An Architecture for Efficient Document Clustering and Retrieval on a Dynamic Collection of Newspaper Texts

Alan F. Smeaton, Mark Burnett, Francis Crimmins and Gerard Quinn

School of Computer Applications
Dublin City University
Glasnevin, Dublin 9, IRELAND
asmeaton@compapp.dcu.ie

Abstract

Clustering of related or similar objects has long been regarded as a potentially useful contribution to helping users navigate an information space such as a document collection. When documents are related by virtue of being about the same or similar topics, then this is often a good indicator that they will be relevant to the same queries and this can be used during the retrieval operation. Many clustering algorithms and techniques have been developed and implemented since the earliest days of computational information retrieval but as the sizes of document collections have grown these techniques have not been scaled to large collections because of their computational overhead. In this paper we describe a technique for clustering a collection of documents such as a collection of online newspapers which uses a number of short-cuts to make the process computable for large collections. Furthermore, our design is extensible in that it caters for a dynamic collection of documents which would be periodically, perhaps nightly, updated, amended or have deletions. An implementation of the clustering on an archive of the *Irish Times* newspaper is reported here.

1. Introduction

Document clustering is a technique for identifying clusters or groups of documents which share some common features or have overlapping content. These groupings of documents can be useful in document retrieval as much previous work in information retrieval has shown. Prior work has developed many different document clustering algorithms and techniques which have been shown to make a useful contribution to retrieval [1,2,3].

One of the major problems with document clustering is the computational overhead of creating the clusters which usually requires the creation of an NxN similarity matrix where N is the number of documents and each entry in this matrix is the pairwise similarity between documents. As document collections have grown in size this bottleneck has prevented a more widespread deployment of clustering in information retrieval. In this paper we report progress in developing a cluster-based search for an archive of newspaper articles where the clustering process is implemented in an efficient way. This is achieved by applying several thresholds within the cluster generation process and yields a clustering architecture that is scaleable to large document archives. One of the most important aspects of our system architecture and implementation is that it can cater for dynamic updates and additions to the document archive and still re-generate clusters in reasonable time. Such a requirement is central to the chief application of our work, searching through an archive of online newspaper articles.

The remainder of this paper is organised as follows. In the next section we motivate the use of document clustering in a newspaper searching environment, in our case the *Irish Times* online newspaper which we have used in our initial experiments [4]. Our system architecture is presented in section 3 and in section 4 we describe a retrieval system we have developed which uses these clusters. Related work is presented in section 5 and plans for future work and extensions are described in section 6.

2. Searching an Archive of Newspaper Stories

There are now well over 1,000 daily newspapers which have a presence on the web and most of these provide an online version of their printed form. The reasons for a newspaper doing this seem to be more for investment and prestige than for revenue generation. The vast majority of online newspapers record a net loss for their online versions but they still do this in order to have an early presence within the emerging online market and it is perceived that waiting until money can be made before launching an online newspaper would be too late. Reasons for attracting readers to a given newspaper site would include the reputation or customer loyalty to the printed version

which is especially true of users reading from abroad, the timeliness and currency of information and a good user interface which permits easy navigation and includes some kind of search facility.

A collection of newspaper articles has interesting characteristics which distinguish it from other document collections. Newspaper articles are normally of the same order of magnitude, varying from a paragraph to a number of columns, though they are not as consistent in size as, say, document abstracts. Newspaper articles are timely meaning they have an importance which is related to their date of publication, i.e. yesterday's news is old news. However the most distinguishing characteristic of newspaper articles for our purpose is that most of them are related to previously and to subsequently published articles. News stories happen and evolve over time. A child is murdered, an investigation is launched, evidence is gathered, a suspect is arrested and charged, the charge is reduced to manslaughter, a trial happens, the jury find the suspect guilty, the judge over-rules the jury and sets the accused free, she writes a book, makes a million dollars and lives happily ever after. Each of these would be reported as a news story and although the Louise Woodward case is probably atypical because it was a huge story and happened over a long period of time, almost all news has this element of a continuum with each individual news story (newspaper, radio or TV) is a snapshot of a story at a given point in time. What this means for a corpus of newspaper stories is that the dependencies between and among stories is potentially huge and these dependencies cannot be ignored when it comes to navigating the archive.

Most online newspapers provide archive searching but only back to the point of the launch of their online version. Articles prior to that point are generally not freely available and it is believed by online newspapers that this represents their potential goldmine. When electronic commerce really arrives we can envisage charged searches through large online archives and search support for article dependencies would be important in encouraging browsing/searching through large archives.

At the time of publishing an online version of a newspaper story, an editor can choose to incorporate hypertext links back to related stories. Thus today's story about the opening of a trial could be linked to earlier stories reporting the arrest of the suspect, the discovery of the body, etc. Many online publications such as HotWired [5] and the Electronic Telegraph [6] newspaper do this but there is a huge overhead in that this must be done manually. In the case of the Electronic Telegraph newspaper, a search is done through the whole archive of past stories to determine what hypertext links are appropriate for each new story published. The disadvantages of this process are twofold; on the one hand there is clearly the huge labour cost. The second cost is due to the fact that because this linking is done at the time of publication of an article all such manually created links must always point backwards in time whereas sometimes when we read an article we want to be able to follow links backwards and forwards in time. One could create hypertext links which would allow 2-way temporal linking but this does not cover cases where stories cross over, merge or split. In our work we have explored document clustering as a technique for generating links between related documents as the collection is updated and presenting these links as a result of a search.

The *Irish Times* newspaper, with whom we work in the project reported here, produces an online version on the web and has done so for about 3 years [4]. This is a duplicate of the printed version in that it contains all the journalistic articles but is minus advertising, classified columns, TV and cinema listings, and so on. Each day, the online version of the news paper is put on the web at about 2 a.m. and consists of about 300 new stores which must be integrated into the clustered search on this archive which we have developed. The system architecture and algorithms for doing this will now be described.

3. System Architecture and Clustering Algorithm

The approach we took in the present work automatically structures or clusters document collections based on their content. A cluster contains two or more documents closely related in content and each cluster is a composite entity with a hierarchical structure that contains sub-clusters. The possible application of this type of document clustering for information seeking that we are interested in is where a standard search mechanism is used to display clusters of documents rather than a single ranked list and in this way present documents potentially relevant to the user's query. Alternative ways of using clusters as an integrated part of the retrieval process are beyond the scope of our present work.

Cluster analysis is a statistical technique used to automatically classify hitherto undifferentiated objects into classes based on attributes of those objects. Although it has had little impact in information retrieval until quite recently, early studies showed that clustering methods are appropriate for document analysis. This is because the multivariate analysis inherent in clustering techniques is well suited to documents, which typically have a wide vocabulary with a number of key concept words occurring in related documents.

One of the reasons clustering analysis has had little impact on IR is the high overhead involved in generating the cluster structures. This was addressed in our work by storing an $N \times k$ similarity matrix for N documents, instead of a full $N \times N$ as had been normal practice. The value for the constant k is a compromise between clustering efficiency and storage (memory) requirements against completeness of the clustering.

In a broader sense document clustering may be viewed as an attempt to automatically assign meta-document descriptors to a document. Typically IR systems use the words and/or keywords of a document to index it. The automatic classification described here allows the use of cluster descriptors to provide a higher level description of a document. These descriptors characterise and summarise the contents of the cluster. Document clustering has long been perceived as having value as a means of automatic classification of documents and has potential for improving the effectiveness of retrieval but the computational effort involved in generating clusters, particularly for large (> 10000) document collections has meant that it is a relatively untested technique.

There are two distinct approaches to cluster analysis: hierarchical and partitioning methods of data classification. Partitioning methods seek to partition the data into a specified number of disjoint groups, often in an attempt to optimise a mathematical criterion. Hierarchical methods produce the inverted tree structures or dendrograms typical of a taxonomy. This type of structure is particularly useful in IR as it allows a document collection to be viewed at different levels of graininess.

In clustering parlance the technique which we eventually decided to use is agglomerative & hierarchical. It is agglomerative in that starting with unclustered data documents join clusters based on their similarity to the members of a cluster. Our method is hierarchical because the resulting cluster is structured hierarchically with closely related items at the leaves of the cluster tree, and less closely related at the root. The method used is called complete-link clustering though during the course of our work we looked at other techniques before deciding on this one [1]. This method uses the smallest similarity within a cluster as the cluster similarity, and every item within the cluster is related to every other with at least the similarity of the cluster. Small, tightly-bound clusters are characteristic of this method which is well-suited to the application of searching a newspaper archive. The more commonly used single link method tends to produce large poorly defined and straggly clusters due to the non-zero word overlap of unrelated documents. This tends to lead to poor classification with every document seemingly related to every other which is undesirable in our application. The complete link clustering which we used is known to produce large numbers of small, tightly bound clusters [2] which we hope would correspond to the large number of real world events reported in a newspaper with some of these reported as a series of individual but related stories.

In order to test the appropriateness of clustering for a document collection on an intranet of documents, we clustered all of the newspaper articles published by the *Irish Times* during a 9-month period in 1993, as our first experiment. This totalled over 100 Mbytes of text for the almost 34,768 individual newspaper articles. The clustering technique we used requires that the similarity of each document with respect to every other is known. We used a conventional retrieval technique with term weighting to compute inter-document similarity which was implemented using an IR system developed by our research group in previous work [7]. This was accomplished by treating every document in turn as a query and using our search engine to rank it against the existing text base of documents. Inter-document similarity then simply maps to the relevance score for the appropriate pair of documents. A symmetric similarity matrix with entries in the range 0.0 - 1.0 results with 1's along the main diagonal. Where there are 100's or 1000's of documents, this is typically a sparse matrix with many zero or near-zero values. Document collections greater in size than about 10,000 pose a problem for this type of analysis in that in addition to the computation costs, the similarity matrix requires $N \times N$ - 1 units of memory per matrix element for storage. If each matrix element requires 4 bytes (a conservative estimate) then 10000 x 10000 x 4 = 400 Mb of memory is needed for 10,000 documents, and that is only 10,000 documents. For 1,000,000 documents the memory required would be 4Tb, which could be handled but which would be unwieldy for the clustering algorithm.

The solution to this problem which we developed is to keep an $N \times k$ matrix, where k is a constant value and $k << N$. This means keeping the top k hits for each document query. For a collection of 35,000 Irish Times news articles clustering was performed with k = 10, 20, 30 and 40. Inspection of the quality of the resulting clusters showed that k = 30 gave results comparable to k = 40, mainly because there are frequently few if any documents that are cluster candidates lower than about 30th place on the search hitlist. This is due to the fact that similarity scores drop rapidly down the hitlist, showing that documents beyond about roughly 30th place typically have little in common with the query document, given that the "document" in this case is actually the full text of an article published in the *Irish Times*. A different scenario would result for a much shorter user query issued to the same document collection. It was found that for the test subset of the 1993 *Irish Times* collection that we initially worked with, the quality of the clusters changed little as k was decreased from N to 30.

When a clustering is generated from an NxN matrix then the complete document collection is clustered and

every document is integrated into some clustering. Clustering continues from small groupings right up to a single root node as shown in Figure 1(a). When an N x *k* similarity matrix is used then small clusters form but these do not get integrated into a single overall clustering. The clustering process yields many small clusters and some unclustered documents which, although inelegant from a clustering standpoint, is exactly what is appropriate for a user examining clusters of related newspaper articles.

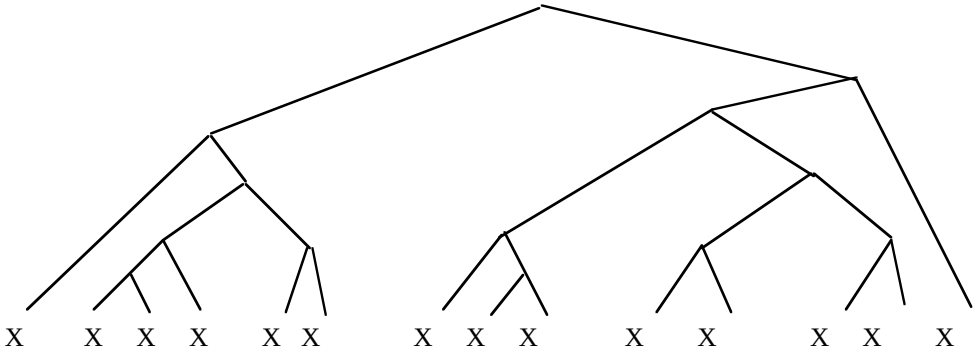


Figure 1(a): A complete clustering based on NxN matrix

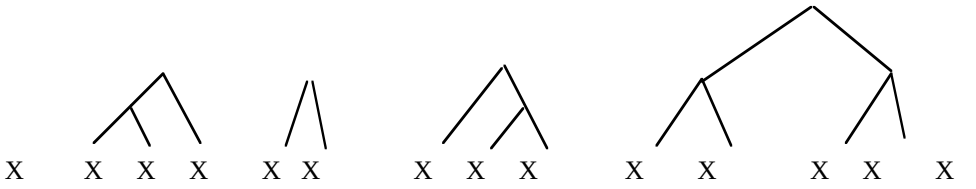


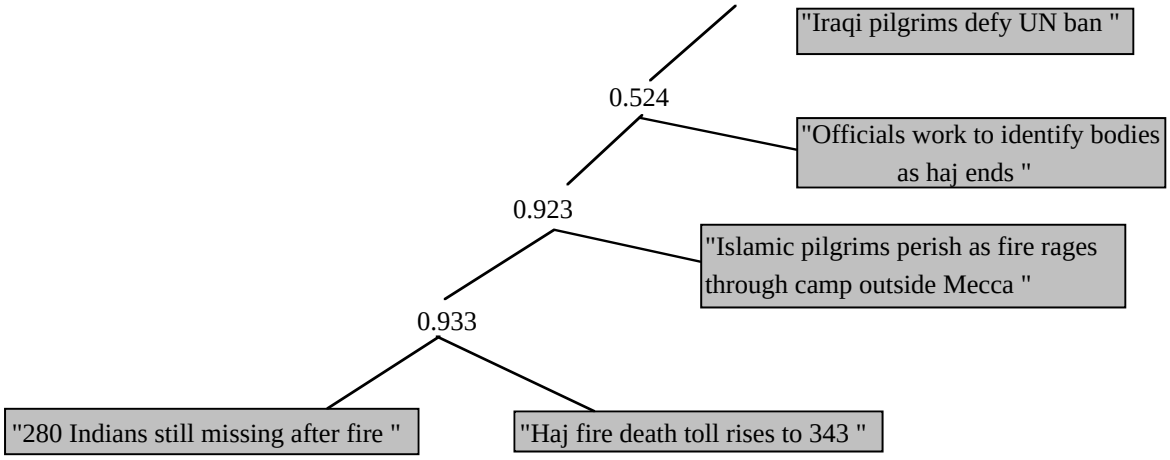
Figure 1(b): A partial clustering based on Nx*k* matrix

The primary data structure for the clustering software is the similarity matrix. A form of this called a cluster index that is suitable for incremental addition was developed. This allows an additional group of documents to be added to an existing archive by reading in the cluster index, running each of the new documents as a query and adding entries to the cluster index, and doing a complete re-cluster (a relatively fast process). This avoids a time-consuming regeneration of the entire cluster index.

The search engine we use for our work here is described in [7] and is novel in that it includes three thresholding techniques designed to reduce the computation time but which do this without loss of retrieval effectiveness. The first thresholding technique, called postings list thresholding operates by processing only some portion of the postings list entry for a given search term. The second technique called query term thresholding, operates by processing only some portion of the search terms, depending on the length of their posting list entries. The third technique creates a reduced set, *DR*, of document accumulators or registers instead of the conventional approach of reserving a document score register for each document in the corpus. When processing entries in the inverted file, only the first *DR* unique document identifiers encountered will actually be assigned similarity scores. When our search engine is used in a conventional IR application in past experiments we have shown that this not only saves a huge amount of computation but does so without loss of retrieval effectiveness [7]. It is thus quite appropriate for the batch processing of each of a set of new stories added to an online corpus. For this work we set the value of *DR* to be 500.

As an example of the kind of clusters we generate, the cluster containing the document entitled "Officials work to identify bodies as haj ends" is:

0.476



The overall similarity of the cluster is 0.476 but the highest score on the hitlist for document "Officials work to identify bodies as haj ends" is 0.221. This means that each of the other documents in the cluster scored below 0.221. With a standard complete-link algorithm the document in question would not have joined the cluster, although it is a good member.

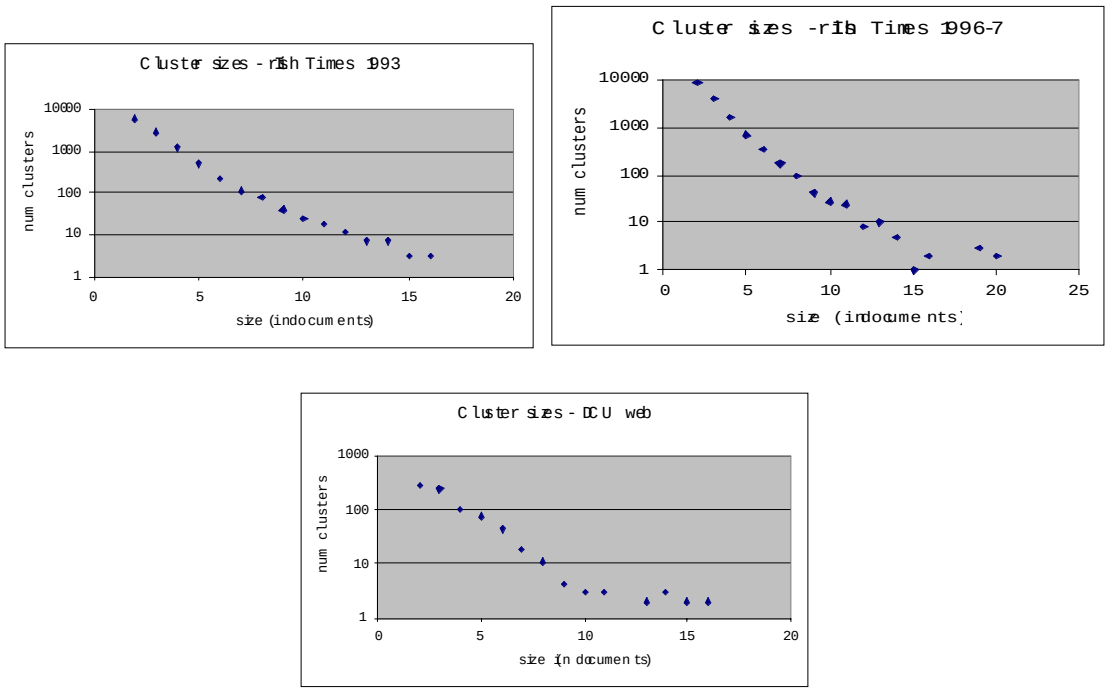
The time taken to complete the clustering process is worth mentioning here. Using a shared SUN UltraSparc with 128 Mbytes RAM but no local disks (accessed over NFS) the entire *Irish Times* collection of almost 35,000 documents was indexed in less than 2 hours, the $N \times k$ similarity matrix was constructed in about 11.75 hours (effectively running 35,000 queries, many of some hundreds of terms in size) and the clusters generated in about 8 minutes. For a daily update of 300 news stories, these can be indexed and added to the overall inverted file in about 15 seconds, each new document can be added to the reduced similarity matrix by running it as a query and updating all matrix entries at a rate of almost 50 documents per minute, and the computation of the clustering takes about 8 minutes. This totals less than 15 minutes of processing time.

In order to assure that the techniques we were developing for clustering document collections were not being tailored to the dataset in question we applied our clustering and search processes to a more recent collection of published articles from the *Irish Times* newspaper. We also downloaded a set of web pages from the DCU intranet and applied the same processing to this data. Comparative statistics and some general observations on the clustering of three collections of documents are given below:

34,768 *Irish Times* news articles from 1993 (approximately 100Mb of data).
48,050 *Irish Times* news articles from 1996-7 (approximately 130Mb of data).
3050 Web pages from Dublin City University (approximately 20Mb of data).

The web pages vary in size from one or two words up to tens of thousands. In contrast the news article sizes have a smaller spread and typically consist of one or two paragraphs of text. The content of the web pages varies considerably within and across the sub-intranets, but it contains broad unifying themes such as Dublin City University (naturally), teaching, research and general University information. As expected of a national daily newspaper, the news articles have a wide spectrum of themes. The major difference between the web pages and the news articles is that a single web page often has many distinct themes or strands within it, whereas a news article is usually about one major topic. This topic coherency makes the news articles much more suitable to a clustering process that tends to group together documents sharing a common theme. Also the web pages are already structured with links within the intranet. The type of connections that clustering tries to make have already been authored manually. Typically these are more accurate than a fully automated system like clustering can achieve.

For the purposes of comparison of the performance of clustering across these datasets, useful statistics include the number and size of clusters, and the number of documents unclustered. The graphs below show the number of clusters against the size of cluster for each of the three collections.



The table below shows the number of unclustered documents in each of the data sets.

Data set	Number of Documents	Number unclustered
1993 Irish Times	34768	3750
1996-7 Irish Times	48050	5010
DCU web	3050	232

These figures show that about 10% of the documents in the collections remain unclustered, the vast majority of clusters are of 2 to 5 documents and the largest clusters are 15 to 20 documents in size, though these are rare. Such a distribution of cluster sizes is appropriate for our needs, i.e. presenting clusters as part of retrieval results.

In some of the previously published work on document clustering, the clusters have been used as an inherent part of retrieval and the effectiveness of the clustering operation has been evaluated in terms of its impact on precision and recall. Our proposed use of document clusters is in the presentation of results, where a document ranking is obtained using a conventional IR approach but the ranking presented to users consists of a ranked list of document *clusters*. This is discussed more in the following section.

4. Retrieval Using Document Clusters

When a great deal of effort has been expended in creating a judicious clustering of documents in a collection, this should be used effectively in subsequent browsing and retrieval by users. In the searching application we have developed for the clustered data we aim to satisfy a user's information need by a combination of searching and browsing as this is the paradigm with which many users read online newspapers. Conventionally, clustering has been used in information retrieval by automatically incorporating documents related through cluster membership, into a single ranking of documents. The retrieval tool we have developed for searching our clusters is called NewsLink and it is an advanced search facility for news archives. It combines conventional information retrieval full-text searching with the techniques described in this paper that automatically link related news stories. By grouping together stories that share a common theme, NewsLink provides an effective means of finding and tracking news items, where a topic can re-emerge many months after it first appeared, without the cost of having to manually create such an information link.

NewsLink is designed to overcome some of the traditional failings of full-text retrieval systems that typically retrieve only those documents containing one or more of the search terms entered by the user. The lack of overlap between the words in the query (usually only a few words) and the vocabulary in the article results in many potentially relevant articles not being retrieved.

As an example, a search against 34,768 Irish Times articles from 1993 with the query "G7 yen Japan" gives a hitlist with the relevant article "Hosokawa move to stimulate economy seen as inadequate" in the top 10. The cluster containing this article allows links to be made to the following articles:

0.157 "Japan aims to spend its way out of recession"

0.090 "G7 may discuss yen"

0.089 "EC joins lobby to cut Japan's trade surplus"

The linked items all relate to the information request but were not returned in the top 10 retrieved articles and indeed are way down the document ranking. Instead, these articles are ranked by their similarity, in a range 0 - 1, with respect to the article in the cluster that has been retrieved, namely "Hosokawa move to stimulate economy seen as inadequate". While this is really only little more than anecdotal evidence to support clustering for finding information in a hypermedia environment, it illustrates the kind of search/browse interaction we are supporting and is sufficient for us to pursue a more thorough evaluation of the search alternatives technique given that the system architecture and our implementation can deliver the clustering on real and live data.

The difference between the clustering process in NewsLink and “find me more documents like this one” which is becoming commonplace on web search engines is as follows. The “find me more” process simply looks at the hitlist of the document of interest and gives the top N as being similar to it. In practice this results in a large number of false drops – that is documents that really are not closely related to the one of interest. Clustering performs a much more sophisticated analysis of the hitlists of all the documents in the collection a priori, and presents as related only those documents that are clustered with the relevant document. This results in fewer false drops and very coherent “see also” lists.

5. Related Work

As mentioned earlier, clustering of documents as part of a search process has recently started to generate interest in the IR community as its implementation on large collections becomes feasible. For example, Silverstein and Pedersen [8] present an algorithm for almost-constant-time clustering using the Scatter/Gather clustering approach where the Scatter/Gather browsing paradigm clusters and re-clusters document sub-sets on the fly during retrieval. This dynamic re-clustering process as applied to retrieved sub-sets of documents, has been motivated as an alternative to a global hierarchy in response to the time taken for such a global clustering process on a large set of documents. Our work reported here goes some way towards neutralising this disadvantage, especially for dynamic document collections.

One of the relevant recent papers which deals with a global cluster hierarchy [9] looks at the efficiency aspects and some ways to speed up the clustering process by looking at the computation involved in the distance calculations. As we know, the real bottleneck in clustering is computing the $N \times N$ similarity matrix and [9] looks at some pruning techniques for this based on term truncation and how this affects the resulting cluster qualities traded off against the speedup in cluster creation. The term truncation approach is quite simple and is based on representing each document to be clustered by the top 50 or 20 terms occurring within the document. Evaluation of the clustering is accomplished based on evaluating the performance of a subsequent retrieval operation on a corpus of 75k newspaper documents. Using a very powerful SUN machine, an order of magnitude improvement in execution time for the cluster creation was achieved. Retrieval effectiveness is measured by comparing cluster based retrieval based on full clustering against cluster-based retrieval using the reduced clustering techniques proposed but us still far short of similarity-based retrieval. Our contention is that this is the case because the clustering technique used in this work imposed a limit of 400 clusters with an average size of 200 documents whereas as we have seen, we generate far more, and smaller, clusters which is more appropriate for retrieval from newspaper texts.

In a recent special issue of *Information Processing & Management* on the topic of Electronic News, almost all the papers in the issue had content-based association between news stories as a fundamental feature of news retrieval and news access. Work by Saarel *et al.* [10] presented a logical model for an electronic newspaper

implemented on top of an OODBMS which has support for versioning and clustering for search/retrieval, but no details of the clustering process or how daily increments to the database would be handled, were given. A paper by Carrick and Watters [11] looks at the automatic association among themselves of news items but in this case the associated items are of differing media, specifically a text story and an image. This opens the door to a different kind of clustering or similarity measure between stories, the fact that they would use the same image as an illustration suggests a certain similarity between stories. By highlighting clustering as a mechanism to discover relationships between stories, all this reported work is further evidence in support of our belief about the importance of clustering based on similarity, in newspaper access.

6. Conclusions and Plans for Future Work

In this paper, we have described an architecture and an implementation of a system to dynamically cluster an archive of online newspaper articles which facilitates efficient update of the clusters as new stories are published and added to the archive. While the effectiveness and utility of our implementation in terms of an improved service to users has yet to be evaluated formally, the contribution of this paper is in the architectural design and implementation of the system. In a sense, we have made the not unreasonable assumption that as clustering is known to be a good thing for helping users search, it will be a good thing for our implementation.

To evaluate the effectiveness of our document clustering in retrieval we plan to use it as one of our systems in the TREC-7 High Precision track during 1988. By using two versions of our IR system, one with a simple ranked list of documents provided as the output of a user's search and the other with a ranked list of clusters each containing the top-ranked documents, this will examine whether users find benefit from having clusters of related documents presented to them. Results from this will be reported later.

Acknowledgements:

The authors would like to acknowledge the financial support provided by FORBAIRT grant ST/??? and by Dublin City University, and the provision of data by the *Irish Times* newspaper.

References

1 El-Hamdouchi A. and Willett P. Comparison of Hierarchic Agglomerative Clustering Methods for Document Retrieval. The Computer Journal, 32(3):220-227, 1989.

2 Willett P. Recent Trends in Hierarchic Document Clustering: A Critical Review. Information Processing and Management, 24(5):577-597, 1988.

3 Voorhees E.M. The efficiency of Inverted Index and Cluster Searches. In: Proceedings of the ACM Conference on R&D in IR, Pisa, Sept 1986, pp164-174.

4 The Irish Times is at http://www.irish-times.com/

5 HotWired is at http://www.hotwired.com/

6 The Electronic Telegraph is at http://www.telegraph.co.uk/

7 Kelledy F. and Smeaton A.F. TREC-5 Experiments at Dublin City University: Query Space Reduction, Spanish Stemming and Character Shape Coding. In: Proceedings of TREC-5, NIST Special Publication, 1997 (in press).

8 Silverstein C. and Pedersen J.O. Almost-Constant-Time Clustering of Arbitrary Corpus Subsets. In: Proceedings of ACM SIGIR Conference, Philadelphia, July 1997, pp.60-66.

9 Schütze H. and Silverstein C. Projections for Efficient Document Clustering. In: Proceedings of the ACM SIGIR Conference, Philadelphia, July 1997, pp.74-81.

10 Saarela J., Turpeinen M., Puskala T., Korkea-Aho M. and Sulonen R. Logical Structure of a Hypermedia Newspaper. Information Processing & Management, 33(5), 599-614, 1997.

11 Carrick C. and. Waters C. Automatic Association of News Items. Information Processing & Management, 33(5), 615-632, 1997.