A bird view at the Incremental and Hierarchical Record Clustering

(* Dr. Rui Alberto Cardoso, ** Dr. Paulo Jorge de Sousa Gomes)
(*Faculty of Informatics Engineering, University of Coimbra,
**Faculty of Science and Technology, University of Coimbra)

Abstract

It is critical that we discover tools to automatically arrange these huge collections of files. Report clustering is an lively studies domain that allows the automatic corporation of documents into clusters. This thesis targets to develop a new algorithm of hierarchical file clustering with a completely incremental and unsupervised approach. There are many distinctive record clustering algorithms, with exclusive motivations and approaches. Maximum algorithms receive a fixed of documents and system them as a whole. However, in nowadays online environment it is essential that a device can acquire and system files constantly.

The algorithm can be primarily based on well-known conceptual clustering algorithms that have seldom been applied to textual content. Hierarchical because it produces a tree of clusters that facilitates browsing. By incremental we mean that there is no need that all documents had been gift at the beginning. Each report is processed as soon as it's miles to be had and the clusters are permanently updated. Also, as lots of properly- established strategies in document clustering are not appropriate for incremental structures, one of the essential challenges of this research is to adapt these mechanisms (file illustration, function choice, assessment) to incremental document clustering.

Introduction

There is no to be had hierarchical record clustering machine which could claim to be absolutely incremental and unsupervised. One such set of rules might be extremely useful to deal with the overwhelming document surplus of these days. So, the primary goal of this

research is the advent of a brand-new document clustering set of rules with those assumptions. On this chapter we will introduce the report clustering region, gift a few key concepts associated with clustering and record clustering, that are imperative to this work.

Clustering

Clustering is the division of records into businesses of comparable items or "the art of finding agencies in statistics" (Kaufman & Rousseeuw, 1990). Every group (cluster) is manufactured from objects which can be similar between themselves (excessive intra- cluster similarity) but multiple to items of different clusters (low inter-cluster similarity). Clustering (or cluster evaluation) differs from type, where a hard and fast of predefined lessons is also supplied (supervised gaining knowledge of). In clustering, the system should determine now not most effective to which cluster every item must be assigned but also which clusters have to be created (unsupervised getting to know). The clustering end result (the set of clusters created) is primarily based entirely at the object representation, the similarity measure and the clustering set of rules. (Rosell, 2006).

There are numerous exclusive clustering algorithms, relying on one of a kind statistics kinds, one-of-a-kind programs and exclusive consumer requirements (Jain, 2010; Jain, Murty, & Flynn, 1999; Xiao, 2010; R. Xu & Wunsch, 2009). Typically, objects are represented by way of a fixed of features, denoted with the aid of a vector of values (numeric or nominal). It's been utilized in a wide style of fields: records, social sciences, biology, genomics, advertising, astronomy, picture segmentation, sample recognition, information retrieval, system studying and facts mining.

Report illustration

The terminology used throughout this idea calls for an evidence. We'll use record or textual content with the that means of a sequence of phrases. A file collection also can be known as a fixed of texts or a corpus (corpora in plural). Also, we will use phrase and term interchangeably to refer a sequence of alphanumeric characters delimited through non-

alphanumeric characters. Earlier than documents can be utilized in a clustering set of rules, they have to be represented in a suitable shape. The most used representation technique for report is based on vector space version (Salton, Wong, & Yang, 1975) in which every report is represented with the aid of a vector of weights of m "functions" extracted from the file:

$$di = [wi1, wi2, \dots , wim]$$

where wij represents the weight of the jth feature of document i. In order to account for documents of different sizes, each document vector is normalized to unitary length. Many approaches are possible to vector space model, depending on what a feature is and how the weights are computed.

When the features are the words occurring in the collection, which is the most usual, we have the bag-of-words model because a document is represented as a set of words, ignoring word order or any syntactical structure. More formally, in a collection with n documents that contains m different terms (words), the collection of documents can be represented by a document-term matrix of dimensions n x m:

	t1	t2	…	tk	…	tm
d1	w11	w12	…	w1k	…	w1m
d2	w21	w22	…	w2k	…	w2m
…	…	…	…	…	…	…
di	wi1	wi2	…	wik	…	wim
…	…	…	…	…	…	…
dn	wn1	wn2	…	wnk	…	wnm

Figure 2.1 - Document-term matrix representing a collection of documents

This matrix is normally extremely sparse because a document doesn't contain many distinct words (few hundreds) but a collection can contain tens of thousands of different words. There are many different ways to determine weights (Manning, Raghavan, & Schütze,

2008): The most intuitive is the term frequency denoted as tfd,t that counts how many times the term t appears in the document d. The simplest is term occurrence: a binary choice between the values 0 (if the term doesn't occur in the document) or 1 (if the term occurs).

However, in clustering we want more discriminative terms to have the biggest weights, and a term that occurs in all documents isn't discriminative. So we use the notion of document frequency (dft) as the number of documents in the collection that contains the term t. We can also calculate the collection frequency (cft) as the number of occurrences of term t in all documents of the collection, but document frequency is preferred. Since we want the terms appearing in fewer documents to have a higher weight, we must use the inverse document frequency (idf) of a term t defined as:

$$\text{idf}_t = \log \frac{N}{df_t}$$

where N is the total number of documents in the confiscation.

So the idf of a rare term is high and the idf of a frequent term is likely to be low. It should be stressed that idf evaluates the discriminating power of a term within a collection of documents so idf is not a local, but a global measure. The most used weighting scheme in document clustering is tf-idf (Term Frequency – Inverse Document Frequency) that combines term-frequency and inverse document frequency to assign a weight to each term in each document (Spärck Jones, 1972). There are many variants of formula, but the most usual is:

$$\text{tf-idf}_{d,t} = \text{tf}_{d,t} \times \text{idf}_t = \text{tf}_{d,t} \times \log \frac{N}{}$$

So, weight assigned by tf-idf to a term in a document is (Manning et al., 2008): higher when the term occurs frequently in a small number of documents (giving a high discriminating power to those documents); lower when the term occurs in many documents or occurs fewer times in a document (and zero when the term occurs in all documents).

Some alternatives to tf-idf have been considered (Manning et al., 2008). We can

consider term weights as being obtained by the product of a local weight ld,t, a global weight gt and (sometimes) a normalization factor nd, like this:

$$w_{d,t} = f(l_{d,t}, g_t, n_d) = l_{d,t} \times g_t \times n_d$$

Local weight is a function of the number of occurrences of t in document d, global weight is a function of the number of documents containing term t in the entire collection and the normalization factor for document d corrects for discrepancies in the lengths of the documents.

In the "classic" tf-idf already described we have $l_{d,t}=tf_{d,t}$, $g_d=idf_d$ and $n_d=1$.

There are many other options for each of the three terms, and each combination gives a potential new weight scheme. We'll briefly describe two of them: Sub-linear tf scaling – it is unlikely that ten occurrences of a term in a document carry ten times the significance of one just occurrence. There has been some research proposing changes on term frequency that goes beyond simple counting the number of occurrences. A common modification is to use the logarithm of the term frequency, which assigns a weight given by:

$$w_{ft,d} = \log(1 + tf_{t,d})$$

where a is a smoothing value between 0 and 1, generally set to 0.4 or 0.5. Many variations to vector space model have been proposed, including the replacement of words by other document features.

One of the most usual alternatives is multi-word terms, also called compound words or phrases. Using multi-word terms has the advantage of carry more information and, simultaneously, reduces the dimensionality. This approach should, in principle, produce better results but practical tests are not clear. There are still some researches where documents are represented by compound and simple words.

This representation have some advantages: is more sensitive to grammatical and typographical errors, all the possible terms of the vocabulary are known in advance and doesn't require any preprocessing step, which makes it language independent. However, for N=1 or N=2 the representation has little expressive power and for N≥3 the dimensionality is still a problem.

Where indicates the vector dot product and ‖d‖ is the length of vector d. How vectors are usually normalized to unit length the cosine measure can be calculated just by:

$$sim \quad (d1, d2) = d1 \bullet d2 = \cos(\measuredangle(d1, d2)$$

Types of Document Clustering Algorithms

Document clustering algorithms can be classified in many ways (Andrews & Fox, 2007; Jain, 2010; Jain et al., 1999). According to the number of levels of the clustering created, the methods can be divided in hierarchical clustering which creates a tree of clusters and partitioning clustering which creates a flat partition of clusters. Hierarchical Clustering and K-means, the most used partitioning method, will be presented in the next chapter.

According to the way the algorithm processes the documents, the algorithms can be classified in batch algorithms if the document set is processed together, eventually with several iterations, gradually improving an evaluation function, and incremental algorithms which don't need that all documents were present at the beginning and process one document at a time (document by document), without the need of reprocessing the previous documents. The most of the actual document clustering systems are batch but we do believe that incremental systems will become more and more important in this globally connected world.

Evaluation of clustering

What is a good clustering? We can evaluate clustering quality in many different ways, and the performance of clustering algorithms can vary substantially depending on which

criterion is used. However, if one algorithm performs consistently better than other clustering algorithms on many of these measures, then we can say that it is the best clustering algorithm for that situation.

For clustering, two types of measures of cluster quality are used. One type allows the comparison of clustering without any external knowledge and is called an internal quality measure. The other type of measures evaluates how well the clustering is working by comparing the groups produced by the clustering techniques to externally known classes - external quality measure.

Internal Quality Measures.

F-measure

Another external measure is F-measure, a single measure that is a trade-off between precision and recall, two notions from information retrieval (van Rijsbergen, 1979). We can treat each cluster as the result of a query (retrieved documents) and each class as the desired set of documents for this query (relevant documents).

	Retrieved	Not retrieved
Relevant	TP	FN
Not relevant	FP	TN

(Possible results of a query)

In the same vein, the Precision is the fraction of retrieved documents that are relevant (or the fraction of members of cluster j that are members of class

$$\text{Precision}(i, j) = nij = TPnj\ TP + FP$$

where ni is the number of documents in category i, nj is the number of documents in cluster

j and nij is the number of members of class i in cluster j. F-measure is given by the harmonic mean of precision and recall:

$$2 \times \text{Precision}(i, j) \times \text{Recall}(i, j) \text{ Precision}(i, j) + \text{Recall}(i, j)$$

The harmonic mean is always less than or equal to the arithmetic and geometric means. When the values of precision and recall are much different, the harmonic mean is closer to their minimum than to their arithmetic mean.

Hierarchical Clustering

Hierarchical document clustering organizes clusters into a tree, with a single, all-inclusive cluster at the top and singleton clusters of individual points at the bottom. Each intermediate level can be viewed as combining two clusters from the next lower level (or splitting a cluster from the next higher level). The result of a hierarchical clustering algorithm can be graphically displayed as a dendrogram.

Hierarchical methods can be divided, depending on the direction of tree construction, in divisive (start with a node with all documents and iteratively splits it in a top-down manner until only singleton nodes remain) and agglomerative (start with a node for each document and, and each step, join the most similar pair of clusters, in a bottom-up fashion, until the most general node with all documents is created).

Hierarchical Agglomerative Clustering algorithm

One popular approach is hierarchical agglomerative clustering (HAC). This method builds the hierarchy bottom-up, by iteratively computing the similarity between all pairs of clusters and then merging the two most similar. Different variations may employ different similarity measuring schemes.

Hierarchical methods usually suffer from their inability to perform adjustment once a merge or split has been performed. This inflexibility often lowers the clustering accuracy.

Furthermore, due to the complexity of computing the similarity between every pair of clusters these methods are not scalable for handling large data sets in document clustering (Fung, Wang, & Ester, 2003).

In contrast to other techniques, in hierarchical clustering we don't need to specify the number of desired clusters. If we want a different number of clusters we can cut the tree at an intermediate level. Hierarchical methods do not scale well: time complexity is O(n2), where n is the number of objects.

The main difference between hierarchical schemes (besides direction of tree building) is how they choose which clusters to merge, i.e., how they choose to define cluster similarity. There are several options: Single-link - In single-link (or nearest neighbor) clustering, the distance between two clusters is set equal to the distance of the two closest neighbors in the different clusters. This rule produces clusters (sometimes too) elongated clusters.

$$sim(ci, cj) = x \in mci, iyn \in cj \; sim(x, y)$$

Complete-link - In complete link (or furthest neighbor) clustering, the distance between two clusters is determined by the greatest distance between any two objects in the different clusters. This method tends to create compact clusters.

It has been proved that UPGMA is the best performing hierarchical technique (Steinbach et al., 2000). However, when applied to document clustering, these hierarchical methods do not perform well because of the nature of documents, i.e., documents share "core" vocabularies and nearest neighbors of documents often belong to different classes. This causes agglomerative hierarchical clustering techniques to make mistakes that cannot be fixed later, since these techniques don't have backtracking operators.

Partitioning Clustering: k-manner

K-method (Steinbach et al., 2000) is the most extensively used clustering set of rules due to its simplicity and performance. Extra than 50 years after its advent it is nonetheless

considered one of the ten most crucial algorithms in facts Mining (Wu & Kumar, 2009). It's miles a distance-based clustering algorithm that builds a flat (non-hierarchical) partition of ok clusters. The variety of clusters is a person predefined parameter.

Frequent object set-primarily based Clustering

A specific method to information clustering is the usage of common item sets delivered in (okay. Wang, C. Xu, & Liu, 1999) and tailored some years later to text clustering in HFTC system (Beil, Ester, & X. Xu, 2002). By way of treating a report as a transaction and phrases as items, we can use the instinct (borrowed from other areas of information mining) that many frequent items must be shared within a cluster. A frequent itemset is a hard and fast of phrases that seem collectively in greater than a certain quantity of documents.

Common Itemset-primarily based Hierarchical Clustering (FIHC) is the maximum famous algorithm based totally on frequent item sets (Fung, 2002; Fung, ok. Wang, & Ester, 2003). It is an hierarchical and non-incremental algorithm and begins through coming across the present common item sets. Then, it creates a cluster for every frequent itemset and vicinity each file inside the best matching preliminary cluster. The set of clusters may be considered as a fixed of subjects inside the document set. In a 2d step, the algorithm builds the cluster tree and subsequently prune the tree in case there are too many clusters.

Conceptual Clustering

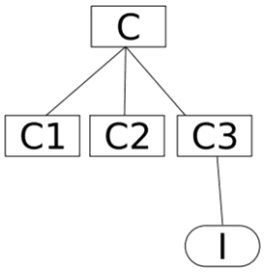
A one-of-a-kind technique is conceptual clustering. Those methods are incremental and construct a hierarchy of probabilistic concepts. COBWEB and its successor CLASSIT are the maximum splendid amongst them. Not like conventional hierarchical techniques (that use similarity measures) they use category application because the cluster best degree.

COBWEB

COBWEB (Fisher, 1987) is an algorithm that incrementally cluster statistics with nominal attributes into concept hierarchies. It creates a complete concept hierarchy (class tree) with each leaf representing a single object and the basis containing all items. Each node

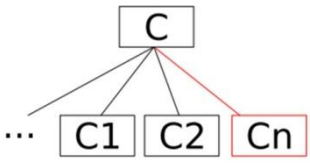
is a probabilistic idea which stores probability of being matched and, for every characteristic, the conditional chance of every feasible cost. COBWEB only supports items described through nominal characteristic-price pairs. Inter-magnificence dissimilarity: percentAi=Vij) - the bigger this possibility, the fewer the objects that share this fee Vij and the extra predictive the price is of class Ck. So this chance measures the predictiveness of the elegance given the value of the characteristic.

While classifying an item, the COBWEB set of rules considers, at each node C, 4 feasible operations, and selects the only that yields the highest CU characteristic fee: Classify the object in an current cluster - The algorithm evaluates the CU value of location the item in each of the kids of C and selects the one with the best rating.



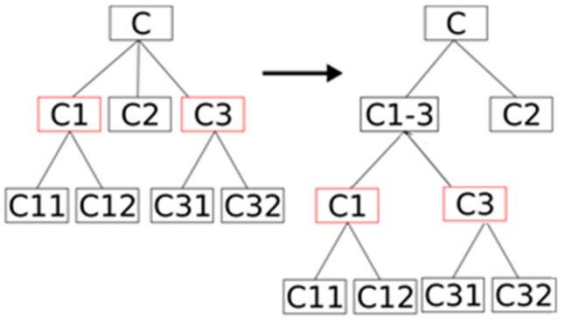
Place the instance in a child of C

Create a new cluster – Next, the algorithm considers creating a new cluster specifically for the new object.



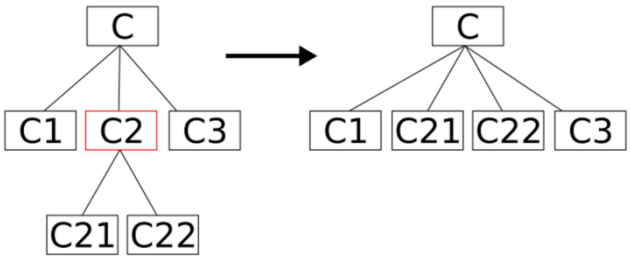
Creating a new cluster

Merge - The COBWEB algorithm considers combining the two clusters with the highest and second highest scores into a new cluster.



Operator Merge

Split – Finally, the COBWEB algorithm considers dividing a cluster with the highest score in its children.



Incremental and hierarchical structures

A few work in incremental textual content clustering has been achieved as a part of topic Detection and tracking (TDT) initiative. The TDT is a challenge to have a look at the detection and tracking of recent occasions in movement of news broadcast. Among its fundamental undertaking is on-line new event detection that includes incremental clustering (Allan, Papka, & Lavrenko, 1998). Most of TFT experiments with record clustering produce flat walls and use the time stamp connected to every textual content, the usage of decaying functions to reduce the significance of older stories and produce non-hierarchical clustering solutions.

As some distance as we realize, the first usage of conceptual clustering algorithms in

incremental and hierarchical record clustering became (Sahoo, 2006; Sahoo, Callan, Krishnan, Duncan, & Padman, 2006). This work commenced displaying why CLASSIT, without modifications, is fallacious for textual content clustering. CLASSIT makes use of numeric attributes and assumes that values of each characteristic follow a regular distribution. This is a valid assumption for widespread actual valued attributes but not for phrase occurrences in textual content that are non-bad integer counts and are a long way from normally allotted.

Paintings Plan

The author of this thought enrolled in 2006/2007 inside the Doctoral software in statistics technology and era of the department of Informatics Engineering of the college of Sciences and technology of the college of Coimbra. In that yr, the candidate finished 3 courses: superior subjects on synthetic Intelligence (18), data (16) and superior topics on Ubiquitous Computing (12). It has to be stated that, on the time, the candidate changed into now not dedicated totally to research, in view that he changed into coaching on the Polytechnic Institute of Guarda, and he became not granted any unique conditions to wait the Doctoral program (on the contrary!).

A few expert and private contingencies prevented the comply with-up of the Doctoral program. In September 2010 he resumed it, beneath the regime of different dedication, when you consider that his coaching activities have temporarily come to a halt. This yr the candidate completed already the missing route, advanced subjects on Cognitive Modeling (18), and has written this concept. This bankruptcy provides the work plan for the PhD program. It starts by way of reporting a few paintings made until now and follows describing the plan for the reminder research. Eventually, we list the primary goal for the e-book of our work.

Modern-day paintings

The author of this notion has achieved already some work in Conceptual Clustering area inside the final task of his Bachelor diploma (Encarnação & Marques, 1993). The

paintings consisted in the creation of a set of rules of Conceptual information Clustering (Cob Fusion) that supported objects represented by means of (weighted) attributes nominal or numerical.

This is the work started out with a bibliographic research on information clustering and statistics mining that enabled the candidate to make a survey of current clustering algorithms and also take a look at some regions and techniques that may be used in report clustering. Subsequent, the bibliographic studies targeted on report clustering. This research revealed the most recent advances in the area and enabled the identification of some techniques and thoughts of information clustering algorithms now not yet carried out to document clustering.

Then the candidate has been conducting a comparative study of the principal algorithms of report clustering, focusing both on the troubles and blessings of each technique and, at same time, has been checking and testing which functions may be used on this greater particular area of incremental file clustering.

Conclusions

This research challenge is an answer to the growing call for automated record enterprise equipment. Report clustering has been used in lots of conditions, many of them simply as a education step to different packages.

On this study, we want to create a brand-new document clustering utility combining capabilities seldom used in conjunction: incremental and hierarchical. To make this possible we will ought to adapt some of the standard techniques in text Mining to incremental surroundings. On the identical time we can seek for an answer for some unsolved questions in incremental documental clustering. Amongst them we spotlight record representation and dimensionality reduction troubles

We additionally pressure the purpose to completely examine our system, comparing it

to present ones and make experimentation with the nicely-hooked up corpora of text mining. We assume that our paintings could be important to create a framework for future incremental and hierarchical file clustering studies.

In the end, we do believe that our machine may be an invaluable tool to prevent us from staying crushed with documents.

References

- Beil, F., Ester, M., & Xu, X. (2002). Frequent time period-primarily based text Clustering. Court cases of the 8th ACM SIGKDD international conference on understanding Discovery and information Mining - KDD'02 (pp. 436- 442). Edmonton, Canada: ACM Press. Doi:10.1145/775047.775110
- Callan, J. (1996). Document filtering with inference networks. Lawsuits of the nineteenth annual worldwide ACM SIGIR convention on research and development in data retrieval (pp. 262–269). ACM. Retrieved from http://portal.Acm.Org/citation.Cfm?Identity=243273
- Encarnação, R., & Marques, P. G. (1993). COBFUSION - Algoritmo de Clustering Conceptual (p. 177).
- Gennari, J. H., Langley, P., & Fisher, D. H. (1989). Models of Incremental concept Formation journal of artificial Intelligence, forty, eleven-61.
- Gluck, M. A., & Corter, J. E. (1985). Information, uncertainty and the utility of classes. Proceedings of the seventh Annual convention of the Cognitive technological know-how Society (pp. 283–287). Irvine, CA, united states.
- Katz, S. M. (1996). Distribution of content material phrases and terms in textual content and language modelling natural Language Engineering, 2(1), 15-59.
- Manning, C. D., & Schütze, H. (2000). Foundations of Statistical natural Language Processing (p. 704).

- Cambridge, England: The MIT Press.
- Milios, E. E., Shafiei, M. M., Wang, S., Zhang, R., Tang, B., & Tougas, J. (2004). A scientific have a look at on file representation and Dimensionality reduction for textual content Clustering, 1-29.
- Retrieved from http://www.Cs.Usask.Ca/~spiteri/DR_Proj_ver04.Pdf
- V an Rijsbergen, C. J. (1979). Records Retrieval (second ed., Vol. Thirteen, p. 153). London:
- Rosell, M. (2006). Creation to information Retrieval and text Clustering (pp. 1-25).
- Sahoo, N. (2006). Incremental Hierarchical Clustering of text documents. ACM Press, ny, the big apple, america.
- Engineering Workshop, 770-779. Ieee. Doi:10.1109/ICDEW.2007.4401066