as Spearman’s, as a linear function of the distance between the two vectors of ranks. In this research, this corresponds to defining a coefficient of the form

$$W_2D_i^2 = A + B \sum_{i=1}^{n}(R_i - Q_i)^2(2n + 2 - R_i - Q_i)^2$$

where the conversations are such that it takes values between -1 and +1. In order to find $A$ and $B$, we will start by doing a specific data transformation and then compute the Pearson’s coefficient on the transformed data. The transformation obtained is exactly of the form, from where the constants $A$ and $B$ follow. The transformation consists in substituting the value of observation $i$ in the first variable by the value of $R_i = R_i(2n + 2 - R_i)$, where $R_i$ is the rank of that observation. It is clear from above that the computation of the new correlation coefficient is equivalent to do a data transformation to each variable as $R_i = R_i(2n + 2 - R_i)$ and then compute the Pearson’s correlation coefficient. $R_i$ represents the rank of each observation value; usually the smallest value has rank 1, the second smallest rank 2, and so on (Uma and Suguna, 2015).

**Objective Function For Co-Clustering Ensemble:**

Given $t$ partitions, with the $q$th partition $(\mu(q), \nu(q))$ having $k(q)$ row clusters and $c(q)$ column clusters. $T$ is defined as a consensus function $N^{(mn \times nt)} \rightarrow N^{(mn)}$ mapping a set of co-clusterings to an integrated co-clustering:

$$T: \left\{(\mu(q), \nu(q)) | q \in \{1, ..., t\} \right\} \rightarrow \{(\mu, \nu)\}$$

Let the set of partitions $\left\{(\mu(q), \nu(q)) | q \in \{1, ..., t\} \right\}$ be denoted by $\Phi$. If there is no background information about the relative importance of the individual partitions, then a reasonable goal for the consensus solution is to seek a co-clustering that shares the most information with the original co-clusterings.

In order to quantify the statistical information shared between two co-clusterings, mutual information is used as a symmetric measure in our work. Here, the objective function is proposed by adapting the original definition given in [Strehl and Ghosh, 2003] to handle the problem of co-clustering ensemble:

$$\sum_{i=1}^{n} (R_i - Q_i)^2(2n + 2 - R_i - Q_i)^2$$
where \((\mu, v)^{[k\times \text{opt}]}\) is the optimal combined co-clustering and it is the one that has maximal average mutual information with all individual partitions in \(\Phi\) given that the number of consensus row clusters desired is \(k\) and the number of column clusters is \(l\). In detail, the average normalized mutual information \((\text{ANMI})\) between a single co-clustering \((\mu, v)\) and a set of \(t\) co-clusterings can be defined as

\[
\phi^{(\text{ANMI})}(\Phi, \mu, v) = \frac{1}{t} \sum_{q=1}^{t} \phi^{(\text{NMI})}(\mu, v, \mu^{(q)}, v^{(q)})
\]

As mentioned before, mutual information (MI) is a symmetric measure and can be used to quantify the statistical information shared between two distributions. Thus, MI provides a sound indication of the shared information between a pair of co-clusterings. The normalized mutual information (NMI) is defined as

\[
\text{NMI}(X, Y) = \frac{I(X, Y)}{\sqrt{H(X)H(Y)}}
\]

where \(X\) and \(Y\) denote two vectors, \(I(X, Y)\) denotes the mutual information between \(X\) and \(Y\). \(H(X)\) denotes the entropy of \(X\) and \(H(X) = I(X, X)\). Suppose there are two co-clusterings \((X_r, X_c)\) and \((Y_r, Y_c)\), i.e., \((X_r, Y_r),(X_c, Y_c)\) denote the row and column cluster labeling variables respectively. Then, the NMI between two co-clusterings can be defined as

\[
\text{NMI}(X_r, X_c) = \text{NMI}(Y_r, Y_c) = 1, \text{ as desired.}
\]

According to Eqs. (11), (9) can be further rewritten as

\[
\phi^{(\text{ANMI})}(\Phi, \mu, v) = \frac{1}{t} \sum_{q=1}^{t} \phi^{(\text{NMI})}(\mu, v, \mu^{(q)}, v^{(q)})
\]

Eq. (12) needs to be estimated by the sampled quantities provided by the co-clusterings. Then, the normalized mutual information estimate \(\hat{\phi}^{(\text{NMI})}\) can be defined as

\[
\hat{\phi}^{(\text{NMI})}(\mu^{i}, v^{i}, \mu^{j}, v^{j}) = \phi^{(\text{NMI})}(\mu^{i}, \mu^{j}) + \phi^{(\text{NMI})}(v^{i}, v^{j})
\]

\[
= \frac{\sum_{a=1}^{k(i)} \sum_{b=1}^{k(j)} O_{a,b} \log \left( \frac{O_{a,b}}{O_i} \right)}{\sqrt{\sum_{a=1}^{k(i)} O_{a} \log \left( \frac{O_{a}}{O_i} \right)} \sqrt{\sum_{b=1}^{k(j)} O_{b} \log \left( \frac{O_{b}}{O_i} \right)}} + \frac{\sum_{a=1}^{k(i)} \sum_{b=1}^{k(j)} O_{a,b} \log \left( \frac{O_{a,b}}{O_j} \right)}{\sqrt{\sum_{a=1}^{k(i)} O_{a} \log \left( \frac{O_{a}}{O_j} \right)} \sqrt{\sum_{b=1}^{k(j)} O_{b} \log \left( \frac{O_{b}}{O_j} \right)}}
\]

where \(|O|\) and \(|F|\) denote the number of objects and features in a co-cluster respectively. \((O_{a}, F_{a})\) denotes the number of objects and features in co-cluster \(CO_{a}\) according to \((\mu^{i}, v^{i})\) and \((O_{a}, F_{a})\) denotes the number of objects and features in co-cluster \(CO_{b}\) according to \((\mu^{j}, v^{j})\).

**Spectral Co-Clustering Ensemble Algorithm For Enzyme Clustering:**

In this work, the final ensemble step can be formulated as a partition problem on a bipartite graph. For convenience of discussion, we use small-bold letters such as \(u, v\) as vectors. Capital-bold letters such as \(M, E, L\) will denote matrices, and capital letters such as \(V, R\) will denote vertex sets.
Denote the bipartite graph \( G = (V_r, V_c, E) \) containing two sets of vertices including row labeling vertices \( V_r \) and column labeling vertices \( V_c \) respectively. It is easy to verify that the adjacency matrix \( M \) of the bipartite graph can be written as

\[
M = \begin{bmatrix}
0 & E_r \\
E_c & 0
\end{bmatrix}
\]

(14)

where

\[
E = \begin{bmatrix}
C_{rr} & C_{rc} \\
C_{cr} & C_{cc}
\end{bmatrix}
\]

(15)

\( C_{rr} \) denotes the edge-weights between row labeling vertices that are both in \( V_r \). \( C_{cc} \) denotes the edge-weights between labeling vertices with one in \( V_r \) and the other in \( V_c \). \( C_{rc} \) and \( C_{cr} \) are defined similarly. Let \(|E|_{ij}\) denote the \((i,j)\)th element of \( E \), \(|E|_{12} \) is the edge weight between two vertices and can be obtained according to Eq.(13). More specifically,

\[
|E|_{ij} = \frac{\Sigma_{a=1}^{k_i} \Sigma_{b=2}^{k_j} \alpha_{a,\beta} \log \left( \frac{|D|_{a,b} \alpha_{a,\beta}}{|F|_{a,b} \alpha_{a,\beta}} \right)}{\sqrt{\left( \Sigma_{a=1}^{k_i} \alpha_{a,\beta} \log \left( \frac{|D|_{a,b}}{|F|_{a,b}} \right) \right)^2}}
\]

(16)

if the \( i \)th and \( j \)th vertices are both the row labeling vertices for enzyme clusters;

\[
|E|_{ij} = \frac{\Sigma_{a=1}^{k_i} \Sigma_{b=1}^{k_j} \alpha_{a,\beta} \log \left( \frac{|F|_{a,b} \alpha_{a,\beta}}{|D|_{a,b} \alpha_{a,\beta}} \right)}{\sqrt{\left( \Sigma_{a=1}^{k_i} \alpha_{a,\beta} \log \left( \frac{|D|_{a,b}}{|F|_{a,b}} \right) \right)^2}}
\]

(17)

if the \( i \)th and \( j \)th vertices are both the column labeling vertices for enzyme clusters. Otherwise \(|E|_{ij} = 0 \).

According to the bipartite graph \( G = (V_r, V_c, E) \) given above, now we define the co-clustering partition matrix \( Y \) as

\[
Y = \begin{bmatrix}
Y_r \\
Y_c
\end{bmatrix}
\]

(18)

where \( Y_r \) is the partition on row labeling vertex set \( V_r \) and \( Y_c \) is the partition on column labeling vertex set \( V_c \). Thus, the laplacian matrix \( L \) can be defined as

\[
L = D - M
\]

(19)

where

\[
D = \begin{bmatrix}
D_r & 0 \\
0 & D_c
\end{bmatrix}
\]

(20)

\( D_r \) and \( D_c \) are diagonal matrices such that \(|D_r|_{ii} = \Sigma_{j} E_{ij} \), \(|D_c|_{jj} = \Sigma_{i} E_{ij} \). Note that the key step is to find the minimum cut vertex partitions on the bipartite graph. The normalized-cut objective function can be expressed as

\[
\min_Y \left( \text{tr}(Y^TLY) \right)
\]

(21)

One way to solve the partition problem of the bipartite graph is to compute the left and right eigenvectors of the matrix \( A \) defined as

\[
A = D^{-1/2} E D^{-1/2}
\]

(22)

After the left and right eigenvectors of matrix \( A \) are obtained, the left and right eigenvectors of the second to the \((\omega + 1)\)th eigenvalues are selected as

\[
U = [u_1, u_2, ..., u_{\omega+1}]
\]

and

\[
V = [v_1, v_2, ..., v_{\omega+1}]
\]

respectively. Here, the \( \omega = \log_{2k} \) singular vectors \( u_1, u_2, ..., u_{\omega+1} \) and \( v_1, v_2, ..., v_{\omega+1} \) often contain k-modal information about the original co-clustering labeling. Thus, the k-dimensional data matrix can be written as

\[
X = \begin{bmatrix}
D_r^{-1/2} U \\
D_c^{-1/2} V
\end{bmatrix}
\]

(23)

At last, the classical k-means algorithm is preformed on \( X \), and the final consensus co-clustering result is obtained.

**Algorithm description:**

According to the above inference, we design an algorithm based on spectral method for enzyme co-clustering ensemble. The algorithm procedure is described step by step as follows.

**Algorithm (Spectral Co-Clustering Ensemble)**

**Input:**

- **Original data matrix** \( X_{nm} \), **num. of row clusters** \( k \), **num. of column clusters** \( \ell \) (i.e., \( K \times \ell \) clusters in total)

01. Divide \( X_{nm} \) into \( k \) row clusters and \( \ell \) column clusters by the co-clustering algorithms and the base co-clustering labeling are obtained.
02. Compute pairwise similarities of the base co-clustering labelling according to Eqs. (10) and (11). Construct the adjacency matrices \( M \).
03. Construct the diagonal matrices \( D_r, D_c \) where \(|D_r|_{ii} = \Sigma_{j} E_{ij} \) and \(|D_c|_{jj} = \Sigma_{i} E_{ij} \)
04. Calculate \( A \) as defined in Eq. (16).
05. Perform singular value decomposition (SVD) on matrix A. Compute $\omega = \log_2 k$ singular vectors of $A$, $u_2, \ldots, u_{\omega+1}$ and $v_2, \ldots, v_{\omega+1}$. Denote the left and right eigenvectors of the 2nd to the $(\omega + 1)\text{th}$ eigenvalues as U and V respectively.

06. Construct $X_r = D_r^{-1/2}U$ and $X_c = D_c^{-1/2}V$.

07. Run k-means algorithm on the x-dimensional data $X_r$ to get the row labelings partition matrix $Y_r$. Similarly get $Y_c$ from $X_c$.

Output:
The final consensus co-clustering result.

It can be observed that the main computational cost is to perform SVD on the matrix A on Step 5. Consider Lanczos algorithm to compute the eigenvectors [Shi et al., 2010]. The complexity of our algorithm is $O(eN(|m| + |n|)^2)$, where $e$ is the number of eigenvectors desired, $N$ is the number of Lanczos iteration steps and $(|m| + |n|)^2$ is the upper bound of the nonzero entries of matrix M. More performance in detail are recorded in the next section.

About The Dataset:

Table 1: Data Sets for Text Pairwise (Document-Word) Coclustering

<table>
<thead>
<tr>
<th>Name</th>
<th>Datasets</th>
<th>Data Structure</th>
<th>No. of clusters</th>
<th>No. of documents</th>
</tr>
</thead>
<tbody>
<tr>
<td>CT1</td>
<td>oh15</td>
<td>Adenosine-Diphosphate, Blood-Vessels</td>
<td>2</td>
<td>154</td>
</tr>
<tr>
<td>CT2</td>
<td>oh15</td>
<td>Aluminium, Blood-Coagulation-Factors</td>
<td>2</td>
<td>122</td>
</tr>
<tr>
<td>CT3</td>
<td>re0</td>
<td>Interest, reserves</td>
<td>2</td>
<td>261</td>
</tr>
<tr>
<td>CT4</td>
<td>re0</td>
<td>housing, jobs</td>
<td>2</td>
<td>55</td>
</tr>
<tr>
<td>CT5</td>
<td>re0</td>
<td>housing, interest, jobs</td>
<td>3</td>
<td>274</td>
</tr>
<tr>
<td>CT6</td>
<td>oh15</td>
<td>Aluminium, Blood-Vessels, Leucine</td>
<td>3</td>
<td>207</td>
</tr>
<tr>
<td>CT7</td>
<td>re0</td>
<td>cpi, housing, ipi, lei, retail</td>
<td>5</td>
<td>144</td>
</tr>
<tr>
<td>CT8</td>
<td>re0</td>
<td>bop, cpi, gnp, housing, interest, ipi, jobs, lei, money</td>
<td>10</td>
<td>1150</td>
</tr>
</tbody>
</table>

Table 2: Data Sets for Text High-Order (Word-Document-Category) Coclustering

<table>
<thead>
<tr>
<th>Name</th>
<th>Datasets</th>
<th>Data Structure</th>
<th>No. of clusters</th>
<th>No. of documents</th>
</tr>
</thead>
<tbody>
<tr>
<td>HT1</td>
<td>oh15, re0</td>
<td>{Adenosine-Diphosphate, Aluminium, Cell-Movement}, [cpi,money]</td>
<td>2</td>
<td>899</td>
</tr>
<tr>
<td>HT2</td>
<td>oh15, re0</td>
<td>{Blood-Coagulation-Factors, Enzyme-Activation, Staphylococcal-Infections}, [jobs,reserves]</td>
<td>2</td>
<td>461</td>
</tr>
<tr>
<td>HT3</td>
<td>oh15, re0</td>
<td>{Aluminium, Blood-Coagulation-Factors, Blood-Vessels}, [housing,retail]</td>
<td>2</td>
<td>256</td>
</tr>
<tr>
<td>HT4</td>
<td>oh15, re0</td>
<td>[Aluminium, Cell-Movement, Staphylococcal-Infections], [cpi,jobs]</td>
<td>2</td>
<td>391</td>
</tr>
<tr>
<td>HT5</td>
<td>WAP, re0</td>
<td>[media, film, music], [cpi, jobs]</td>
<td>2</td>
<td>404</td>
</tr>
<tr>
<td>HT6</td>
<td>Newsgroup</td>
<td>{rec.por.baseball, rec.sport.hockey}, [talk.politics.guns, talk.politics.mideast,talk.politics.misc]</td>
<td>3</td>
<td>300</td>
</tr>
<tr>
<td>HT7</td>
<td>Newsgroup</td>
<td>{comp.graphics, comp.os.ms-windows.misc}, [rec.autos,rec.motorcycles], [sciencrypt,sci.electronics]</td>
<td>2</td>
<td>3932</td>
</tr>
<tr>
<td>HT8</td>
<td>Newsgroup</td>
<td>{comp.graphics, comp.os.ms-windows.misc}, [sci.electronics,sci.med]</td>
<td>3</td>
<td>5942</td>
</tr>
</tbody>
</table>

Table 3: Data Sets for Gene Expression Pairwise (Condition-Gene) Coclustering

<table>
<thead>
<tr>
<th>Name</th>
<th>Datasets</th>
<th>Data Structure</th>
<th>No. of clusters</th>
<th>No. of documents</th>
</tr>
</thead>
<tbody>
<tr>
<td>BT1</td>
<td>ALL/AML</td>
<td>ALL, AML</td>
<td>2</td>
<td>72</td>
</tr>
<tr>
<td>BT2</td>
<td>Breast Cancer</td>
<td>Relapse, Non-relapse</td>
<td>2</td>
<td>97</td>
</tr>
<tr>
<td>BT3</td>
<td>Central Nervous</td>
<td>Class1, Class2</td>
<td>2</td>
<td>60</td>
</tr>
<tr>
<td>BT4</td>
<td>Colon Tumor</td>
<td>Positive, Negative</td>
<td>2</td>
<td>62</td>
</tr>
<tr>
<td>BT5</td>
<td>Lung Cancer</td>
<td>MPM, ADCA</td>
<td>2</td>
<td>181</td>
</tr>
<tr>
<td>BT6</td>
<td>Ovarian Cancer</td>
<td>Cancer, Normal</td>
<td>2</td>
<td>253</td>
</tr>
<tr>
<td>BT7</td>
<td>ALL/MLL/AML</td>
<td>ALL, MLL,AML</td>
<td>3</td>
<td>72</td>
</tr>
</tbody>
</table>

Several datasets have been taken for the performance analysis. The datasets for text pairwise co-clustering is shown in Table 1. The datasets for Text High-Order (Word-Document-Category) co-clustering is presented in Table 2. The datasets for gene expression pairwise (Condition-Gene) co-clustering is given in Table 3. The datasets for Image High-Order (Color-Image-Texture) co-clustering is depicted in Table 4.
RESULTS AND DISCUSSIONS

Performance of RECCA is made a comparison with Semisupervised Non-negative Matrix Factorization (SS-NMF) (Yanhua Chen et al., 2010), Non-negative Matrix Factorization (NMF) (Xu et al., 2003), Combinatorial Markov Random Field (CMRF) (Bekkerman and Jeon, 2007), Semisupervised Combinatorial Markov Random Field (SS-CMRF) (Bekkerman and Sahami, 2006), Spectral Relational Clustering (SRC) (Long et al., 2006) and Transductive Support Vector Machines (TSVM) (Joachims, 1999) in terms of accuracy and computation time. Figure 1 uses the Text Pairwise (Document-Word) Coclustering datasets depicted in Table 1. Figure 2 uses the Gene Expression Pairwise (Condition-Gene) Coclustering datasets depicted in Table 3. Figure 3 uses the Text High-Order (Word-Document-Category) Coclustering datasets depicted in Table 2. Figure 4 uses the Image High-Order (Color-Image-Texture) Coclustering datasets depicted in Table 4.

The experiments are performed on a Windows 8.1 machine with Intel Core i3 processors and 4 GB DDR III RAM. The experiments on algorithms are evaluated using MATLAB R2012a.

Table 4: Data Sets for Image High-Order (Color-Image-Texture) Coclustering

<table>
<thead>
<tr>
<th>Name</th>
<th>Datasets</th>
<th>No. of Modalities</th>
<th>No. of clusters</th>
<th>No. of documents</th>
</tr>
</thead>
<tbody>
<tr>
<td>IT1</td>
<td>eggs, decoys</td>
<td>3</td>
<td>2</td>
<td>200</td>
</tr>
<tr>
<td>IT2</td>
<td>dawn, foliage</td>
<td>3</td>
<td>2</td>
<td>200</td>
</tr>
<tr>
<td>IT3</td>
<td>decoys, dawn</td>
<td>3</td>
<td>2</td>
<td>200</td>
</tr>
<tr>
<td>IT4</td>
<td>decoys, firearms, cards, buses</td>
<td>3</td>
<td>4</td>
<td>400</td>
</tr>
<tr>
<td>IT5</td>
<td>abstract, dawn, foliage, waves</td>
<td>3</td>
<td>4</td>
<td>400</td>
</tr>
<tr>
<td>IT6</td>
<td>eggs, decoys, dawn, foliage</td>
<td>3</td>
<td>4</td>
<td>400</td>
</tr>
<tr>
<td>IT7</td>
<td>eggs, decoys, buses, abstract, texture, dawn</td>
<td>3</td>
<td>6</td>
<td>600</td>
</tr>
</tbody>
</table>

Table 5: Comparison of Average Accuracy for Text Data

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>Percentage of Constrained Pairs</th>
<th>TSVM</th>
<th>SS-KK</th>
<th>SS-CMRF</th>
<th>SS-NMF</th>
<th>RECCA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0%</td>
<td>0.52</td>
<td>0.56</td>
<td>0.63</td>
<td>0.66</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.5%</td>
<td>0.57</td>
<td>0.58</td>
<td>0.76</td>
<td>0.79</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1%</td>
<td>0.58</td>
<td>0.62</td>
<td>0.78</td>
<td>0.81</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3%</td>
<td>0.61</td>
<td>0.66</td>
<td>0.82</td>
<td>0.84</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5%</td>
<td>0.64</td>
<td>0.7</td>
<td>0.87</td>
<td>0.89</td>
<td></td>
</tr>
<tr>
<td></td>
<td>10%</td>
<td>0.69</td>
<td>0.77</td>
<td>0.89</td>
<td>0.91</td>
<td></td>
</tr>
</tbody>
</table>

Fig. 1: Comparison of Average Accuracy for Text Data

Figure 1 shows the performance evaluation of average accuracy for text data. It is evident that the proposed RECCA mechanism using Enhanced PCA outperforms other mechanisms in terms of document clustering performance with least prior knowledge. The performance values are depicted in Table 5.
Fig. 2: Comparison of Average Accuracy for Gene Expression Data

Figure 2 presents the performance evaluation of average accuracy for gene expression data. It is most visible that the proposed RECCA mechanism using Enhanced PCA outperforms other mechanisms in terms of increasing percentage of pairwise constraints for semisupervised condition coclustering. The performance values are depicted in Table 6.

<table>
<thead>
<tr>
<th>Algorithms Percentage of Constrained Pairs</th>
<th>TSVM</th>
<th>SS-KK</th>
<th>SS-CMRF</th>
<th>SS-NMF</th>
<th>RECCA</th>
</tr>
</thead>
<tbody>
<tr>
<td>0%</td>
<td>0</td>
<td>0.54</td>
<td>0.66</td>
<td>0.59</td>
<td>0.61</td>
</tr>
<tr>
<td>0.5%</td>
<td>0.48</td>
<td>0.57</td>
<td>0.69</td>
<td>0.78</td>
<td>0.8</td>
</tr>
<tr>
<td>1%</td>
<td>0.54</td>
<td>0.59</td>
<td>0.73</td>
<td>0.8</td>
<td>0.83</td>
</tr>
<tr>
<td>3%</td>
<td>0.58</td>
<td>0.62</td>
<td>0.76</td>
<td>0.83</td>
<td>0.87</td>
</tr>
<tr>
<td>5%</td>
<td>0.62</td>
<td>0.67</td>
<td>0.79</td>
<td>0.86</td>
<td>0.89</td>
</tr>
<tr>
<td>10%</td>
<td>0.67</td>
<td>0.71</td>
<td>0.82</td>
<td>0.88</td>
<td>0.92</td>
</tr>
</tbody>
</table>

Fig. 3: Comparison of Average Accuracy – Text High Order Coclustering
Figure 3 presents the performance comparison of average accuracy for text high order co-clustering. It is most obvious that the proposed RECCA mechanism using Enhanced PCA outperforms other mechanisms. The performance values are depicted in Table 7.

Table 7: Comparison of Average Accuracy – Text High Order Co-clustering

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>Percentage of Constrained Pairs</th>
<th>SS-CMRF</th>
<th>SS-NMF</th>
<th>RECCA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0%</td>
<td>0.48</td>
<td>0.52</td>
<td>0.54</td>
</tr>
<tr>
<td></td>
<td>1%</td>
<td>0.51</td>
<td>0.56</td>
<td>0.59</td>
</tr>
<tr>
<td></td>
<td>3%</td>
<td>0.54</td>
<td>0.59</td>
<td>0.62</td>
</tr>
<tr>
<td></td>
<td>8%</td>
<td>0.57</td>
<td>0.62</td>
<td>0.65</td>
</tr>
<tr>
<td></td>
<td>13%</td>
<td>0.6</td>
<td>0.67</td>
<td>0.69</td>
</tr>
<tr>
<td></td>
<td>15%</td>
<td>0.64</td>
<td>0.72</td>
<td>0.76</td>
</tr>
</tbody>
</table>

Figure 4 presents the performance comparison of average accuracy for image high order co-clustering. It can be perceived that the proposed RECCA mechanism using Enhanced PCA outperforms other mechanisms. The performance values are depicted in Table 8.

Table 8: Comparison of Average Accuracy – Image High Order Co-clustering

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>Percentage of Constrained Pairs</th>
<th>SS-CMRF</th>
<th>SS-NMF</th>
<th>RECCA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0%</td>
<td>0.67</td>
<td>0.68</td>
<td>0.7</td>
</tr>
<tr>
<td></td>
<td>1%</td>
<td>0.69</td>
<td>0.69</td>
<td>0.72</td>
</tr>
<tr>
<td></td>
<td>3%</td>
<td>0.72</td>
<td>0.72</td>
<td>0.75</td>
</tr>
<tr>
<td></td>
<td>8%</td>
<td>0.76</td>
<td>0.77</td>
<td>0.79</td>
</tr>
<tr>
<td></td>
<td>13%</td>
<td>0.78</td>
<td>0.79</td>
<td>0.82</td>
</tr>
<tr>
<td></td>
<td>15%</td>
<td>0.81</td>
<td>0.82</td>
<td>0.85</td>
</tr>
</tbody>
</table>
Figure 5 presents the performance of computational time (number of samples in the central data type - Nc) and the results proved that the proposed RECCA mechanism using Enhanced PCA approach delivers significant better performance over other methods. The performance values are depicted in Table 9.

Table 9: Comparison of Computational Speed - In Log(Seconds) For Increasing Nc

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>Percentage of Constrained Pairs</th>
<th>NMF</th>
<th>SS-NMF</th>
<th>RECCA</th>
<th>CMRF</th>
<th>SS-CMRF</th>
<th>SRC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1000</td>
<td>0.05</td>
<td>0.23</td>
<td>0.21</td>
<td>3</td>
<td>6</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>1500</td>
<td>0.2</td>
<td>0.45</td>
<td>0.38</td>
<td>9</td>
<td>48</td>
<td>62</td>
</tr>
<tr>
<td></td>
<td>2000</td>
<td>0.1</td>
<td>0.36</td>
<td>0.52</td>
<td>34</td>
<td>69</td>
<td>89</td>
</tr>
<tr>
<td></td>
<td>2500</td>
<td>0.4</td>
<td>0.62</td>
<td>0.57</td>
<td>52</td>
<td>82</td>
<td>172</td>
</tr>
<tr>
<td></td>
<td>3000</td>
<td>0.52</td>
<td>0.84</td>
<td>0.74</td>
<td>92</td>
<td>107</td>
<td>352</td>
</tr>
</tbody>
</table>

Fig. 6: Comparison of Computational Speed - In Log(Seconds) For Increasing Np
Figure 6 presents the performance of computational time (the maximum feature dimension for all feature modalities - N_p) and the results proved that the proposed RECCA mechanism using Enhanced PCA approach delivers significant better performance over other methods. The performance values are depicted in Table 10.

Table 10: Comparison of Computational Speed - In Log(Seconds) For Increasing N_v

<table>
<thead>
<tr>
<th>Algorithms Percentage of Constrained Pairs</th>
<th>NMF</th>
<th>CMRF</th>
<th>SS-CMRF</th>
<th>SS-NMF</th>
<th>SRC</th>
<th>RECCA</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>3.8</td>
<td>4.2</td>
</tr>
<tr>
<td>200</td>
<td>0.2</td>
<td>0.9</td>
<td>1.1</td>
<td>1.2</td>
<td>3.9</td>
<td>4.3</td>
</tr>
<tr>
<td>300</td>
<td>0.2</td>
<td>1.2</td>
<td>1.22</td>
<td>1.3</td>
<td>4.2</td>
<td>4.6</td>
</tr>
<tr>
<td>400</td>
<td>0.1</td>
<td>1.5</td>
<td>1.9</td>
<td>2.1</td>
<td>4.7</td>
<td>4.9</td>
</tr>
<tr>
<td>500</td>
<td>0.2</td>
<td>1.8</td>
<td>2.2</td>
<td>2.9</td>
<td>5</td>
<td>5.2</td>
</tr>
<tr>
<td>600</td>
<td>0.3</td>
<td>2.5</td>
<td>2.8</td>
<td>3.8</td>
<td>6.3</td>
<td>6.6</td>
</tr>
<tr>
<td>700</td>
<td>0.4</td>
<td>3.3</td>
<td>3.7</td>
<td>4</td>
<td>8.9</td>
<td>9.2</td>
</tr>
<tr>
<td>800</td>
<td>0.2</td>
<td>4.5</td>
<td>4.8</td>
<td>5.6</td>
<td>13.6</td>
<td>13.8</td>
</tr>
<tr>
<td>900</td>
<td>0.4</td>
<td>6.8</td>
<td>7</td>
<td>8.1</td>
<td>17.9</td>
<td>18.2</td>
</tr>
<tr>
<td>1000</td>
<td>0.2</td>
<td>7.9</td>
<td>9</td>
<td>10.9</td>
<td>20</td>
<td>20.8</td>
</tr>
</tbody>
</table>

Conclusion:

This paper presented a mechanism with improved preprocessing technique for enzyme clustering. Initially the proposed work RECCA deals with the enhanced principal component analysis for preprocessing. The objective function for the co-clustering ensemble towards application to enzyme clustering is presented and also described. The objective function plays a major role which can perform co-clustering. Simulation results show that the proposed mechanism RECCA performs better in terms of accuracy and computation time. Regarding the future direction of this work, RECCA can be hybrid with optimization techniques for the much better performance of accuracy and computation time.

REFERENCES


prediction”, Computational Biology and Chemistry, 40: 15-19, ISSN 1476-9271.


Yanhua Chen, Lijun Wang, Ming Dong, Member, 2010. "Non-Negative Matrix Factorization for Semisupervised Heterogeneous Data Co-clustering”, Ieee Transactions On Knowledge And Data Engineering, 22.