International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)
Web Site: www.ijettcs.org Email: editor@ijettcs.org
Volume 3, Issue 5, September-October 2014 **ISSN 2278-6856**

Comparative Study of Document Similarity Algorithms and Clustering Algorithms for Sentiment Analysis

Rugved Deshpande[1], Ketan Vaze[2], Suratsingh Rathod[3], Tushar Jarhad[4]

1Department of Computer Engineering, P.E.S Modern College of Engineering, Shivajinagar, Pune, India

2Department of Computer Engineering, P.E.S Modern College of Engineering, Shivajinagar, Pune, India

3Department of Computer Engineering, P.E.S Modern College of Engineering, Shivajinagar, Pune, India

4Department of Computer Engineering, P.E.S Modern College of Engineering, Shivajinagar, Pune, India

Abstract

Sentiment analysis is the field of study that analyzes people's opinions, sentiments and emotions towards entities such as products, services,events, topics, and their attributes.With the explosive growth of social media (e.g. reviews,blogs,Twitter,comments in social network sites)on the Web, individuals and organizations are increasingly using the content in these media for decision making.In the real world, businesses and organizations always want to find consumer or public opinions about their products and services. Individual consumers also want to know the opinions of existing users of a product before purchasing it. Document similarity is a metric defined over a set of documents, where the idea of distance between them is based on the likeness of their meaning or semantic content. Clustering is a useful technique that organizes a large quantity of unordered text documents into a small number of meaningful and coherent clusters.

Keywords:- Sentiment Analysis, Similarity Techniques, Clustering, Cosine and K-means algorithm.

1. INTRODUCTION

In this paper, we have presented a comparative study of document similarity and clustering algorithms for sentiment analysis. A basic task in sentiment analysis is classifying the polarity of a given text in the document, sentence level, whether the expressed opinion in a document or a sentence aspect is positive, negative, or neutral. Beyond polarity sentiment classification looks, at emotional states such as "angry," "sad," and "happy"[2].

"Similarity is the state or fact of being similar while similar is referring to a resemblance in appearance,character, or quantity, without being identical".In order to compute the similarity of documents we need some mathematical expression or an algorithmthe computer can work with. This is called similarity or distance measure witch maps down the similarityor difference to one single numeric value[1].

2. CLASSIFICATIONOF DOCUMENT SIMILARITY MEASURE TECHNIQUES

- Jacard similarity measure
- Metric similarity measure
- Euclidean Distance measure
- Cosine similarity measure

2.1 Jacard similarity: The Jacard coefficient (Tanimoto coefficient) measures similarity as the intersection divided by the union of the objects. For text document, it compares the sum weight of shared terms to the sum weight of terms that are present in either of the two documents but are not the shared terms. The formal definition is

$$\mathrm{SIM_j}(\overrightarrow{t_a}, \overrightarrow{t_b}) = = \frac{\overrightarrow{t_a} \cdot \overrightarrow{t_b}}{|\overrightarrow{t_a}| + |\overrightarrow{t_b}| - \overrightarrow{t_a} \cdot \overrightarrow{t_b}}$$

"The Jacard coefficient is, a similarity measure that ranges between 0 and 1".It'svalue is 1 When $t_a=t_b$ and 0 when they are disjoint.Where 1 means the two objects are the same and 0 means they are completely different. The corresponding distance measure is $DJ = 1 – SIM_j$[7].

2.2 Metric similarity: Ameasure 'd' must satisfy the following four conditions to qualify as a metric:

Let x and y be any two objects in a set and d(x, y) be the distance between x and y.

- The distance between any two points must be nonnegative i.e.

$d(x, y) \geq 0$.

- The distance between two objects will be zero if and only if the two objects are identicali.e.

d (x, y) = 0 if and only if x = y.

- The distance must be symmetric i.e. distance from x to y is the same as the distance from y to x, that is

d(x, y) = d(y, x).

- The measure must satisfy the triangle inequality, which is

$$d(x, z) \leq d(x, y) + d(y, z)$$[3].

2.3 Euclidean Distance: Euclidean distance is the ordinary distance between two points and can be easily measured with a ruler in two- or three-dimensional space. It is a standard metric for geometrical problems. Euclidean distance is widely used in clustering problems, includingtext clustering. It satisfies all the above four conditions and therefore is a true metric. It is also the default distance measure used with the *K*-means algorithm. Measuring distance between text documents, given two documents d_a and d_b represented by their term vectors t_a and t_b respectively, the Euclidean distance of the two documents is defined as

$$D_E(\overrightarrow{t_a}, \overrightarrow{t_b}) = (\sum_{t=1}^{m} |w_{t,a} - w_{t,b}|^2)^{1/2}$$

Where the term set is T = {t1, . . . , tm}. As mentioned previously, we use the tfidf value as term weights, i.e.
w_t,a = tfidf(d_a, t)[5].

2.4 Cosine similarity: In Cosine similarity documents are represented as term vectors. The similarity of two documents corresponds to the correlation between the vectors. This is quantified as the cosine of the angle between vectors that is known as cosine similarity. Cosine similarity is one of the most popular similarity measure applied to text documents. Given two documents t_a and t_b their cosine similarity is

$$\mathbf{SIM_c}(\overrightarrow{t_a}, \overrightarrow{t_b}) = = \frac{\overrightarrow{t_a} \cdot \overrightarrow{t_b}}{\overrightarrow{t_a} \times \overrightarrow{t_b}}$$ [7].

Where t_a and t_b are m-dimensional vectors over the term set T = {t1,……tm}. Each dimension represents a term with its weight in the document, which is always non-negative. As a result, the cosine similarity is non-negative and it's value varies between [0,1]. An important feature of the cosine similarity is its independence of document length. For example, If we combine two identical copies of a document d' to get a new pseudo document d0, the cosine similarity between d' and d0 will be 1, which means that these two documents are regarded to be identical. Meanwhile, given another document l, d' and d0 will have the same similarity value to l, i.e.sim($t_{d'}$,t_l)= sim(t_d ,t_l). In other words, documents with the same composition but different totals will be treated identically. Strictly speaking, this does not satisfy the second condition of a metric, because after all the combination of two copies is a different object from the original document. However, in practice, when the term vectors are normalized to a unit length such as 1, and in this case the representation of d and d0 is the same [3].

3. CLUSTERING

Clustering is an unsupervised learning task where one seeks to identify a finite set of categories termed clusters to describe the data.A similarity metric is defined between items of data, and then similar items are grouped together to form Clusters.The grouping of data into clusters is based on the principle of maximizing the intra class similarity and minimizing the inter class similarity. A good clustering methodwill produce high quality clusters with high intra-class similarity - Similar to one another within the same cluster low inter-class similarity - Dissimilar to the objects in other clusters. The quality of a clustering result depends on boththe similarity measure used by the method and its implementation[4]. Clustering algorithms can be broadly classified into three categories, in the following subsections together with specific algorithms:

- Partitioning
- Hierarchical
- Density-based

3.1 Partitioning Clustering Algorithms

Partitioning clustering attempts to decompose a set of N objects into k clusters such that the partitions optimize a certain criterionfunction. Each cluster is represented by the centre of gravity (or centroid) of the cluster, e.g. k-means

3.1.1 K-means

K-means is the most popular clustering method in metric spaces. Initially 'k' cluster centroids are selected at random. *k*-meansthen reassigns all the points to their nearest centroids and recomputed centroids of the newly assembled groups. The iterativerelocation continues until the criterion function is fulfilled. e.g. square-error converges.Finally, this algorithm aims at minimizing an objective function; in this case a squared error function. The objective function

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i^{(j)} - c_j \right\|^2 ,$$

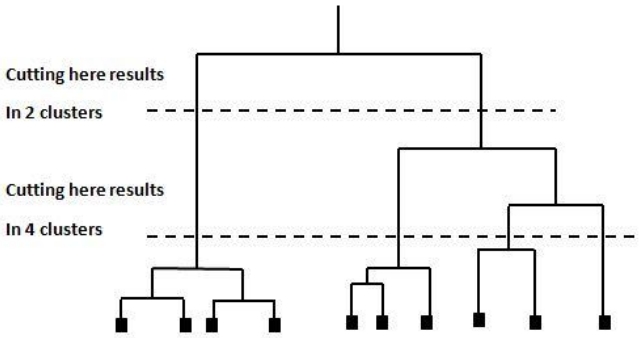
where || xi(j) – cj ||2 is a chosen distance measure between a data point xi(j)and the cluster centre Cj, is an indicator of the distanceof the n data points from their respective cluster centres.The steps of the algorithm are :

1. Choose the number of clusters, *k*.
2. Randomly generate *k* clusters and determine the cluster centers, or directly generate *k* random points as cluster centers.
3. Assign each point to the nearest cluster center.
4. Recompute the new cluster centers.
5. Repeat the two previous steps until some convergence criterion is met [6].

3.2 Hierarchicalalgorithms

Unlike partitioning methods that create a single partition, hierarchical algorithms produce a nested cluster, with a single all-inclusive cluster at the top and singleton clusters of individual points at thebottom. The hierarchy can be formed in top-down orbottom-up fashion and need not

necessarily be extended to the extremes.



The merging or splitting of clusters stops once the desired number of clusters has been formed. Typically, each iteration involves merging or splitting a pair of clusters based on a certain criterion, often it is a measuring the proximity between clusters. Hierarchical techniques suffer from the fact that previously taken steps (merge or split), possibly erroneous, are irreversible. [6] Some representative examples of hierarchical algorithms are:

- CLUSTERING USING REPRESENTATIVES (CURE)
- CHAMELEON
- BIRCH

3.3 Density-based clustering algorithms

Density-based clustering methods group neighbouring objects into clusters based on local density conditions rather than proximity between objects. These methods regard clusters as dense regions being separated by low density noisy regions. Density-basedmethods have noise tolerance and can discover non-convex clusters.Some representative examples of density based clustering algorithms are:

- Density-Based Spatial Clustering of Applications with Noise (DBSCAN)
- Density basedClustering (DENCLUE)[4]

4. DIFFERENTIAL ANALYSIS OF PERFORMANCE

	K-means Algorithm	HC Algorithm	Density Based Algorithm
Size of Dataset (Huge)	Best	Better	Good
Number of Cluster(Huge)	Best	Good	Better
Data Type (Random)	Good	Best	Better
Usability	Easy to implement	Complex	Moderate
Effect of Noise	Sensitive	More Sensitive	Sensitive

In Table I performance of clustering algorithms is compared on the size of dataset,type of the dataset, number of the clusters and usability [4].

5. CONCLUSION

In sentimental analysis of data (such as twitter corpus) we found that the similarity measure being used in clustering has its impact on the resultant clusters [5].In the data mining domain, Cosine Similarity is simple and effective than other Document Similarity Measure Techniques. It is simple, scalable and easy to implement. Its good quality is that it can be used with algorithms in combination to yield good results. That's why it is popular and most widely used. Considering Clustering algorithms, as the number of clusters, k becomes greater, k-means shows better performance than hierarchical clustering algorithms and density based clustering algorithms.All the algorithms have some ambiguity when noisy data is clustered. *K*-means has lower quality than others. The quality (accuracy) of *k*-means algorithm increases when using huge dataset. Hierarchical and Density based algorithms show good results when using small dataset.Hierarchical and Density based algorithms give better results compared to *k*-means when using random dataset. Considering all the factors that affects performance of Document Similarity Algorithm and Clustering Algorithm, it is found that *k*-means algorithm gives overall best result when used with Cosine Similarity.

6. ACKNOWLEDGEMENT

We would like to express our gratitude to Prof. Ms Deipali Gore and Prof. Mrs Manisha Petare who have guided us regarding matters where we needed clarity about the subject. We are thankful for their aspiring guidance, invaluably constructive criticism and advice during the course of this study.

REFERENCE

[1] Eun Hee Ko, Diego Klabjan, "Semantic Properties of Customer Sentiment in Tweets," 2014 28th International Conference on Advanced Information Networking and Applications Workshops.
[2] B. Liu, Sentiment analysis and opinion mining. San Rafael, CA: Morgan & Claypool Publishers, 2012.
[3] Anna Huang, "Similarity Measures for Text Document Clustering," NZCSRSC 2008, April 2008, Christchurch, New Zealand.
[4] Osama Abu Abbas, "Comparison Between Data Clustering Algorithms," The International Arab Journal of Information Technology, Vol. 5, No. 3, July 2008.

[5] K. Gimpel et al., "Part-of-speech tagging for Twitter: annotation, features, and experiments," HLT'11 Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, vol. 2, pp. 42-47, 2010.

[6] Deepti Sisodia and Lokesh Singh, "Clustering Techniques: A Brief Survey of Different Clustering Algorithms" , International Journal of Latest Trends in Engineering and Technology,Vol. 1 Issue 3 September 2012.

[7] Sapna Chauhan and Pridhi Arora , "Algorithm for Semantic Based Similarity Measure",International Journal of Engineering Science Invention, Volume 2 Issue 6 ǁ June. 2013.