OPTICS on Text Data: Experiments and Test Results

Deepak P[*Δ], Shourya Roy[#]
[*]Department of CS&E, IIT Madras, Chennai, India
[#]IBM India Research Lab, IIT Delhi, Hauz Khas, New Delhi, India
deepakswallet@gmail.com, rshourya@in.ibm.com

Abstract

Clustering, particularly text clustering, in data mining has been attracting a lot of attention of late. There have been conventional techniques like K-means, which involve parameters that can't be easily estimated. With the emergence of density-based clustering algorithms which have significant advantages, a lot of attention has been devoted to them. OPTICS [1] is the latest and most sophisticated technique in this direction, and has been shown to be considerably tolerant to value changes in parameters. To the best of our knowledge, this is the first study on the applicability of OPTICS on text data. We perform a variety of experiments towards this end using various feature selection techniques (which,as we show, assume greater significance in the context of density based clustering), quantify our results by way of explanations and list conclusions.

1. Introduction

Clustering is a classical data mining task which aims at grouping data such that elements in the same group are more similar to each other than elements which are in different groups. With the exploding volume of text data due to the advent of the Web, clustering unstructured text documents has become a very useful task. Density based clustering has gained a lot of attention of late, mainly due to its numerous advantages over traditional clustering techniques like K-means [2]; most significantly due to the fact that density based algorithms don't require the number of clusters (the difficulty of estimating which has been widely acknowledged [3,4]) as input and can discover non-convex clusters as opposed to K-Means. Secondly, density-based algorithms usually provide a clustering [5] rather than a dendrogram (e.g., hierarchical agglomerative clustering, HAC) [6] which doesn't map to an obvious unique clustering. Hierarchical algorithms, apart from their scalability problems (due to quadratic complexity) require a termination condition [7] to determine when to stop merging. Thirdly, density-based algorithms such as OPTICS [1] have been shown to be useful for hierarchical clustering of late [8]. It may be noted here that OPTICS is cheaper (computationally) compared to HAC [9] when a spatial index is used. Although these advantages have been demonstrated by various researchers using synthetic low-dimensional datasets, this, to the best of our knowledge, is the first attempt on using density based clustering algorithms on text data apart from a vague related mention that density-based clustering methods may not scale well with increase in dimensions [10].

Section 2 reviews various density-based clustering algorithms. Sections 3 and 4 describe the performance measures and feature selection techniques used respectively. Section 5 describes our experiments with OPTICS. Section 6 lists the major contributions and conclusions.

2. Density Based Clustering Algorithms

Density based clustering methods cluster data based on a local cluster criterion such as density connected points. Typically, density based algorithms can discover clusters of arbitrary shapes and are relatively noise-tolerant. DBSCAN [5], the earliest density based clustering algorithm, introduces many concepts which are used by later density based clustering algorithms. It classifies points as *core points* if they have many data elements in their vicinity. Thereafter, a cluster can be represented by the set of core points it contains. The algorithm can identify clusters of arbitrarily shape opposed to K-Means and its variants. DENCLUE [11], a generalization of DBSCAN associates each data element with an influence function wherein clusters can be identified as regions which have high densities with respect to the influence function. Another algorithm, CURE [12] extends K-means, to allow multiple representative points for a single cluster

2.1 OPTICS [1]

OPTICS produces an augmented ordering of the elements in the dataset representing its clustering structure and has been shown to be quite insensitive to the input parameters [1] (as opposed to pre-OPTICS algorithms) provided that the values of the parameters are large enough to get a 'good' result. OPTICS builds a *reachability plot*, in which valleys correspond to clusters. The OPTICS plot is the plot of data elements, against their reachability distance, data elements ordered according to the time at which OPTICS stops considering them. The reachability distance of an

[Δ] Work done while doing internship at IBM India Research Lab element is determined by the distance to its nearest core point which has already been considered by OPTICS. Relative insensitivity to parameters (which enables it to identify clusters of varying densities) was the main motivation for us to choose OPTICS from among other density based algorithms for our experiments. Secondly, it has been shown that OPTICS can be used for hierarchical clustering [13]. Finally, OPTICS would work well in all cases where DBSCAN and DENCLUE would work well, although the vice versa isn't true.

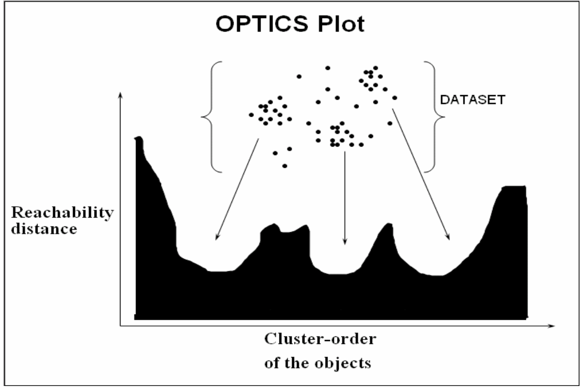

Figure 1. OPTICS Plot Example

Clusters can be identified [1] from the OPTICS reachability plot as a sequence of points which forms a valley (a region bounded by a downward slope followed by an upward slope). Such an algorithm causes an artificial reduction in cluster purity in cases of dense-sparse cluster interactions due to the inability to distinguish the sparse cluster from the dense cluster. Further, it may put a data element in multiple clusters which is undesirable for performance evaluation. Seeking to redeem these problems, we put forward our *Simple Cluster Identification* algorithm assumption as follows. The algorithm is intuitively derivable.

> "Simple Cluster Identification Algorithm" assumes that a cluster is a maximal sequence of points in the OPTICS plot which have comparable OPTICS values. A sequence of points have comparable values if the OPTICS values of the extreme points, i.e., those which have extreme OPTICS values, are not more than "e" apart in their OPTICS (reachability) values.

SCI identifies a sequence of points with "close" OPTICS values as a cluster and may split nested clusters, retaining the purity of the OPTICS clustering. We argue that SCI defines an upper bound on the purity (Ref. Section 3) of the clustering that can be given by any of the state-of-the-art algorithms for cluster identification [14] from the OPTICS plot. It may however be noted, that the average cardinality of the clusters identified by SCI would be much lower than that given by other algorithms and hence, the SCI clusters are not particularly useful. Extremely pure SCI clusters would give us enough confidence to go ahead and try out other cluster identification algorithms, whereas low values of purity would enable us to infer that OPTICS isn't well-suited for text data.

3. Performance Measures Used

Purity of a Cluster: The purity of a cluster [15] is the fraction of documents labeled with the maximally represented label in the cluster.

$$Purity(c) = \frac{\max(\{|d_c(c_i)|, c_i \in labels\})}{|c|}$$

$d_c(c_i)$ is the number of documents with label c_i in c.

Purity of a set of Clusters: The purity of a set of clusters is defined as the weighted sum of the purities of the clusters [15], each cluster (represented by C_1, C_2 etc in the formula below) weighted by its cardinality.

$$Purity\ (C_1, C_2, ..., C_k) = \frac{\sum_{i=1}^{k} Purity\ (C_i) * |C_i|}{\sum_{i=1}^{k} |C_i|}$$

Coverage: It is defined as the fraction of the dataset, D that is clustered by the Clustering algorithm, C. Coverage is of significance because the quality of the clustering is meaningful only when a large majority of data elements are clustered. Reduction in coverage may be caused, both due to the feature selection technique and the inability of the density based algorithm to cluster some data elements.

$$Coverage(D, C) = \frac{|\{d \mid d \in D \wedge \exists (C_i \in C), d \in C_i\}|}{|D|}$$

4. Feature Selection Methods used

The inherent high-dimensionality of text data, where each unique word is considered as a feature, makes feature selection all the more important. The main feature selection techniques that we deployed in the course of our experiments are Information Gain (IG) [16], Document Frequency (DF) [17], Dash-Liu Entropy (DL) [17,18], Entropy Based Ranking (EBR) [19] and Scaled Entropy [26]. We use the IG measure, which assumes existence of class label information, as a benchmark to test the performance of other unsupervised feature selection algorithms.

5. OPTICS on Reuters flat Clusters

In this section, we present the methodology, results and observations of our experiments with OPTICS on text data. We use the Reuters dataset [20] for our experiments. Only the documents uniquely labeled by one of "crude", "trade", "grain", "money-fx", "ship" and “interest” were used. This subset (R6 dataset) has 1570 documents which contain 11019 unique words.

5.1 OPTICS with Unsupervised Feature Selection

We test the performance of the OPTICS-SCI combination on the R6 dataset using four different unsupervised feature selection methods, DF, EBR and SE. It may be noted that the number of features used in these experiments is in the order of 10 – 100 and thus we use only less than 1% of the original set of features.

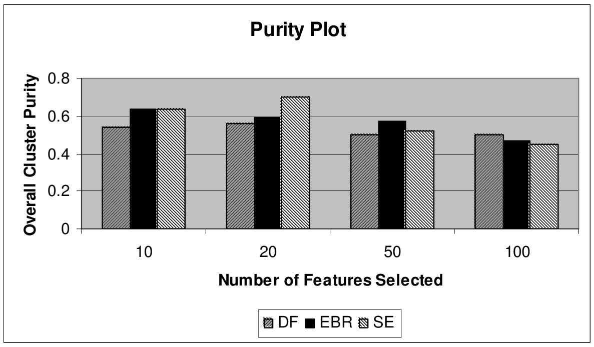

Figure 2. OPTICS-SCI on Selected Features: Purities

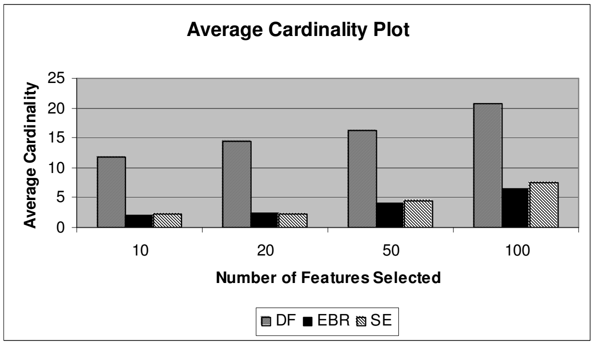

Figure 3. OPTICS -SCI on Selected Features: Average Cardinalities

The first striking observation (Figs. 2 and 3) is that the purity decreases by large proportions as the number of features considered increases. But with the decrease in the number of features considered, it was found that the coverage decreases drastically, especially for EBR and SE techniques. Deterioration with increase in the number of dimensions cannot be done away with as it is a property of the OPTICS algorithm. Therefore, the target of optimization would be in selecting features for consideration.

5.2 OPTICS with Supervised Feature Selection

As the earlier experiment suggests the usage of a much reduced feature space (say, 10-20 features), we use DL, EBR, SE and IG methods to do so. Good performance on a highly reduced feature space may mean that better unsupervised feature selection may aid OPTICS.

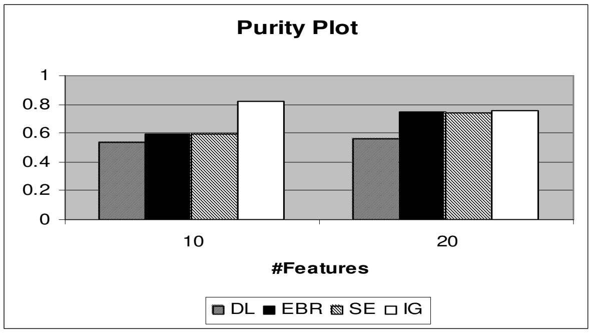

Figure 4. OPTICS-SCI on Selected Features: Purities

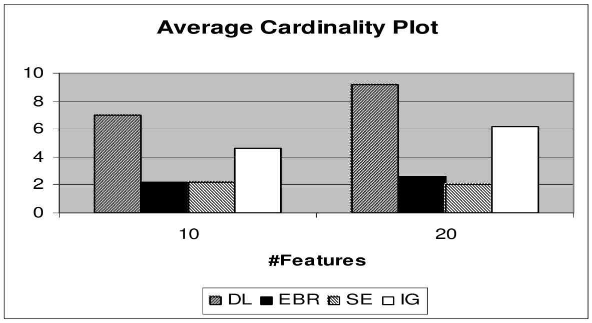

Figure 5. OPTICS-SCI on Selected Features: Average Cardinalities

As can be seen (Figs 4 and 5), with 20 of the supervised features, a purity of 0.75 is achievable using IG. This might be regarded as an upper bound on the performance any optimization (on unsupervised feature selection techniques) could give based on the assumption that unsupervised feature selection would not work better than IG [21].

5.3 K-Means on Reduced Feature Sets

We now experiment with the K-means algorithm using CLUTO [22] to gauge the performance of the algorithms which we would like OPTICS to compete with. We use the IG and the EBR feature selection methods on the R6 dataset. Here we use the same amount of features as in Section 5.1

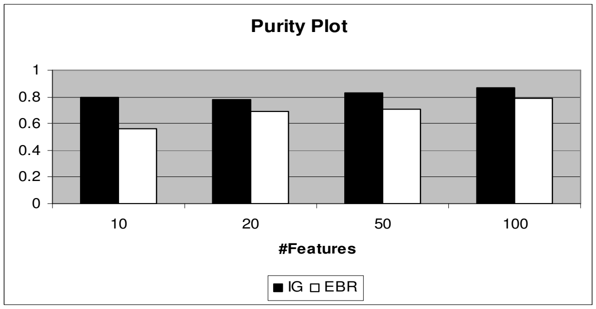

Figure 6. K-Means on Selected Features: Purities

As can be seen (Fig. 6), purity of the clusters is directly related to the number of features. This is contrary to our observation in the case of density based clustering algorithms where they are inversely related. This represents a significant finding, that K-means clustering works better with more features, whereas OPTICS deteriorates with the increase in the number of features.

5.4 OPTICS and Random Projections

As our next step in trying to get OPTICS working, we proceed to use simpler methods to select features. It has been proved repeatedly that the Occam's razor does work well in more cases than not. A classical example in this case is random projections [23]. We project the data on various random directions and get the clustering results from these directions and combine the results using the following algorithm.

Table 1. Aggregating Clusterings: The Algorithm

```
Aggregate-Clusterings(Set of Clusterings, {$C_1$,..$C_k$})
{
   Any two documents <$d_i$,$d_j$> are linked must-link if they occur in the same cluster in the majority of the k clusterings taken as input;
   The aggregated clustering would be the transitive closure of the must-links  thus generated;
}
```

This algorithm, we would argue, plays safe and is optimized towards getting clusters of better purity rather than in clustering most of the data. The R6 dataset (pruned to 500 features using IG feature selection, to get an upper bound on performance) used for the experiments, details of which comprise Table 2.

Table 2. Experiments using Random Projections

```
Experiment Methodology
Clusterings allclusterings = nullset;
for(i=0;i<11;i++)
{
    Project data on 10 random normalized directions;
    Apply the OPTICS-SCI combo to get clustering C;
    allclusterings = allclusterings U C;
}
Apply the clustering aggregation algorithm on allclusterings;
Output the results;
```

Table 3. Results with Random Projections

Results
Purity: 0.82 Average Cardinality: 1.71
After discarding clusters of size less than < 3 Purity: 0.69 Average Cardinality: 3.92

The results do not mark a significant improvement of performance over the previous experiments. The average cardinality of the clusters is too low to infer anything from the purity values.

6. Contributions and Conclusions

As a part of this work, we implemented an analyzed OPTICS on text data and gathered valuable insights into the working of OPTICS and it's applicability on text data. The SCI algorithm presented in this paper to identify clusters from the OPTICS plot can be used as a benchmark to test for the performance of OPTICS based on purity and coverage performance measures. Further, we have shown that the Scaled Entropy measure works considerably better than the EBR feature selection technique. We also present a method to aggregate different clusterings which can be used to arrive at a soft upper-bound on the purity of the clustering achievable by the combination of different clusterings. Having tried the various different feature selection techniques with OPTICS and not having arrived at good results (to match K-Means and variants), we deem ourselves competent enough to conclude that it is very less probable to get OPTICS working well on text data.

Many results of reasonable significance could be derived out of this study. Firstly, the fact that K-means works well on text data clustering implies that the bias of K-means that clusters are convex, does hold good in text data. It shows that the idea of a cluster being represented by its core points which enables OPTICS to identify non-convex clusters hasn't worked too well. This can be read in tune with the apprehension that the documents, regardless of their clusters, tend to be separated by the roughly the same distances in higher dimensions [24]. Secondly, OPTICS is worst affected by the increase in dimensionality whereas K-means benefits by the same. This follows from the fact that K-means improves and OPTICS deteriorates with the increase in dimensionality. This provides a very important pointer for future work with OPTICS, i.e., feature selection is almost an inevitable pre-processing step for OPTICS like algorithms and the better the feature selection, better would be the performance. Thirdly, our results raise some serious questions about the validity of the OPTICS assumptions. OPTICS borrows the DBSCAN idea of a crisp classification of data elements into core elements and non-core elements. It has been shown that each cluster clusters well in a small set of dimensions [25] which can be considered as characteristic dimensions of the cluster. With the addition of many more dimensions, as does happen in a high-dimensional setting, different clusters get distorted by different measures. Thus, a single global value for e or minPts is rendered insufficient even for OPTICS, which is very insensitive to parameter values. This leads to the requirement of a fuzzy characterization of points as core or non-core, or a learning technique which can roughly identify the different parameter values for different regions in the vector space.

Future work in this area would include trying various other techniques for dimensionality reduction and improved techniques for feature selection to get the OPTICS algorithm working on text data. Further we are contemplating exploration of other techniques that do not require the number of clusters (as an input parameter) and their applicability to text data.

7. References

1. Ankerst, Breunig, Kriegel, Sander, "OPTICS: Ordering Points to Identify the Clustering Structure", SIGMOD Conference, 1999

2. MacQueen, J. B. (1967). "Some methods for classification and analysis of multivariate observations", Fifth Symposium on Math, Statistics, and Probability, Berkeley, CA, 1967

3. Dan Pelleg, Andrew Moore, "X-means: Extending K-means with Efficient Estimation of the Number of Clusters", ICML Conference, 2000

4. Hamerly, Elkan, "Learning the k in k-means", NIPS Conference, 2003

5. Ester, Kriegel, Sander, Xu, "A Density based algorithm for discovering clusters in large spatial databases with noise", KDD Conference, 1996

6. Jain, Dubes, "Algorithms for Clustering Data", Prentice-Hall, 1998

7. García J.A., Fdez-Valdivia J., Cortijo F. J., and Molina R. 1994. "A Dynamic Approach for Clustering Data", Signal Processing, Vol. 44, No. 2, 1994.

8. Stefan, Kriegel, Kroger, Pfeifle, "Visually Mining Through Cluster Hierarchies", SIAM DM Conference, 2002

9. Milenova, Campos, "O-Cluster: Scalable Clustering of Large High Dimensional Data Sets", ICDM Conference, 2002

10. Sugato Basu, "Semisupervised clustering: Learning with limited user feedback", Technical Report, UT Austin, 2003, UT-AI-TR-03-307

11. Hinneburg, Keim, "An efficient approach to clustering in large multimedia databases with noise", KDD Conference, 1998

12. Guha et. Al., "CURE: An Efficient Clustering Algorithm for Large Databases", J. Info. Systems, 2001

13. Brecheisen, Kriegel, Kroger, Pfeifle, "Visually mining through cluster hierarchies", SIAM DM Conference, 2004

14. J Sander, X Qin, Z Lu, N Niu, A Kovarsky, "Automatic Extraction of Clusters from Hierarchical Clustering Representations", PAKDD Conference, 2003

15. Zhao, Karypis, "Criterion Function for Document Clustering: Experiments and Analysis", Department of CS, University of Minnesota, TR#01-40

16. George Forman, "An extensive empirical study of feature selection metrics for text classification", JMLR, 2003

17. Liu, Liu, Chen, Ma, "An Evaluation of feature selection for clustering", ICML Conference, 2003

18. Manorjan Dash, Liu, "Feature Selection for Clustering", PAKDD Conference, 2000

19. Liu, Li, Wong, "A comparative study on feature selection and classification methods using Gene Expression Profiles and Proteomic patterns", Genome Informatics, 13: 51-60, 2002

20. Koller, D., Sahami, "Hierarchically classifying documents using very few words", ICML Conference, 1997

21. McCallum, "Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering", 1996

22. Karypis, "CLUTO: A Clustering Toolkit", TR 02-017, Technical Report, Department of Computer Science, University of Minnesotta, 2002

23. Bingham, Mannila, "Random projection in dimensionality reduction: Applications to image and text data", KDD Conference, 2001

24. Beyer, Goldstein, Ramakrishnan, "When is nearest neighbors meaningful", ICDT Conference, 1999

25. Agarwal et. Al., "Automatic Subspace Clustering of High Dimensional Data for Data Mining Applications", SIGMOD Conference, 1998

26. Deepak P, Shourya Roy, "Scaled Entropy and DF-SE: Different and Improved Unsupervised Feature Selection Techniques for Text Clustering", to appear in the International Workshop on Feature Selection for Data Mining (FSDM 2006) to be held in conjunction with the SIAM DM Conference, 2006