Presented at the Research Congress 2013
De La Salle University Manila
March 7-9, 2013

MEASURING THE EFFECTIVENESS OF RANDOM PROJECTION METHOD FOR DIMENSIONALITY REDUCTION OF TEXT DOCUMENTS

Arnulfo Azcarraga[1], Maynard Landrito[1]
[1]De La Salle University

Abstract: Text documents usually contain a high number of dimensions when represented as data for processing and analysis, making it time-consuming to perform experiments on them. To compress text documents and reduce dimensionality while attempting to preserve the structure and properties of the data, Random Projection Method (RPM) can be used. The structure and properties of the data can be visualized using Self-Organizing Maps (SOM), an artificial neural network that maps high-dimensional data to a grid of nodes and serve as visualization. In this project, we compress a high-dimensional collection of text documents using RPM and visualize its structure using SOM. Various quality measures are also used to quantify the quality of the compression. The experiments show that RPM is effective in compressing the data, but the quality deteriorates rapidly as more compression is applied. Also, it is also shown that the structure of the data is somehow retained even at different levels of compression.

Key Words: Random Projection; Self-Organizing Maps; Text Document Processing

1. INTRODUCTION

Text documents are among the most common types of data and medium of information in this digital age. The number of text documents grows exponentially in volume, especially in the ever-growing use of the Internet. The high volume of data presents a real problem in text document processing and this only gets worse as more and more text documents become available on the web.

The bag-of-words approach to data representation usually results in very high-dimensional data because of the huge number of unique words present in a large collection of text documents. To alleviate the high-dimensionality problem, one approach is to reduce the number of dimensions through a technique called *Random Projection Method* (RPM), as used in (Azcarraga, Yap, Tan, and Chua, 2004; Dy, 2011). With RPM, a dataset with thousands of dimensions can be compressed into a mere hundred, while generally preserving the structure and relative distances of the data points.

One method to visualize data and its underlying structure is by using *Self-Organizing Maps* (SOM). The SOM is an artificial neural network model that can be used to visualize high-dimensional data (Chifu and Letia, 2010). Since SOM preserves the structure of the data in the input environment, this SOM feature will be used to visualize the effectiveness of RPM at different compression levels. After training the SOM, it is labeled based on the dataset used to train it and various quality measures are calculated to quantify the effectiveness of the RPM, along with visual analysis of the SOM.

The rest of the paper is organized as follows: section 2 discusses the dataset used for this project; section 3 discusses RPM; section 4 discusses SOM; section 5 discusses the different quality measures used; section 6 discusses the results of the experiments; and section 7 presents the conclusions and recommendations.

2. DATASET

The dataset used in this project is the Reuters Collection Volume 1 – version 2, referred to as RCV1-v2. The dataset is an archive of over 800,000 manually categorized newswire stories from Reuters, Ltd. The original dataset has been preprocessed, prepared and improved by (Lewis, Yang, Rose, Li, 2004). The dataset has been split into two files – one for training and one for testing. For this project, only the training set is used for both training and labeling of the SOM. The test set consists of 23,149 documents with 47,152 unique words in the entire test set corpora. The news documents appeared within the period August 20 - 31, 1996.

The RCV1-v2 has a hierarchical category system; however, this project will only use the top-level categories. There are 4 top-level categories, which are Economics (ECAT), Corporate/Industrial (CCAT), Government/Social (GCAT) and Markets (MCAT). The ECAT category comprises of 3,449 instances (14.47%); CCAT with 9,949 instances (42.98%), GCAT with 5,004 instances (21.62%); and MCAT with 4,747 instances (20.51%). The RCV1-v2 underwent preprocessing before being released, including stemming and stop words removal. Each document in the dataset is represented as a collection of stemmed words. From here, the term count is computed per document. The final representation of each item in the dataset before undergoing the Random Projection Method is then a vector of term counts for each term.

3. RANDOM PROJECTION METHOD

The Random Projection Method (RPM) is an algorithm to compress vectors with ***n*** dimensions into ***m*** dimension. Aside from ***m***, another parameter ***r*** is supplied to determine the number of random placement of each feature. This project used the RPM algorithm used in (Dy, 2011).

After the algorithm has been applied, the RPM dataset will have ***m*** dimensions. The algorithm merges multiple dimensions from the original ***n*** dimensions to compress the dataset into ***m*** dimensions.

After applying RPM, the term frequency is used as values of the new dataset, which is calculated as the ***dimension value*** / ***total of all dimensions***. The values are then normalized to fit in the range of 0 to 1, with the mean value set to 0.5. For this project, three different RPM parameter sets have been used to generate 3 different datasets from the preprocessed RCV1-v2 dataset. Table 1 shows the values of ***r*** and ***m*** for each encoded dataset.

Table 1. Different parameters used for RPM.

Encoded Dataset	***m*** value	***r*** value
Dataset 200-20	200	20
Dataset 100-10	100	10
Dataset 50-5	50	5

4. SELF-ORGANIZING MAP

The self-organizing map (SOM) is a neural network model that produces a 2D visualization or map of the dataset. This is done by starting on a map of size ***n*** x ***n*** with each node in the map having

starting values (usually random) for each dimension or feature. The map is then updated gradually by shifting the values of nodes closer to each input item. This is repeated continuously until a predefined number of iterations. The algorithm for the SOM is explained in more detail in (Chifu and Letia, 2010).

The final result of the SOM is a map of nodes whose relative values reflect that of the dataset. When the dataset items are mapped on the nodes of the SOM, those that are close to each other should also fall on topologically close nodes.

The similarity metric used in this project is the cosine similarity measure, as opposed to the Euclidean distance measure, because cosine is more suited and is more popularly used as similarity measure for text documents (Huang, 2008).

For this project, the SOM size ***n*** x ***n*** is 20x20 nodes, to balance between having more nodes for finer visualization, and the time needed to train the SOM. As mentioned earlier, the initial neighborhood size is 20, and the final neighborhood size is 1. The initial learning rate is 0.5 and the final learning rate is 0.01. The initial learning rate was chosen to force bigger adjustments in the node values during the beginning of training, and the final learning rate was chosen to have very small and fine adjustments towards the end of the training, since difference between feature values can be very small.

The number of training iterations is a multiple of the number of instances in the dataset, to ensure that all instances are used for training an equal number of times. The total number of iterations for training is 23,149 × 30, or 694,470. This is divided into phase 1 and phase 2. Phase 1 consists of 23,149 * 20, or 462,980 iterations while phase 2 consists of 23,149 * 10, or 231,490 iterations.

After the SOM has been trained, the nodes have to be labeled based on the dataset items. This allows for the individual dataset items to be topologically placed on the SOM, which in turn allow for the computation of the quality measures and visual inspection of the SOM and the structure of the dataset. There are two ways to label SOM nodes. The first is to assign each dataset item to the node to which it is most similar to, according to some similarity metric. This process will be called from this point on as *uploading*. The second approach is to iterate through all the SOM nodes and find the top ***n*** most similar dataset items. These top ***n*** items will be assigned to that particular node and serve as its labeling. This approach will be called from this point on as *downloading*. This project used 20 for the value of ***n***.

5. QUALITY MEASURES

Aside from the visual inspection and analysis of the SOM based on the uploading and downloading of dataset items, various quality measures have been used in this project to quantitatively measure the structure of the dataset and the quality of the labeling of the SOM Nodes.

There five different quality measures used in this project. These are intra-node purity, intra-node similarity, inter-node vector centroid, inter-node position centroid and intra-cluster purity, and each is discussed below. Both uploaded and download dataset items will be used for calculation, except for the intra-node similarity which has upload only, making a total of 9 different measures. All similarity measures will use the cosine similarity metric, except for inter-node position which uses the Euclidean distance metric.

Intra-node purity - The intra-node purity is the ratio of the number of the majority of the assigned dataset items to the total number of assigned items in a node. A high intra-node purity shows that dataset items from the same class are more likely to be assigned on the same node.

Intra-node similarity - The intra-node similarity is the average similarity between a node and its assigned dataset items. A high intra-node similarity shows that dataset items are more tightly packed in a node, and that they are more similar to each other in general.

Inter-node Feature Centroid - The inter-node feature centroid is the average similarity of all nodes that have an assigned dataset item of a certain class against their centroid. The centroid is just the vector of the average values of all dimensions of each node in the set. The average value for all classes will then be used as the final value.

The inter-node feature centroid shows that relative similarity of dataset items (via the nodes they are assigned to) of the same class in the whole map. A high value means that nodes (and indirectly, dataset items) are more similar to each other.

Inter-node Position Centroid - Inter-node position is the average Euclidean distance of all nodes that have an assigned dataset item of a certain class against their centroid. It is similar to inter-node feature centroid, but the difference is that instead of measuring vector similarity, it measures topological distances on the nodes.
A low inter-node position value means that nodes with assigned dataset items of a certain class are topologically closer together.

Intra-cluster Purity - The intra-cluster purity is the ratio of the number of the majority of the assigned dataset items to the total number of assigned items in a cluster. A high intra-cluster purity shows that clusters contain more dataset items of the same class, which can be directly interpreted as the quality of the cluster. The clustering algorithm used is the K-means algorithm. The algorithm is described in (Jain, 2008).

6. RESULTS

Based on the computed quality measure (refer to Table 2), the quality of the clusters generally degrades as the compression increases. In other words, the clusters that emerged for dataset 200-20 have higher quality than the more compressed dataset 50-5. In terms of upload and download intra-node purity, the decrease is much more drastic when moving from dataset 100-10 to dataset 50-5. This suggests that such a high compression starts to compromise the structure of the dataset. Furthermore, the rate of degradation of quality may grow quickly as compression rate increases, although the current data is not enough to confirm this observation.

It is interesting to note that in download inter-node position centroid and upload intra-cluster purity, the quality actually increased when moving from dataset 200-20 to dataset100-10. This shows that there are a lot of factors affecting the final quality measure, including the randomness of RPM, SOM training and K-means algorithm. This randomness means that the values of the quality measure vary for each run of the experiment, and that it is possible to have good results on one run and bad results in another.

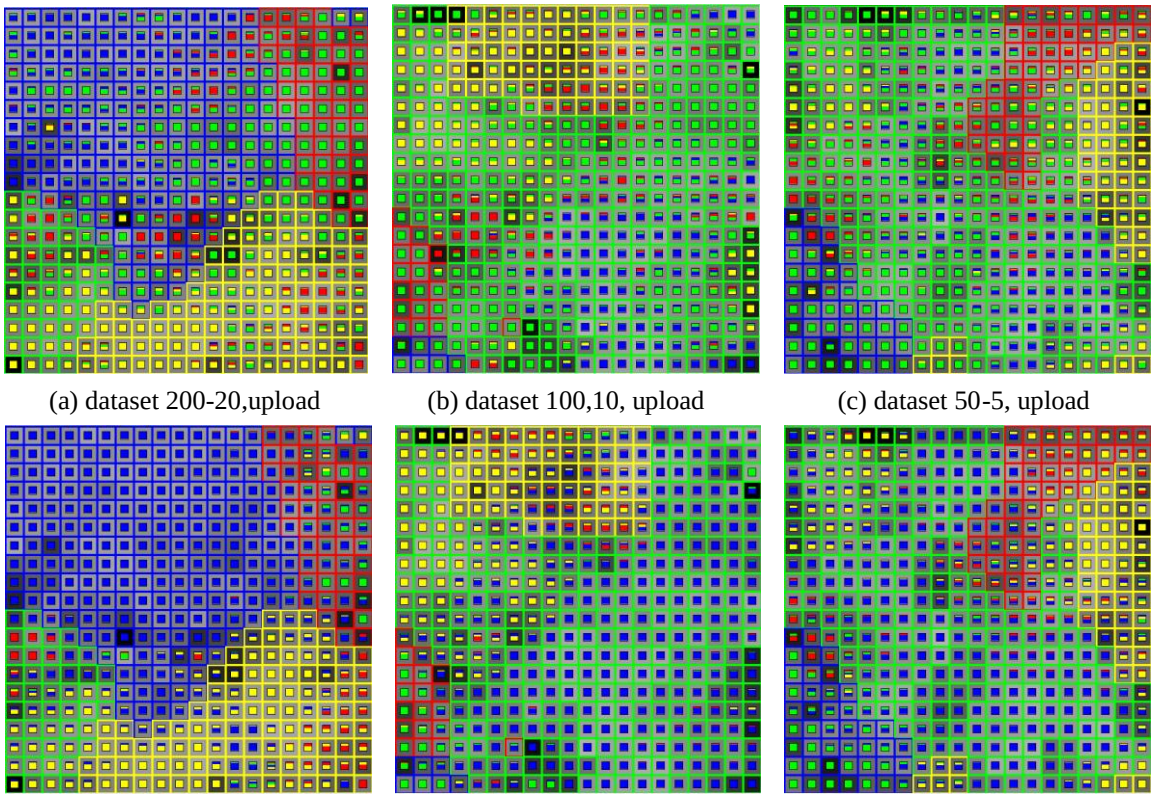
Table 2. Summary of average quality measures.

Quality Measure	Dataset 200-20	Dataset 100-10	Dataset 50-5
upload intra-node purity	0.75172	0.71035	0.62569

download intra-node purity	0.81962	0.81237	0.72549
upload intra-node similarity	0.99746	0.99653	0.99643
upload inter-node feature centroid	0.99956	0.99933	0.99913
download inter-node feature centroid	0.99937	0.99910	0.99894
upload inter-node position centroid	6.30912	6.49991	7.20797
download inter-node position centroid	6.74932	6.62166	7.07850
upload intra-cluster purity	0.60635	0.67682	0.54190
download intra-cluster purity	0.63044	0.60607	0.58823

Another thing to note is that there is no data on the actual structure of the dataset without using RPM, thus making it impossible to compare the results to a benchmark.

Despite the mentioned problems, the data still shows that RPM does preserve the structure of the dataset despite compressing more than 47,000 dimensions into 50, 100 or 200. The relatively high values of intra-node purity suggest that items of the same class still group together, which in turn suggests that the dataset structure is preserved.



(a) dataset 200-20,upload (b) dataset 100,10, upload (c) dataset 50-5, upload

(d) dataset 200-20, download (e) dataset 100-10, download (f) dataset 100-10, download

Figure 1. The different generated SOMs for datasets 200-20, 100-10 and 50-5, with one each for downloading and uploading of dataset items.

In Figure 1, it can be seen that on (a) that in the upload SOM of dataset 200-20, the blue class items (GCAT), the yellow class items (MCAT) and the green class items (CCAT) remain relatively clustered together, with the green class items extending out from the main cluster (top right). The red class items (ECAT) are scattered on the map, forming small sub-clusters.

This trend is generally preserved on the download SOM of dataset 100-10 (b) and dataset 50-5 (c), although with noticeable decrease in quality of clustering. Large clusters of blue, yellow and green are still visible, although sub-clusters are also present somewhere else on the map. This preservation of structure shows the relative effectiveness of RPM of compressing data, as well as showing the SOM's ability to visualize that structure.

In all download SOMs (d), (e) and (f) in figure 1, the blue class is the noticeable majority, with the yellow class also forming its cluster. Again, this consistency in the trend shows the structure-preserving capability of RPM.

7. CONCLUSIONS AND RECOMMENDATIONS

Based on the results of both the quantitative data and visual analysis of the SOMs, this project concludes that RPM is effective at compressing high-dimensional data while preserving its structure, although deterioration of quality is evident as compression level increases. Furthermore, it is hypothesized that the degradation of quality grows exponentially as the compression level increases.

Despite not being able to directly compare the results to the SOM of the actual dataset, having visual structure in the SOMs as well as the relatively high values of quality measure suggest that RPM is effective. This project recommends that further experiments are conducted regarding the quality of the preservation of structure, which includes comparing results to the actual dataset without RPM. Additionally, the individual effects of the RPM parameters *r* and *m* could be studied.

8. REFERENCES

Azcarraga, A. P., Yap, T. N., Tan, J., & Chua, T. S. (2004). *Evaluating Keyword Selection Methods for WEBSOM Text Archives.* IEEE Trans. on Knowl. and Data Eng., 16 , 380

Chifu, E. S., & Letia, I. A. (2010). *Self-organizing maps in web mining and semantic web*. In G. K. Matsopoulos (Ed.), (p. 357-380).

Dy, J. B. (2011). *Keyword Extraction for Very High Dimensional Datasets using Random Projection as Key Input Representation Scheme*. Master of Science Thesis, De La Salle University.

Huang, A. (2008, 14-18). *Similarity measures for text document clustering*. In J. Holland, A. Nicholas, & D. Brignoli (Eds.), New Zealand computer science research student conference (pp. 49–56).

Jain, A. K. (2008). *Data clustering: 50 years beyond k-means.*

Lewis, D. D., Yang, Y., Rose, T. G., Li, F. (2004). Rcv1: *A new benchmark collection for text categorization research*. Journal of Machine Learning Research, 5:361–397.