FRECCA based Sentence Level Text Clustering

Nikita Gangshetty[1] Prof. Mr. Manoj Kumar H[2]
[1]M.Tech. Student [2]Assistant Professor
[1] Computer Networking Engineering Department
[1] RNS Institute of Technology, Bangalore, Karnataka, India.

Abstract— Sentence clustering plays an important role in many text processing activities. In comparison with hard clustering methods, in which a pattern belongs to a single cluster, fuzzy clustering algorithms allow patterns to belong to all clusters with differing degrees of membership. This is important in domains such as sentence clustering, since a sentence is likely to be related to more than one theme or topic present within a document or set of documents. For example, consider web mining, where the specific objective might be to discover some novel information from a set of documents initially retrieved in response to some query. By clustering the sentences of those documents intuitively can expect at least one of the clusters to be closely related to the concepts described by the query terms. However, because most sentence similarity measures do not represent sentences in a common metric space, conventional fuzzy clustering approaches based on prototypes or mixtures of Gaussians are generally not applicable to sentence clustering. It is a novel fuzzy clustering algorithm that operates on relational input data; i.e., data in the form of a square matrix of pair wise similarities between data objects. The algorithm uses a graph representation of the data, and operates in an Expectation-Maximization framework in which the graph centrality of an object in the graph is interpreted as likelihood. Results of applying the algorithm to sentence clustering tasks demonstrate that the algorithm is capable of identifying overlapping clusters of semantically related sentences, and that it is therefore of potential use in a variety of text mining tasks. It also includes results of applying the algorithm to benchmark data sets in several other domains.

Keywords: Sentence Clustering, PageRank, Graph based Centrality.

I. INTRODUCTION

Irrespective of the specific task (e.g., summarization, text mining, etc.), most documents will contain interrelated topics or themes, and many sentences will be related to some degree to a number of these. The work described is motivated by the belief that successfully being able to capture such fuzzy relationships will lead to an increase in the breadth and scope of problems to which sentence clustering can be applied. However, clustering text at the sentence level poses specific challenges not present when clustering larger segments of text, such as documents. Now lets highlight some important differences between clustering at these two levels, and examine some existing approaches to fuzzy clustering.

Clustering text at the document level is well established in the Information Retrieval (IR) literature, where documents are typically represented as data points in a high dimensional vector space in which each dimension corresponds to a unique keyword, leading to a rectangular representation in which rows represent documents and columns represent attributes of those documents (e.g., tf-idf values of the keywords).

This type of data, which is referred to as "attribute data," is amenable to clustering by a large range of algorithms. Since data points lie in a metric space, it can be readily applied to prototype-based algorithms such as k-Means, Isodata, Fuzzy c-Means (FCM), and the closely related mixture model approach, all of which represent clusters in terms of parameters such as means and covariances, and therefore assume a common metric input space. Since pairwise similarities or dissimilarities between data points can readily be calculated from the attribute data using similarity measures such as cosine similarity, and can also apply relational clustering algorithms such as Spectral Clustering and Affinity Propagation, which take input data in the form of a square matrix W (often referred to as the "affinity matrix"), where w_{ij} is the (pairwise) relationship between the i^{th} and j^{th} data object.

II. RELATED WORK

The vector space model has been successful in IR because it is able to adequately capture much of the semantic content of document-level text. This is because documents that are semantically related are likely to contain many words in common, and thus are found to be similar according to popular vector space measures such as cosine similarity, which are based on word co-occurrence. However, while the assumption that (semantic) similarity can be measured in terms of word co-occurrence may be valid at the document level, the assumption does not hold for small-sized text fragments such as sentences, since two sentences may be semantically related despite having few, if any, words in common. To solve this problem, a number of sentence similarity measures have recently been proposed. Rather than representing sentences in a common vector space, these measures define sentence similarity as some function of intersentence word-to-word similarities, where these similarities are in turn usually derived either from distributional information from some corpora (corpus-based measures), or semantic information represented in external sources such as WordNet (knowledge-based measures). Some of these measures are described. The important to note is that these measures do not represent sentences in a common metric space, and this means that prototype-based clustering algorithms such as those described above are generally not applicable. The topic of interest, therefore, is fuzzy relational clustering, i.e., fuzzy clustering based on (pairwise) relational input data.

To distinguish it from attribute data, this data is referred as "relational data." A broad range of hierarchical clustering algorithms can also be applied. The vector space model has been successful in IR because it is able to adequately capture much of the semantic content of

document-level text. This is because documents that are semantically related are likely to contain many words in common, and thus are found to be similar according to popular vector space measures such as cosine similarity, which are based on word co-occurrence. However, while the assumption that (semantic) similarity can be measured in terms of word co-occurrence may be valid at the document level, the assumption does not hold for small-sized text fragments such as sentences, since two sentences may be semantically related despite having few, if any, words in common.

III. PROPOSED WORK

The first successful fuzzy relational clustering model is generally considered to be Hathaway et al.'s Relational Fuzzy c-Means (RFCM) algorithm. However, RFCM is a variant of Fuzzy c-Means, and is implicitly based on the notion of prototype. Thus, while RFCM operates on relational data input, it still requires that the relation expressed by this data be Euclidean (i.e., it assumes that there exists a set of data points in some space such that the squared Euclidean distance between points in this space match those in the dissimilarity relation). Non-Euclidean relations can be transformed into Euclidean ones by a transformation that adds a positive number to all off diagonal elements of the dissimilarity matrix, but the problem is to determine an appropriate value for such that this Euclidean condition is met without leading to excessive loss of cluster information.

Despite its success, the Euclidean requirement in RFCM was considered restrictive, and various alternatives have been proposed. For example, the ARCA algorithm uses an attribute-based representation in which an object is represented by a vector of its relationships with other objects in the data set. Thus, while the algorithm still takes relational data as input, it treats each row of the relational input matrix as a data object, thus allowing standard Fuzzy c-Means to be applied. Prototypes in this system are therefore objects (not necessarily present in the original data set) whose relationship with all objects in the data set is representative of the mutual relationships of a group of similar objects. A limitation of this approach is the high dimensionality introduced by representing objects in terms of their similarity with all other objects. The contribution here is novel fuzzy relational clustering algorithm. Inspired by the mixture model approach, we model the data as a combination of components. However, unlike conventional mixture models, which operate in a Euclidean space and use a likelihood function parameterized by the means and covariances of Gaussian components, we abandon use of any explicit density model (e.g., Gaussian) for representing clusters. Instead, we use a graph representation in which nodes represent objects, and weighted edges represent the similarity between objects. Cluster membership values for each node represent the degree to which the object represented by that node belongs to each of the respective clusters, and mixing coefficients represent the probability of an object having been generated from that component. By applying the PageRank algorithm to each cluster, and interpreting the Page-Rank score of an object within some cluster as a likelihood, we can then use the Expectation-Maximization (EM) framework to determine the model parameters (i.e., cluster membership values and mixing coefficients).

The algorithm uses Expectation Maximization to optimize these parameters. Assume in the following that the similarities between objects are stored in a similarity matrix S, between objects i and j.

This paper presents a novel fuzzy clustering algorithm that operates on relational input data; i.e., data in the form of a square matrix of pairwise similarities between data objects. The algorithm uses a graph representation of the data, and operates in an Expectation-Maximization framework in which the graph centrality of an object in the graph is interpreted as likelihood. Results of applying the algorithm to sentence clustering tasks demonstrate that the algorithm is capable of identifying overlapping clusters of semantically related sentences, and that it is therefore of potential use in a variety of text mining tasks.

A. *Advantages of Proposed System:*

- Able to achieve superior performance to benchmark Spectral Clustering and k-Medoids algorithms when externally evaluated in hard clustering mode
- Comparisons with the ARCA algorithm on each of these data sets suggest that FRECCA is capable of identifying softer clusters than ARCA, without sacrificing performance as evaluated by external measures.

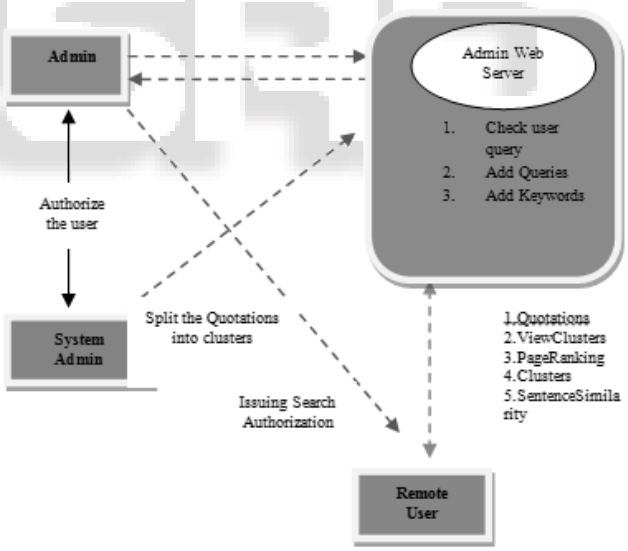

Fig. 1: System Architecture

IV. IMPLEMENTATION

This algorithm uses the PageRank score of an object within a cluster as a measure of its centrality to that cluster. These PageRank values are then treated as likelihoods. Since there is no parameterized likelihood function as such, the only parameters that need to be determined are the cluster membership values and mixing coefficients. The algorithm uses Expectation Maximization to optimize these parameters. Assume in the following that the similarities between objects are stored in a similarity matrix S, between objects i and j.

A. Initialization:

Assume here that cluster membership values are initialized randomly, and normalized such that cluster membership for an object sums to unity over all clusters. Mixing coefficients are initialized such that priors for all clusters are equal.

B. Expectation:

The E-step calculates the PageRank value for each object in each cluster. PageRank values for each cluster are calculated as described with the affinity matrix weights w_{ij} obtained by scaling the similarities by their cluster membership values; i.e.,

$$w_{ij}^{m} = s_{ij} \times p_i^m \times p_j^m$$

Where w_{ij} is the weight between objects i and j in cluster m, s_{ij} is the similarity between objects i and j, and P_i^m and P_j^m are the respective membership values of objects i and j to cluster m.

C. Maximization:

Since there is no parameterized likelihood function, the maximization step involves only the single step of updating the mixing coefficients based on membership values calculated in the equation,

$$p_i^m = (\pi_m \times l_i^m) / \sum_{j=1}^{c} \pi_j \times l_i^j$$

D. Initialization Step:

Begin

// initializing all values

for i=1 to N

for m=1 to C

Initialize P_i^m =random values

Normalize P_i^m such that the membership value of an object sums to unity

end for

end for

End

Find mixing coefficient, $\pi_m = 1/C$.

E. Expectation Step:

Begin

// calculating PageRank values

for m=1 to C

Create a weighted affinity matrix using,

$$w_{ij}^{m} = s_{ij} \times p_i^m \times p_j^m$$

Calculate Pagerank values

end for

End

// Assign PageRank scores to likelihoods

//Calculate new cluster membership values using likelihoods.

F. Maximization Step:

Begin

for m=1 to C

// Update mixing coefficients

Using Cluster membership values

end for

End

V. Conclusion

For a given sentence the method provides output, to which cluster does the sentence belongs to. Any keyword related to a cluster or any sort of synonym related would be considered to select the cluster name for a sentence. The method was motivated by interest in clustering of sentence-level text, and the need for an algorithm which can accomplish this task based on input data.

The method takes FRECCA as a reference to cluster the sentences, where the pagerank value of a particular word or a synonym is taken into consideration so as to find out the cluster name. The enhancement has logins for Admin as well as particular users, where the users can ask details about any kind of info. The users may send queries to the admin . The admin then can check the queries and he can reply to the queries asked by a particular user. The admin has rights to add extra keywords and make the method reliable to solve for various sentences. For a given sentence the method provides output, to which cluster does the sentence belongs to. Any keyword related to a cluster or any sort of synonym related would be considered to select the cluster name for a sentence. The method was motivated by interest in clustering of sentence-level text, and the need for an algorithm which can accomplish this task based on input data.

Acknowledgement

I take this Opportunity to express my profound gratitude and deep regards to my guide Prof, Mr. Manoj Kumar H Assistant Professor, RNS Institute of technology, Bangalore, for his exempla nary guidance, and constant encouragement throughout.

I would also like to thank Director Dr. H N Shivashankar, Principal Dr. M K Venkatesha and Dr. M V Sudhamani, professor and Head, Dept of Information Science and engineering, RNSIT, for constant encouragement in implementing this paper and pursuing this paper.

References

[1] Eloize Rossi Marques Seno Maria, *SiSPI: A Short-Passage Clustering System* das Graças Volpe Nunes, Janeiro, 2008

[2] Raymond Kosala *Web Mining Research: A Survey* Department of Computer Science Katholieke Universiteit Leuven Celestijnenlaan 200A, B3001 Heverlee, 2010

[3] Rasim Alguliev, Ramiz Aliguliyev, Makrufa *Multi-Document Summarization Model Based on Integer Linear Programming* Institute of Information Technology of National Academy of Sciences of Received August 28, 2010; revised October 1, 2010; accepted October 3, 2010

[4] Eduard Hovy and Chin-Yew Lin *Automated Text Summarization in SUMMARIST* Information Sciences Institute of the University of Southern California 4676 Admiralty Way Marina del Rey, CA, 2011
[5] Lili Kotlerman, Ido Dagan Bar-Ilan *Sentence Clustering via Projection over Term Clusters* University Israel, 2012
[6] Martina Naughton, Nicholas Kushmerick, and Joe Carthy *Clustering sentences for discovering events in news articles* School of Computer Science and Informatics, University College Dublin, Ireland, 2007
[7] Richard Khoury *Sentence Clustering Using Parts-of-Speech* Department of Software Engineering, Lakehead University, Thunder Bay (ON), Canada, 2003
[8] Vasileios Hatzivassiloglou, Judith L. Klavans, Melissa L. Holcombe, Regina Barzilay, Min-Yen Kan, and Kathleen R. McKeown *SIMFINDER: A Flexible Clustering Tool for Summarization* Department of Computer Science Columbia University,2008
[9] Yuhua Li, David McLean, Zuhair A. Bandar, James D. O'Shea, and Keeley Crockett *Sentence Similarity Based on Semantic Nets and Corpus Statistics* IEEE Transactions on Knowledge and Data Engineering, Vol. 18, No. 8, August 2006
[10] Joe Parker McDivith *Multi-document Summarization using Spectral Clustering* California state science fair 2011
[11] Hsun-Hui Huang Yau-Hwang Kuo Horng-Chang Yang *Fuzzy-Rough Set Aided Sentence Extraction Summarization* credit1, credit, Dept. of Computer Science Dept. of Computer Science Dept. of Computer Science,2006
[12] Rada Mihalcea and Paul Tarau *TextRank: Bringing Order into Texts* Department of Computer Science University of North Texas, 2006
[13] Eduardo Bezerra Semi-*Supervised Clustering of XML Documents*: Getting the most from Structural Information CEFET/RJ & COPPE/UFRJ Rio de Janeiro, RJ, Brazil, 2011
[14] ZHANG Pei-ying *Automatic text summarization based on sentences clustering and extraction* College of Computer & Communication Engineering China University of Petroleum Dongying, Shandong An Unsupervised Center Sentence-based Clustering Approach for Rule-based Question Answering 2011 IEEE Symposium on Computers & Informatics, 2006.

