

Online Evolving Clustering of Web Documents

Anthony Evans, Plamen Angelov, and Xiaowei Zhou
Lancaster University
Lancaster, LA1 4WA United Kingdom
Phone: +44 (0)1524-510398, Fax: +44 (0)1524-510493
email: {a.d.evans, p.angelov, x.zhou}@lancs.ac.uk

ABSTRACT: In this paper an approach that is using evolving, incremental (on-line) clustering to automatically group relevant Web-based documents is proposed. It is centred on a recently introduced evolving fuzzy rule-based clustering approach and borrows heavily from the Nature in the sense that it is evolution-inspired. That is, the structure of the clusters and their number is not predefined, but it self-develops, they grow and shrink when new web documents are accessed. Existing Web-based search engine technology returns long lists of web pages that contain the user's search term query but are not presented in order of contextual similarity. For example, search terms that have more than one meaning such as "cold" are presented to the user in a list containing documents relating to the Cold War and the common cold. If these results could be clustered "on the fly" then this improved presentation of results would allow the end user to find relevant documents more easily by requiring the inspection of one cluster of contextually similar documents rather than entire list of documents containing information pertaining to irrelevant contexts. An issue that is paid a special attention to is the similarity measure between the textual documents. Euclidean, Levenstein, and Cosine similarity measures have been used with cosine dissimilarity/distance performing best and addressing the problem of different number of features in each document. The proposed evolving classifier has also learning capability – it improves the result on-line with any new document that has been accessed. Finally, the proposed approach is characterized by low complexity. This paper reports the results of research that is going on for more than two years at Lancaster University on development of a novel clustering method that is suitable for real-time implementations. It is based on evolution principles and tries to address the limitations of existing clustering algorithms which cannot cope in an online mode with high dimensional datasets. This evolution-inspired and Nature-inspired approach introduces the new concept of potential values which describes the fitness of a new sample (web document) to be the prototype of a new cluster without the need to store each previously encountered documents but taking into account the contextual similarity density between all previous documents in a recursive and thus computationally efficient way (thereby reducing memory requirements and improving speed compared with existing approaches). This paper examines the clustering of documents by contextual similarity using extracted keywords represented in a vector space model.

KEYWORDS: Clustering, Information Retrieval, Data Mining, Incremental Clustering, On-line Clustering

INTRODUCTION

Dramatic developments in Internet and information systems lead to the situation when we are '*drowning in information and starving for knowledge*' according to the expression by R. D. Roger. Despite the existence of huge amount of information on the Web its use is still hampered by the fact that it is disorganized and often irrelevant to our aims and objectives. One approach to cope with this problem is the Semantic Web, but it requires a bottom-up work on rebuilding the monstrous amount of web pages that already exists. An alternative is to 'get the best out of the current situation'. The evolving on-line clustering method developed in this paper can be considered as a technique associated to the latter approach when we try to extract useful knowledge of the data stream represented by the Web documents [1,2].

Cluster analysis involves a number of data analysis algorithms and techniques for grouping similar objects into categories so that those within a cluster have greater similarity to each other than objects outside that cluster [3]. It is used to find structure within an unlabelled dataset to reduce dimensionality and has been applied in Information Retrieval applications such as SMART [4] which was one of the first information retrieval research project that introduced the vector space model of representing documents in the 1960s. Contextual clustering is the partitioning of a dataset of contextually unordered documents into subsets so that data in each subset have similar traits [5]. Before a clustering algorithm can be applied, the contextually important keywords need to be extracted by means of regular expressions and noise or irrelevant words need to be filtered by means of a stop list [6]. The document is then transformed into a numerical representation using the vector space model approach [7] which represents each document

numerically according to the number of times a word appears in the document. The similarity measure is determined by the inner product of the document vectors where word overlap indicates similarity. Frequency information is used in the Euclidean, Levenstein [8], or 'cosine' distance measures [13] to apply weights to words so that a word that occurs frequently in all documents is given a lower weight than words that appear frequently in one document only. Normalisation of the sum of weighted words is applied so that very large documents do not skew the results of small documents.

The words in a document that do not describe the content are function words such as conjunctions. The so-called 'tf-idf' method [9] (a modified cosine distance) is used in such a way that the words that occur frequently in most documents are given lower weight. Therefore, this distance measure automatically filters out such noise. However this is not efficient enough and therefore so called 'stop list' is used to remove function words from all documents. The stop list used in this paper contains a list of all function words in English. The manner in which keywords are extracted from text is performed through regular expressions and is language independent.

An important issue in any clustering method is the measure of distance or dissimilarity between the documents to be clustered. The Euclidean distance between two values is their vector difference in so called Euclidean space. The disadvantage of Euclidean distance is that it is influenced by variables that have the largest values. Normalising the distance to a value between 0 and 1 is therefore used to prevent this. Cosine similarity takes into account the weight of words in a document, which is determined by the frequency of occurrence. The weight of the word in a document is determined by the tf-idf formulation (term frequency - inverse document frequency) which gives a high weight to words that have a high frequency value in the document and a low occurrence in the whole collection of documents. Unlike Euclidean distance, cosine distance ensures that only words shared by the compared documents are considered (the weight of a word is zero if it does not appear in a document). The cosine of the weights of keywords and their frequency determines the similarity between documents where a cosine value of 0 means that the documents are orthogonal and 1 means that the documents are identical. In [13] cosine distance is used with self-organizing maps (SOM) for clustering web-based documents off-line.

Recently, approaches for on-line clustering textual documents [14,15] have been developed. In [14] Bayesian approach is taken and an assumption about the posterior probability is made which does not hold always. In [15] the authors use subtractive clustering, but they rely on update of the centres/prototypes only and ignore the contextual similarity information contained in all the documents. In the approach that is proposed in this paper contextual similarity to all documents is used without storing them or the information about each one of them.

The software realization of the proposed approach (*SmartSearch*, available at www.onlineclustering.com) is intended to be an add-on to existing search engines and this is achieved by using web services which work using requests and response messages using the RPC (Remote Procedure Call) pattern, except that data is transferred in XML over HTTP instead of binary (as in Java RPC). Web Services allow programmers to create add-ons to existing web based applications in a programming language independent manner. The "Google API®" and "Alexa Web Information Service®" were used. Note, that their APIs are similar. Alexa® which uses REST (Representational State Transfer) was found to be slightly faster than Google because Google® requires a messaging layer on top of the XML called SOAP (Simple Object Access Protocol). This was considered important for the high dimensionality aimed by the application.

THE PROPOSED APPROACH

A novel approach to clustering the data streams in real-time was recently developed [10,11]. It has also been applied for classification problems [12]. In this paper we develop further this approach and apply it to clustering Web-based documents in on-line mode with potential to be implemented in real-time. The proposed online evolving clustering method has the following specific features that make it suitable for this task:

High dimensionality: The algorithm is able to cope with thousands of samples (Web-based documents) without the need to store all the information in memory.

Unsupervised: The algorithm is able to divide a dataset into subsets without requiring initial training data. The rule base is flexible and evolving in the sense that system behaviour is improved by storing the useful previous knowledge. The knowledge is constantly updated as more data is received.

Online: The algorithm is suitable for continuously clustering new data due to its recursive nature. Offline clustering methods typically have a non evolving rule base and are non recursive. Thus they are suitable for a fixed set of data supplied in batch mode. However when the training data is collected continuously, some of the new data will reinforce and confirm the information contained in the previous data sufficient to modify the rule base.

The evolving Clustering (eClustering) method that is used is a prototype-based [10,12]. Fuzzy rules are generated on-line around prototype Web Documents, D . A vector containing the frequencies, F of occurrence of each 'keyword' in the meaningful content of the Document frequency list, L forms the input to eClustering. Note that not only the frequency of each keyword, but also the list of keywords for each web document can be different.

$$L_k = [F_1^k, F_2^k, \dots, F_n^k] \quad (1)$$

The rules generated by eClustering are of the following form:

$$R^i : IF(L^k \text{ is similar to } L^*) THEN(Group D^k \text{ together with } D^{i*}) \quad (2)$$

where R^i denotes the i^{th} fuzzy rule; $i=[1,N]$; N is the number of fuzzy rules.

Fuzzy membership functions of Gaussian type are formed for each 'keyword' of the prototypes, L^* , $[F_1^*, F_2^*, \dots, F_n^*]$.

$$\tau_k^i = e^{-\sum_{j=1}^n \left(\frac{d_{\cos, ik}^j}{\sigma_j^i} \right)^2} \quad i=[1,N] \quad (3)$$

where $d_{\cos, ik}$ is the dissimilarity between a web document and the prototype of the i^{th} group (the focal point of the fuzzy rule); σ^i is the spread of the membership function.

We use so called 'cosine distance' as a measure for dissimilarity between the frequency lists, L [3]:

$$d_{\cos} = 1 - \frac{\sum_{j=1}^n F_j^k F_j^i}{\sqrt{\sum_{j=1}^n (F_j^k)^2 \sum_{j=1}^n (F_j^i)^2}} \quad (4)$$

eClustering assumes that the number of content categories is not pre specified, and the number of fuzzy rules evolves during the process of online clustering. *eClustering* starts with an empty rule-base until the first keyword frequency vector is received. The process of evolving clustering is fully unsupervised. *eClustering* is centered on the concept of data spatial density measured by so called 'potential' [10,12]. Each web document is considered a *potential* cluster centre. A document surrounded by documents will have a higher *potential* value. *Potential* is inversely proportional to the dissimilarity between a document and all the previous documents [10, 12]:

$$P_k(L_k) = \frac{1}{1 + \left(\sum_{i=1}^{k-1} d_{\cos}^2(L_k, L_i) \right) / (k-1)}; k = 1, 2, \dots \quad (5)$$

where k is the index of the current document, which also means the number of total web documents that has been considered so far.

To enable the online one-pass ability of the approach, the processed web documents are not stored in the memory and will only be processed once and discarded immediately when processed. This requires all the calculations for the leaning procedure to be recursive to accumulate the history information (*similarity between one to all documents*) without memorizing the history data (*all the documents*). An online clustering procedure starts with the first document established as a prototype for the first cluster. Its frequency list is used to form the first fuzzy rule. Starting from the next data document onwards the potential of the new documents is calculated recursively by [12]:

$$P_k(L_k) = \frac{1}{2 - \frac{1}{\sqrt{\sum_{i=1}^n (F_k^i)^2}} \sum_{i=1}^n F_k^i a_{k-1}^i}; P_1(L_1) = 1 \quad (6)$$

$$\text{where } a_k^i = a_{k-1}^i + \frac{(F_k^i)^2}{\sqrt{\sum_{l=1}^n (F_k^l)^2}}; a_1^i = \frac{(F_1^i)^2}{\sqrt{\sum_{l=1}^n (F_1^l)^2}}; i = [1, n] \quad (7)$$

In this way, the spatial density at each new document, P_k in respect to **all** previous documents can be recursively calculated using n accumulated values in a single auxiliary variable. This makes possible learning the information of contextual density representing the whole previous history which is the distinctive feature of the proposed algorithm.

Density in the global data space is affected whenever a new web document enters the space; therefore the potentials of all existing prototype keyword frequency vector needs to be updated. This update is also done in a recursive way:

$$P_k(L^*) = \frac{(k-1)P_{k-1}(L^*)}{k-2+P_{k-1}(L^*)+P_{k-1}(L^*)d(L^*,L_k)} \quad (8)$$

Updating of (8) is only based on the current inputs and the current potential $P_{k-1}(L^*)$, therefore, no additional variable needs to be stored/memorized.

The proposed evolving web documents clustering approach starts ‘*from scratch*’, without any pre-defined prototype. Frequency vector of each document are used to upgrade the rule-base. Potential of each document, $P_k(L_k)$ is updated recursively by (6)-(7) from the second document received and onwards. The potential of all existing prototypes, $P_k(L^*)$ is also updated using (8). Comparing the potential of a new document with the potential of each of the existing prototypes:

$$P_k(L_k) < \min(P_k(L^*)); \quad (9)$$

If condition (9) occurs that means the new document is very distinctive and potentially a new group can be formed around it as prototype. We also check whether any of the already existing prototypes is similar enough to (also means well described by) the new prototype. The degree of the membership is therefore tested:

$$\exists i, i = [1, N]; \quad \tau_k^j > 1/3 \quad \forall j, j = [1, n] \quad (10)$$

If both (9) and (10) stands, which means the new web documents is a candidate prototype and very similar to the existing prototype web document, therefore, it can replace this similar prototype; otherwise, if only (9) is true, a new group is formed with the new web document as the prototype.

The procedure of eClustering can be described by the following pseudo code:

BEGIN eClass

```

Initialize (get the first document, D1; define the keyword frequencies, F1
and generate its list L1; initialize the rule base and parameters);
DO for any next document, Fk, k=2,3,...
  Pre-process the document, Extract frequency list, Lk;
  /*---Evolve Rule-base---*/
  Calculate Pk(Lk) using (6)–(7);
  Update P(L*) using (8)
  Compare the potential of new document and that of the existing
  prototypes (9)
  IF ((9)) THEN add a new prototype formed around the new document, Dk;
  IF ((10)) THEN remove the prototype(s), Li* for which this holds;
WHILE More documents are available
END DO

```

EXPERIMENTAL STUDY



(Fig 1)

We have performed extensive experiments and development of the software application, *SmartSearch* (fig 1) available at www.onlineclustering.com. In order to compare the results we have used four other previously existing approaches to clustering Web-based documents, namely:

- Offline matrix based approach (clumping) comparing all pair wise distances with Euclidean distance
- Online agglomerative approach with Cosine distance measures
- Offline kNN where K = 4, using Euclidean distance measures.
- Subjective clustering.

The software application *SmartSearch* that was developed at Lancaster University is a server side PHP application with a web browser based front end. The user is able to choose between the matrix, *kNN*, agglomerative and evolving methods to cluster web documents. The tests were based around three search terms; "Lancaster" (table 1), "Ball" (table 2) and "Cold" (table 3). The success of the results was determined by the total time taken from submitting the query to receiving the results, and the minimum number of clusters formed. Manual subjective clustering was also compared. The results are given in the tables below for grouping 30 documents for each query. Stemming was applied to each word. The full text of the document was used. Note, that alternative faster strategies such as using the snippet or titles only or neighbouring terms [8] are also possible.

	Clumping	Agglomerative	kNN (k=4)	eClustering	Subjective clustering
No of clusters	21	15	4	6	5
Online/Offline	Offline	Online	Offline	Online	Offline
Total Time, s	~30	~ 3	~ 1 per iteration	0.75	> 1000
Threshold	0.5	0.80	NO	NO	NO
Notes/Clusters formed	Significant cluster overlap on all clusters.	Universities, Businesses City of Lancaster Pennsylvania, USA (10 noise clusters)	University, Lancaster city, Businesses, Newspapers	Governments, Educations, Tourism, Businesses	Tourism Local government, Companies, America, News & University

Table 1: Search term = 'Lancaster'

	Clumping	Agglomerative	kNN (k=4)	eClustering	Subjective
No of clusters	15	15	4	5	4
Online/Offline	Offline	Online	Offline	Online	Offline
Total Time, s	~30	~ 3	~ 1 per iteration	0.73	> 1000
Threshold	0.5	0.90	NO	NO	NO
Notes/Clusters formed	Significant cluster overlap on all clusters.	Games, News, People (11 noise clusters)	Ball games, Companies, News, People.	Ball Games Company University	Companies University People Games.

Table 2 : Search term = "Ball"

	Clumping	Agglomerative	kNN (k=4)	eClustering	Subjective
No of clusters	13	15	4	2	5
Online/Offline	Offline	Online	Offline	Online	Offline
Total Time, s	~30	~ 3	~ 1 per iteration	0.42	> 1000
Threshold	0.5	0.85	NO	NO	NO
Notes/Clusters formed	Cold war Music Cold fusion (10 noisy clusters)	Cold war, common cold (13 noise clusters)	Common cold, Cold fusion, Weather, Cold war	Common cold Cold war	Cold fusion Cold weather Common Cold Cold war Music

Table 3 : Search term = "Cold"

RESULTS ANALYSIS AND CONCLUSIONS

The matrix based approach was the least efficient technique and its use of un-weighted keywords resulted in significant cluster overlap. In the agglomerative method, the order in which the documents were presented affected the final clustering and many atomic (noise) clusters were formed. The *kNN* approach required all the training data to be available beforehand. Finally, clumping and agglomerative clustering methods required a predefined threshold value which problem specific.

To automatically group an unknown number of the web documents online, without predefining the clusters, a novel approach called *eClustering* has been developed and implemented as a software on-line application called *SmartSearch*. The proposed approach is a fully unsupervised fuzzy clustering approach that process the web documents incrementally (in one pass), which avoids memorizing the history data and thus makes possible to apply the approach in on-line. No predefined subject-specific threshold is required for the generation of class or prototypes. So called 'cosine'

distance/dissimilarity measure is used, which has advantage in measuring input vectors that consist of discrete values such as 'number of appearance of a word in the text'. Additionally, the length of the feature vectors can be different using cosine distance, which is not the case with Euclidean distance.

The proposed novel approach will revolutionise the next generation of search engines and will also be useful in data analysis to automatically discover the nature or structure of the data such as finding groups of similar documents, their spatial density, and dissimilarity, list of key words per class/group that forms the semantic value of the class etc. Application to clustering desk-based documents, e-mail messages etc. is also under intensive consideration and will constitute a further step of investigations.

REFERENCES

- [1] G. Fayyad, P. Shapiro, P. Smyth, 1996, "From Data Mining to Knowledge Discovery: An Overview, Advances in Knowledge Discovery and Data Mining", MIT Press.
- [2] P. Domingos and G. Hulten, 2001, "Catching Up with the Data: Research Issues in Mining Data Streams, Workshop on Research Issues in Data Mining and Knowledge Discovery", Santa Barbara, CA.
- [3] R. O. Duda, P.E. Hart, and D.G. Stork, 2000, "Pattern Classification", 2nd Edition. John Wiley, Chichester, UK.
- [4] G. Salton and M.E. Lesk, 1965, "The Smart Automatic Retrieval System An Illustration", ACM, 8:6
- [5] T. A. Runkler and J. C. Bezdek, 2003, "Web mining with relational clustering", International Journal of Approximate Reasoning, 32(2-3), pp.217-236.
- [6] C. J. Fox, 1990, "A Stop List for General Text. SIGIR Forum", 24(1-2): pp.19-35.
- [7] G. Salton, A. Wong, and C. S. Yang, 1975, "A vector space model for automatic indexing", ACM, 18(11).
- [8] P. Angelov and T. Evans, 2004, "Semantic categorization of web-based documents", in Proc. 5th Int. Conf. Recent Advances in Soft Computing (RASC), Nottingham, UK, pp.500-505.
- [9] D. Hiemstra, 2004, "A probabilistic justification for using tf×idf term weighting in information retrieval" in International Journal on Digital Libraries, 3(2), pp.131-139.
- [10] P. Angelov, 2004, "An approach for fuzzy rule-base adaptation using on-line clustering," Int. Journal of Approximate Reasoning, 35(3), pp. 275-289.
- [11] P. Angelov, X. Zhou, 2006, "Evolving Fuzzy Systems from Data Streams in Real-Time", 2006 International Symposium on Evolving Fuzzy Systems, Ambleside, UK, IEEE Press, pp.26-32.
- [12] P. Angelov, X. Zhou, F. Klawonn, 2007, "Evolving Fuzzy Rule-based Classifiers", IEEE Intern. Conf. on Computational Intelligence Applications for Signal and Image Processing, April 1-5, Honolulu, Hawaii, USA.
- [13] B. Yang and W. Song, 2005, "A SOM-based web text clustering approach", in Proc. of IFSA, pp. 618-621.
- [14] K. M. A. Chai, H. T. Ng, H. L. Chieu, 2002, "Bayesian Online Classifiers for Text Classification and Filtering", in Proc. SIGIR'02, Aug. 11-15, pp.97-104, Tampere, Finland.
- [15] B. S. Suryavanshi, N. Shiri, and S. P. Mudur, 2005, "Incremental relational fuzzy subtractive clustering for dynamic web usage profiling", in Proc. WebKDD, Chicago.