# Exploring News Archives using Hierarchical Topic Detection

Master of Science thesis

R.B. Trieschnigg

Enschede, August 2005

| | |
|---|---|
| *Graduation committee* | prof.dr. T.W.C. Huibers<br>prof.dr. F.M.G. de Jong<br>dr. P.E. van der Vet<br>dr.ir. W. Kraaij (TNO) |
| *Chair* | Human Media Interaction |
| *Department* | EEMCS<br>University of Twente, Enschede |

## Abstract

The amount of available information in digital news archives is growing and scalable methods are sought for presenting this information in a user friendly way. Grouping related news items in fuzzy hierarchies has this potential, but the fully automated construction of these structures is complex. Furthermore there is the difficulty of evaluating automatically generated hierarchies of topics.

In this thesis is investigated how hierarchical topic detection (HTD) can aid in the exploration and navigation of large news archives. The contribution of this work is a twofold. First of all a simple scalable HTD system is presented for clustering a large collection of documents in a fuzzy hierarchical topic structure. The prototype system has been used in the trial HTD evaluation of the TDT 2004 evaluation program. The participation starts a discussion of the evaluation methodology of hierarchical topic structures in an experimental context. The second contribution is a set of indicators and a visualization method for evaluating a hierarchical topic structure given a set of flat "truth" topics. With these indicators a richer discussion can take place about the desired properties of cluster structures.

# Acknowledgements

This master thesis marks the end of my Computer Science study period at the University of Twente. I have experienced Computer Science as both a theoretical and practical engineering field, so carrying out this project at TNO, where theory meets practice, was an exciting challenge to conclude my study. At TNO I found out that especially in Information Retrieval a strong emphasis is on practical experience, but as the field matures more theoretical methodologies are sought. Also in this project a large effort was on practice but I have tried to make a contribution to a more theoretical approach of Hierarchical Topic Detection as well.

This work could not have been carried out without the help of others, whom I would like to mention.

First of all I would like to thank TNO, for giving me the opportunity to carry out this project at the institute. I want to thank my colleagues at TNO for their keen interest in my work, in particular Hap Kolb and Willem van Hage. I appreciated the assistance of Harry Wedemeijer during my programming efforts.

I would like to thank my supervisors Theo Huibers, Franciska de Jong and Paul van der Vet for their comments and critical questions in their quest for academic relevance of my work. But most of all I want to thank Wessel Kraaij, my supervisor at TNO who spent a lot of his time on this project. His experience in writing articles and as a paper reviewer were very helpful in writing my first publication for DIR 2005 and preparing a poster for SIGIR 2005.

I want to thank my parents for their loving "sponsorship" and their regular check-up on my progress. Finally I want to mention Elske to thank her for her support during my Delft' endeavour and for her enthusiasm to show me there's more than only work.

Enschede, August 2005.

# Contents

# Chapter 1

# Introduction

## 1.1 Background

Information is one of the most important needs of human kind. With the advent of the computer, databases and the Internet, more and more information is available with a single mouse click. Storages can easily contain hundreds of thousands documents, leaving the user with the problem of finding interesting information.

Most Information Retrieval systems require a clearly defined information need, expressed in some kind of query. This requires knowledge of the information being accessed: which information is actually available and what vocabulary is used to express this information?

*Document clustering* is based on the hypothesis that similar documents will match the same information needs [vR79]. An even richer technique to organize information is a hierarchical structuring, in which clusters can be hierarchically collected in supersets [AFB03]. By grouping related documents and presenting this categorization to the user, an overview can be provided of the concealed information. This overview can help the user to formulate a query expressing his or her information need adapted to the collection.

## 1.2 Topic Detection and Tracking program

The onset of this work is participation in the Topic Detection and Tracking evaluation program. Main purpose of the TDT project is to "advance the state of the art in identifying and following events being discussed in multiple news sources" [NIS04].

Although the TDT program provides a definition (see page 21), it raises the question what actually is a *topic*. A topic is an abstraction of what is discussed in one or more news items and its scope may be difficult to identify. Imagine three news items about the launch, repair in space and return of a particular space shuttle. Do the three news items belong to one and the same topic? Or should these three events be treated as separated topics? Or should all items be subsumed in a topic discussing the space program in which this flight is scheduled? Topic detection is already a difficult task without automating it. It makes the TDT evaluation program an even more challenging task.

TDT is organized by The National Institute of Standards and Technology (NIST) and participants include companies (e.g. Dragon, IBM) and research institutes (e.g. Texas A&M University, University of Iowa, University of Maryland). TNO has participated in the Topic Tracking evaluation of TDT 2000 and 2001 [SK01].

The evaluation incorporates different tasks such as Story Segmentation, Topic Tracking, Topic Detection, First Story Detection and Link Detection. Hierarchical Topic Detection (HTD) is included as a trial evaluation in 2004.

Participants can take part in the evaluations after taking a *dry run*. This initial test has to prove if the participation is worthwhile. The corpus consists of a large set of news sources, composed by the Linguistic Data Consortium (*LDC*). The LDC also prepares the *ground truth*, manually annotated meta data of the corpora. After distribution of the corpora the participants have around one month to send in their results. The results are judged by comparing the system structures to the ground truth. The yearly program is concluded by a workshop where participants can present papers about their systems [NIS04].

## 1.3 Problem identification

In this thesis hierarchical clustering methods are investigated to aid exploration of an unlabelled document collection. Major difficulties are the formulation of cluster structure requirements and corresponding evaluation metric, the actual construction of the cluster structure and the final presentation to the user.

First of all the formulation of cluster structure requirements and corresponding metric is difficult. What defines a good cluster structure for exploring and navigating a large news archive? These requirements may heavily depend on the purpose of the structure, which might be unclear. Furthermore this presents the problem of capturing these requirements in some metric or model in which structures can be evaluated and compared preferably across document collections and producing clustering methods. In the case of TDT the hierarchical structure is compared to the ground truth, in fact a flat cluster structure. Such an evaluation is less trivial than it seems, which will be further explained in chapter 4.

Second the construction of a cluster structure for a document collection of realistic size is problematic. Common cluster approaches with high time and space complexities are not feasible for collections containing a few hundred thousand documents. For example, a cluster method comparing each document pair (quadratic complexity) in a collection of $500,000$ documents, would require almost two and a half trillion $(249,999,500,000)$ comparisons. If one comparison would take one thousandth of a second, this algorithm would run for almost 8 years! Therefore a scalable but still effective solution should be searched.

Third there is the difficulty to overcome the information overload. Assuming documents can be naturally grouped in topic clusters, a news archive will typically consist of many unrelated topics. One way to overcome this is to introduce higher levels of abstraction in the hierarchy, done from a certain viewpoint, but this might lead to an unnatural or undesired structure for the user. The question remains how to present such a large number of groups.

## 1.4 Goal, approach & research questions

The goal of this thesis is to investigate hierarchical topic detection in the domain of large digital news archives to improve exploration and navigation.

The main research question is formulated as follows:

> How can an automated Hierarchical Topic Detection system be configured effectively to improve exploration and navigation of news archives?

Bearing this question in mind was participated in the trial HTD task of TDT 2004. The results from this participation influenced the subsequent research.

The good evaluation results seemed in contrast with the intuitive quality of the cluster structure. Instead of further optimizing the used cluster method was decided to study the evaluation metric in more depth.

The following subquestions will be answered in this thesis:

- How can a very large collection of news items be clustered effectively?
- How can the quality of an hierarchical cluster structure be measured?

## 1.5 Thesis overview

The structure of this report is as follows. In the next chapter a general introduction is given to understand the main concepts throughout this report. Chapter 3 presents a prototype system for clustering a large document collection. This system is used for participation in the HTD task of TDT 2004. Chapter 4 discusses the metric used for evaluation of the HTD task and proposes improvements for evaluating hierarchical topic structures. In Chapter 5 these insights are used to reexamine the experiments carried out for the prototype system. The main research questions and conclusions of this thesis are discussed in Chapter 6.

In Appendix E a small demonstration is given how the constructed cluster structures can be utilized. This is merely an onset for future work and hopes to serve as an inspiration for further developments.

Chapter 2

# Background

This chapter will serve as a background to understand the outline of this report. It will start off with a short introduction to Information Retrieval. In section 2.2 various document clustering techniques are described.

## 2.1 Information Retrieval

Retrieving information is as old as the hills. On one hand a lot of information is produced, distributed and collected, on the other hand information is sought after and consumed. The introduction of the computer has led to questions how to automate the retrieval of information. This field of computer science is called *Information Retrieval*.

Salton (1968) defines Information Retrieval in a very general way [Sal68]:

**Definition.** Information retrieval is a field concerned with the structure, analysis, organization, storage, searching, and retrieval of information.

Van Rijsbergen (1979) describes in his book a typical IR system consisting of three components: input (document collection and queries), output (search results) and a processor [vR79]. The system preprocesses a collection of documents from an information source and stores a representation which allows searching. The system assumes the user is capable of expressing his or her information need in the form of a query, for example keywords describing the desired information. The system presents a list of retrieved documents to the user, best matching the query. In an experimental system the results can be used to evaluate the performance of the retrieval system. Figure 2.1 displays a typical IR system.

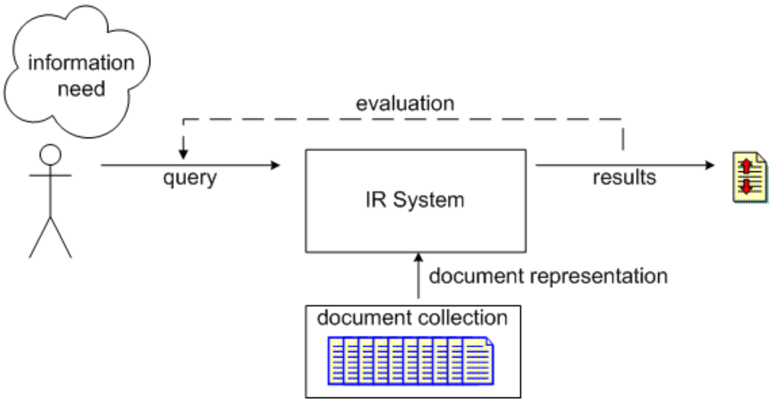


Figure 2.1: A typical IR system; adapated from van Rijsbergen [vR79].

### 2.1.1 Measuring performance

Performance of an IR system can be measured from different viewpoints, e.g. the response time or the user friendliness of the system. It's good to have formal measures, although the original goal of building IR systems should be kept in mind: building a good IR system is not the same as optimizing for a limited metric.

Van Rijsbergen discusses two performance measures [vR79]: *efficiency* and *effectiveness*.

**Definition.** *Effectiveness* is the ability of the system to retrieve relevant documents while at the same time holding back non-relevant one [vR79].

**Definition.** An *efficient* operation is the effective operation as measured by a comparison of production with cost. [MW05].

The cost can be expressed in for example energy, time and money. Van Rijsbergen notes efficiency is hard to measure in a machine independent way.

Two ratios are often used to measure the effectiveness of an IR system: recall and precision. They assume a list of *relevant documents* is known for a particular query, i.e. the desired list of documents for a particular query. By comparing the relevant documents to the retrieved documents, the effectiveness can be measured. Figure 2.2 shows the relationship between relevant and retrieved documents schematically. Table 2.1.1 shows the four categories a document can belong to, given a particular query and retrieved set of documents: it can either be relevant or non relevant to a particular query and it can be retrieved by the IR system or not.

**Definition.** *Recall* is the ratio of the number of relevant documents retrieved to the total number of relevant documents [vR79].

| | | *Is relevant?* | |
|---|---|---|---|
| | | yes | no |
| Is retrieved? | yes | correct | false alarm (false positive) |
| | no | miss (false negative) | correct |

Table 2.1: Contingency table

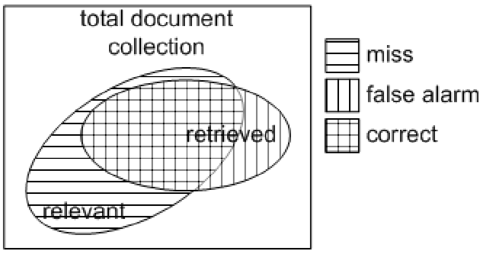


Figure 2.2: Relevant and retrieved documents

Recall gives an impression how complete the list of retrieved documents is.

**Definition.** *Precision* is the ratio of the number of relevant documents retrieved to the total number of documents retrieved [vR79].

Precision gives an impression how pure the list of retrieved documents is.

## 2.2 Document clustering

*Clustering* is the grouping of objects, or *patterns*, in such a way that objects within groups are closely related and the relationship between objects in different groups is weak. Willett (1988) points out the distinction between clustering and classification. During classification objects are assigned to clusters, in contrast to clustering where the number of clusters also needs to be identified [Wil88]. The power of clustering lies in uncovering interesting relationships within a set of unlabelled objects. Cluster analysis is applied in various fields of science including biology, chemistry and data mining.

Van Rijsbergen's *cluster hypothesis* underpins clustering in the context of Information Retrieval [vR79]:

> Closely associated documents tend to be relevant to the same requests.

This hypothesis has been an inspiration for a lot of research to improve the efficiency of IR systems using *document clustering*; by comparing a query to representations of clusters instead of documents, less comparisons have to be made to find groups of related documents. This form of retrieval is called *clusterbased retrieval*. Furthermore has been argued clustering can improve the effectiveness of IR systems [Wil88]. If a cluster is relevant for a particular query, not all the documents in this cluster have to be explicitly related to the query, although the user might be interested in these documents as well.

If a cluster method is actually effective depends on the collection being processed. Van Rijsbergen [vR79], Voorhees [Voo85] and El-Hamdouchi [EHW87] presented methods to find out if a particular document collection has a clustering tendency: these measures give an indication if the document collection has features which allow clustering.

Document clustering can either be done *offline* or *online*. During offline clustering, the document collection is clustered as a whole. Online clustering is done by grouping the results realtime; after retrieving related documents to a query this selection of the document collection is clustered. Especially online clustering has received increasing attention recently, as new personal computers with increased computing speed better allow on the fly clustering.

Jain et al describe the following important aspects of a clustering task [JMF99]:

- Defining a format and extraction method to derive object representations or patterns;
- Defining a measurement to calculate the distance between object representations;
- Defining a method to group object representations in a cluster structure;
- Optionally defining a method to abstract information from the clusters, e.g. giving the clusters a label;
- Defining a method to evaluate the output structure.

Jain points out that these steps have to be adapted to the working domain. In the domain of document clustering characteristic about the document representation is its high dimensionality [JMF99, Sal91], i.e. usually many different features (e.g. terms) are needed to describe a document within a document collection. This high dimensionality of the patterns strongly influences the options for the other aspects.

In the following sections these aspects will be further explained in the context of document clustering.

### 2.2.1 Document representation

Before clustering can take place a suitable representation or *pattern* has to be chosen for the documents.

The most common approach is to represent a document as a *bag of words*. This representation is cheap to store and allows fast and simple (comparison) operations between documents while at the same time it is effective in representing the original document [Sal91]. To obtain this representation, punctuation is removed from the document text and character sequences which do not contain white space are treated as terms. A document is represented by a list of terms and occurrence frequency, called a *term-frequency vector*. Another important value is the number of documents a term occurs in, called the *document frequency* of a term. The term frequency represents the local importance, i.e. the importance in the document. The document frequency can be used to calculate the inverse document frequency ($\log(n/df)$, where $n$ denotes the document collection size and $df$ the document frequency of a particular term), which indicates the global importance of a term.

Different operations can be applied to the term frequency vector, to decrease required storage and to join identical terms (which might be called a cluster operation on its own). To mention a few:

- Converting all the terms to lowercase. Capitalized words at the beginning of a sentence are treated identical as the same words in the middle of a sentence.

- Stemming of terms. Terms are treated by their root form, e.g. "traveling" and "travelled" have the same root "travel". A well-known algorithm is Porter's stemming algorithm [FBY92] which uses plural removal and finds morphological variations. Erroneously the language dependent operation might stem "university" and "universe" to the same root as well ("univers" in the case of Porter's algorithm).

- Removing stop words. Stop words are words which occur often and don't have information value, e.g. "a", "the", "have" etc. Furthermore words can be removed which occur very infrequently as they will not be used in queries or do not co-occur in multiple documents (which makes these terms less useful for clustering).

It should be stressed that choosing a particular representation has a strong impact on the performance of an IR system. During this step a model is created of a document. This abstraction enables calculations and comparisons, but a lot of the original context is thrown away.

Figure 2.3[1] shows how an excerpt of text is processed to its final representation after removal of stopwords and stemming. Note that the stemmed term "appl" represents two different but related concepts from the original text: the talking *apple tree* and the *apples* which grew from it.

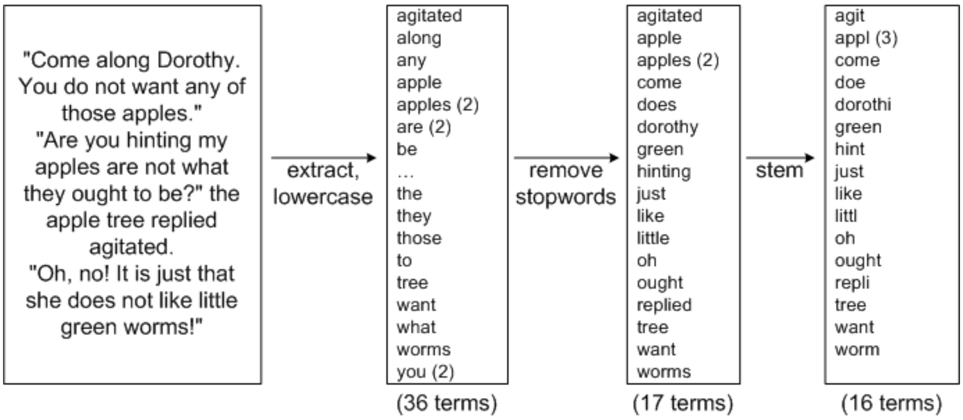

"Come along Dorothy. You do not want any of those apples." "Are you hinting my apples are not what they ought to be?" the apple tree replied agitated. "Oh, no! It is just that she does not like little green worms!"
extract, lowercase
agitated along any apple apples (2) are (2) be … the they those to tree want what worms you (2)
(36 terms)
remove stopwords
agitated apple apples (2) come does dorothy green hinting just like little oh ought replied tree want worms
(17 terms)
stem
agit appl (3) come doe dorothi green hint just like littl oh ought repli tree want worm
(16 terms)

Figure 2.3: From original document to document representation

### 2.2.2 Calculating document distances

It seems intuitively clear that for clustering a measurement is needed to compare documents. This distance metric should give an indication if a document is more similar to one document than to another. Calculating this similarity is strongly related to the document representation discussed before.

Using the previously described representation, a trivial measurement could be the number of co-occurring terms in two documents. The more identical terms occur in two documents, the higher the similarity between those documents. Such a measurement would not work for documents of different sizes however: larger documents containing more terms, will have more identical terms than smaller documents. In some way normalization should take place to prevent this.

Two often used similarity measurements which incorporate normalization are the *dice* and *cosine* similarity measures. The similarity between two documents $d_1$ and $d_2$ is defined as follows:

[1] Excerpt from The Wizard of Oz by L Frank Baum.

$$\mathrm{Dice}(D_1, D_2) = \frac{2 \cdot |D_1 \cap D_2|}{|D_1| + |D_2|} \tag{2.1}$$

$$\mathrm{Cosine}(D_1, D_2) = \frac{|D_1 \cap D_2|}{\sqrt{|D_1| \cdot |D_2|}} \tag{2.2}$$

Where $D_1$ and $D_2$ are the terms of document $d_1$ and $d_2$ respectively.

Note that both similarity measurements are symmetric. Similarity measurements can also be asymmetric: the measurement then indicates the resemblance of a document using the features of another document as starting point.

Term weighting can be applied to emphasize the importance of certain terms; the measures can also incorporate this. For more information see any text book on Information Retrieval, e.g. from Baeza-Yates and Ribeiro-Neto [BYRN99].

### 2.2.3 Clustering methods

Using the distance metric and a set of documents (a set of representations to be more precise) the actual clustering process can take place. The cluster method describes the steps to build the cluster structure.

The resulting cluster structure can be a *flat* or *hierarchical* representation of the document set. In a flat cluster structure there is no relationship between the clusters; each cluster is an independent group of documents. In a hierarchical structure, there can be a relationship between clusters; a cluster can be composed of documents as well as clusters.

A cluster structure can be either *hard* or *fuzzy*. In a hard cluster structure a document can only be assigned to one cluster. In a fuzzy cluster structure a document can be part of multiple unrelated clusters. In a fuzzy structure documents are often assigned to clusters to a certain degree.

In the following sections a number of classical clustering approaches are discussed. In the last section a few hybrid methods are briefly mentioned, which combine several cluster methods or introduce new approaches.

#### K-means clustering

McQueen introduces in 1967 the K-means clustering algorithm [JMF99]. K-means clustering is a flat clustering method, with the goal of partitioning the document collection in $k$ clusters, where there is little similarity across clusters, but great similarity within a cluster.

The method is as follows:

1. $k$ random document patterns are chosen as initial *centroids*, each centroid represents the centre of a cluster.
2. The document patterns are assigned to the cluster of the closest centroid.
3. For each formed cluster the document pattern which is in the centre is assigned as new centroid of that cluster.
4. Step 2 and 3 are repeated until some kind of convergence criterion is met, e.g. no or little patterns are reallocated anymore.

Advantages:

- K-means is simple to implement and has a linear time complexity, $O(n)$ where $n$ is the number of documents.

Disadvantages:

- The number of clusters $k$ has to be determined beforehand.
- The resulting structure is sensitive to the inital centroids, this makes the result not deterministic.

A variant of K-means is K-medioid, which uses artificial document patterns as centres representing clusters (at step 3 of the algorithm). Other resembling methods have tried to optimize finding the initial centroids, or allow splitting of clusters with a high variance and merging of clusters having close centroids [JMF99].

### Hierarchical agglomerative clustering

As the name implies, hierarchical agglomerative clustering (HAC) creates a hierarchical structure. To be more specific, it creates a *dendrogram* by iteratively merging the two most similar clusters until one root cluster remains.

The method is as follows:

1. A distance matrix is created, in which the distances between all document pairs are stored. The documents are viewed as singleton clusters.
2. The most similar clusters are joined in a new cluster.

3. The distances between this newly created cluster and all the remaining clusters are calculated.

4. Step 2 and 3 are repeated until a single, root cluster remains.

The result is a binary tree containing clusters as nodes and singleton clusters (containing one document) as leafs. In the worst case the depth of this tree equals the number of documents minus one; in this case the structure looks like a chain of singleton clusters (see figure 2.4). The smallest depth is achieved having a fully balanced tree with depth $\lceil \log_2 n \rceil$, with $n$ the number of documents.

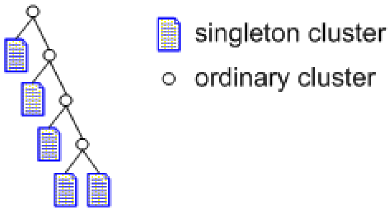


Figure 2.4: Chaining of document clusters

For calculating the distances between newly formed clusters and existing clusters(step 3), three methods are widely used:

**Single link** also known as nearest neighbour. The distance between the new cluster and the existing cluster is the minimum of the distances between the existing cluster and one of the joined clusters. The resulting cluster structure tends to suffer from the previously described chaining effect [Wil88].

**Complete link** also known as furthest neighbour. The distance between the new cluster and the existing cluster is the maximum of the distances between the existing cluster and one of the joined clusters. The resulting cluster structure tends to contain small compact groups of clusters [JMF99].

**Group average link** The distance between the new cluster and the existing cluster is the average of the distances between the existing cluster and the joined clusters.

Figure 2.5 shows these distance measures graphically.

Figure 2.6 shows an example of hierarchical clustering.

Advantages of HAC:

- The resulting cluster structure is deterministic. Processing a collection will always produce the same result.

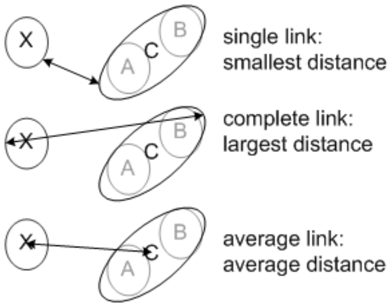


Figure 2.5: Calculation of the distance between a new cluster ($C$ joining cluster $A$ and $B$) and an existing cluster ($X$) using single, complete and average link methods.

Disadvantages of HAC:

- The time and space complexity of HAC methods in general is $O(N^3)$ in time and $O(N^2)$ in space. This makes the method not scalable for large document collections [EHW86].

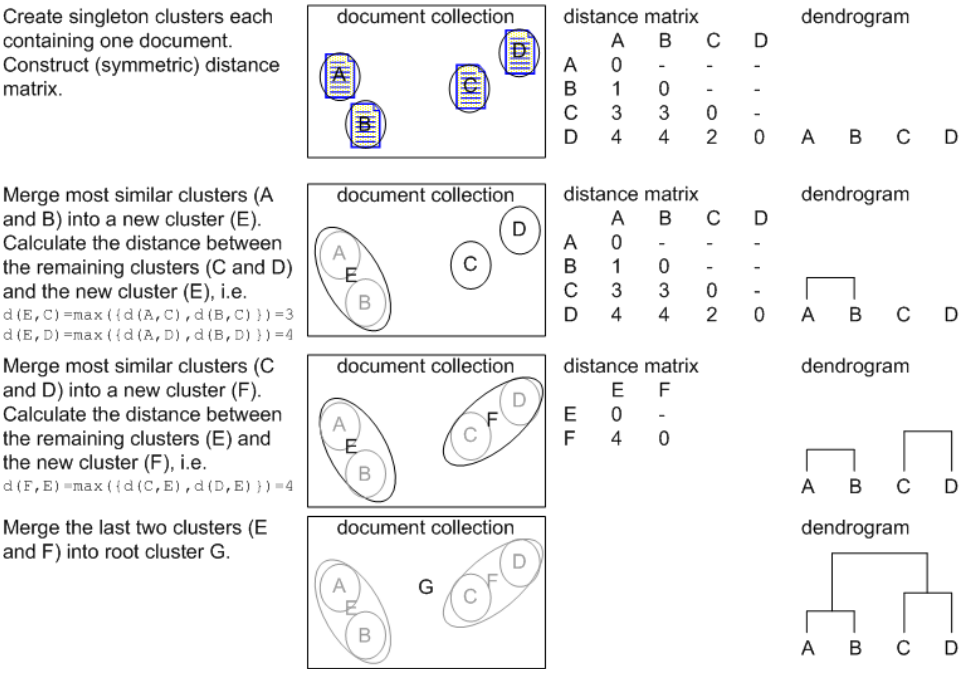
A well-known variation on HAC is Ward's method, also known as minimum variance method [EHW86, FBY92]. At each step of the clustering process, the pair of clusters is merged whose merger minimizes the increase of within-group distances. Ward's method does not outperform other HAC methods in complexity however.

### Bisecting K-means

Although the name is a little confusing the bisecting K-means algorithm actually is a divisive hard hierarchical cluster method. Research has shown it produces better results than the basic K-means algorithm described before [SKK00]. Again the goal is to partition the collection in $k$ clusters.

The method is as follows:

1. Put the complete document collection in a single cluster.
2. Pick a cluster to split, e.g. the largest cluster.
3. Use the basic K-means algorithm to find two subclusters (bisecting step).
4. (Optional to increase performance) Repeat step 3 a number of times and take the best two subclusters.
5. Repeat step 2, 3 and 4 until $k$ leaf clusters are created.



Create singleton clusters each containing one document. Construct (symmetric) distance matrix.

distance matrix

| | A | B | C | D |
|---|---|---|---|---|
| A | 0 | - | - | - |
| B | 1 | 0 | - | - |
| C | 3 | 3 | 0 | - |
| D | 4 | 4 | 2 | 0 |

Merge most similar clusters (A and B) into a new cluster (E). Calculate the distance between the remaining clusters (C and D) and the new cluster (E), i.e.

```
d(E,C)=max({d(A,C),d(B,C)})=3
d(E,D)=max({d(A,D),d(B,D)})=4
```

distance matrix

| | A | B | C | D |
|---|---|---|---|---|
| A | 0 | - | - | - |
| B | 1 | 0 | - | - |
| C | 3 | 3 | 0 | - |
| D | 4 | 4 | 2 | 0 |

Merge most similar clusters (C and D) into a new cluster (F). Calculate the distance between the remaining clusters (E) and the new cluster (F), i.e.

```
d(F,E)=max({d(C,E),d(D,E)})=4
```

distance matrix

| | E | F |
|---|---|---|
| E | 0 | - |
| F | 4 | 0 |

Merge the last two clusters (E and F) into root cluster G.

Figure 2.6: Hierarchical clustering 4 documents using complete linkage.

advantages:

- The method has logarithmic time complexity ($O(\log(n))$).

disadvantages:

- The cluster structure is not deterministic because of its dependence on random initial centers.

### Adaptations

As became clear, an important problem of clustering is the friction between robustness and effectiveness on one hand and efficiency on the other. Hierarchical agglomerative clustering techniques do seem to produce effective clusters [Wil88], but simply are not feasible for large collections. On the other hand, K-means can produce cluster structures with a low computational cost, but tends to give varying results. Various hybrid adaptations have been developed to bundle the advantages and overcome the drawbacks, sometimes presenting additional steps in the process. Three methods are typically used to enable application of complex algorithms on large datasets: sampling, summarizing and partitioning [DL01]. In the following sections a few adaptations will are discussed.

Hearst and Pedersen introduced *Scatter/Gather*, a cluster-based document browsing method which uses a linear cluster algorithm for online document clustering [HP96, CKPT92]. They claim it presents a compact list of topically-coherent groups, labelled with topical terms which characterize these clusters. The user can select (gather) a few clusters of interest and the system presents a new list of clusters scattering the previous selection on-the-fly. Hearst and Pedersen present two algorithms, Buckshot and Fractionation, which help in determining the $k$ initial centres used for K-means clustering. Buckshot uses another (possibly slow) clustering algorithm on a sample of the document collection, to find the centroids. By taking a sample of size $\sqrt{kn}$ (with $k$ the number of clusters and $n$ the number of documents) and with a quadratic clustering algorithm, this approach is linear for the number of documents. Fractionation divides the document collection in a number of buckets of $m$ documents. Another cluster routine is applied to each bucket to find a number ($< m$) of document representations. These document representations are collected and divided into new buckets (again each containing $m$ document representations) and the process is repeated until $k$ document representations remain [CKPT92]. Pirolli et al evaluated Scatter/Gather and concluded it does not improve locating specific documents, but it does help in understanding the topic structure of the document collection [PSHD96].

Zhang, Ramakrishnan and Livny presented *BIRCH* as a clustering method for large databases, not particularly with the goal of document clustering [ZRL96]. This is done by compressing the information of data patterns in *Clustering Features* (CF), which contain the statistics of a selection of closely related data patterns. The clustering features are stored in a *clustering feature tree*, which can be constructed by a single scan of the collection of patterns (documents). The Clustering Features stored as leafs of this tree can be further processed using an arbitrary clustering algorithm. Furthermore the tree can be used to detect and remove outliers, patterns which are not closely related to any other pattern. The resulting clustering feature tree is influenced by the order in which patterns are processed, but by post processing this influence can be decreased is argued by the creators. Wong and Fu adapt this method for document clustering, introducing a document feature similar used to the cluster feature used in BIRCH [WF00]. Both systems require setting a number of parameters, influencing running time and clustering results. The values of these parameters are chosen experimentally.

Pantel and Lin introduce document clustering with committees, which is a variant on K-means clustering [PL02]. The method calculates the centroid of the cluster by averaging a tight subset of the document representations in a cluster. They explicitly argue against monothetic clustering: "Using a single representative from a cluster may be problematic too because each individual element has its own idiosyncrasies that may not be shared by other members of the cluster".

Sanderson derives concept hierarchies from text by using term co-occurrence [SC99]. He prefers using a monothetic clustering method instead of a polythetic clustering one, as used in for example Scatter/Gather. In polythetic clustering the relationship between documents in clusters is based on multiple terms, whereas in monothetic clustering only one term is used. He argues that polythetic clustering does not bring clarity to the user when presenting labels of clusters containing multiple keywords. By discovering term relationships (one subsumes the other), a hierarchy of terms is created.

### 2.2.4 Cluster structure evaluation

Created cluster structures can be evaluated to compare the performance of different cluster methods or to compare the cluster performance of a particular method on two different document collections. The focus of this evaluation again will be on the effectiveness (see 2.1.1) of the resulting structure.

The effectiveness of a cluster method is usually measured using a *ground truth*. A ground truth is a manually composed cluster structure. The document collection may be partially annotated: in this case the ground truth does not completely cover the collection. A common approach is to find for each cluster in the ground truth a cluster in the system structure and to calculate its precision and recall.

The evaluation method depends on the cluster structure of both the system cluster and ground truth structure. A highly fuzzy cluster structure requires a different approach than a hard partitioned set of clusters. Another aspect is the difference in granularity, i.e. level of detail of the structure, between system en truth cluster structure. It's difficult to create a mapping between system and truth clusters if they are of different size and there is no information about relationships between system or truth clusters.

In chapter 4 a metric for evaluating hierarchical cluster structures with a flat ground truth will be discussed.

## 2.3 Conclusion

In this chapter a theoretical background has been provided for this report. It can be concluded that many document clustering methods have been developed through the years, usually balancing between efficiency and effectiveness. The evaluation of the resulting cluster structures is dependent on the type of structure.

Chapter 3

# Prototype

This chapter describes the prototype developed to participate in the TDT 2004. The first section discusses the Hierarchical Topic Detection task and evaluation methodology of the TDT evaluation study. The second section describes the design of the prototype system, followed by a description of the experiments carried out with this prototype system. Section 3.4 discusses the results of carrying out these experiments and is followed by a conclusion.

## 3.1 TDT evaluation study

The Topic Detection and Tracking (TDT) project is an annually held evaluation study in the field of TDT organized by the National Institute of Standards and Technology (NIST).

### 3.1.1 Hierarchical Topic Detection Task

TDT has included a Topic Detection task since its inception in 1996. In this task systems are required to organize news stories in clusters, corresponding to the topics discussed. The result can be regarded as a partition of the corpus, in which each news item is assigned to one and only one partition representing a topic.

The Task Definition and Evaluation Plan of TDT 2004 [NIS04] mentions two reasons for introducing a new *Hierarchical Topic Detection* task:

- A flat partitioned structure does not allow a single news item to belong to multiple topics.

- A flat structure does not allow multiple levels of granularity, i.e. topics cannot be described at various levels of detail.

The new HTD task enables stories to be assigned to multiple clusters. Furthermore clusters may be a subset of, or overlap with other clusters. The resulting structure must be characterizable as a *directed acyclic graph* (DAG) with a single root node. The root node represents the complete document collection whereas child clusters further down the DAG represent finer grained topics. For this initial trial evaluation, the task simplifies treatment of time: the task is treated as retrospective search, i.e. the documents may be processed in any order, in contrast to the old task in which the items should be processed in the order they were published [NIS04].

Figure 3.1 shows a cluster structure of a very small dataset containing five documents. The root node contains all documents (note that no documents are directly attached) and the clusters below the root (named $A$ and $C$) each contain a subset from the document collection. Notice the overlap between the clusters $A$ and $B$ (fuzzy clustering) and the different levels of granularity between for example clusters $B$ and $C$.

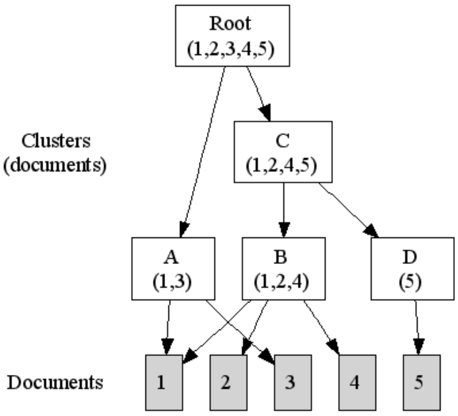


Figure 3.1: Example cluster structure (directed acyclic graph) of five documents.

### 3.1.2 Corpus

The collection of stories is provided by the Linguistic Data Consortium (LDC) and contains collections of news from a number of sources and languages. The TDT 5 corpus used for the TDT 2004 contains English, Mandarin and Arabic data. The non-English sources are also available in machine translated English. The total corpus consists of around 400,000 documents, of which around 280,000 in native English. Participants for the HTD task are required to send in a cluster structure of the English only data and a structure of the complete corpus, constructed by either using the machine translated or original sources.

### 3.1.3 Ground truth

The systems are scored by comparing the system result to a manually composed *ground truth*. The cost of a (cluster) structure defines the "distance" to the ground truth; a better structure has a lower cost. The ground truth is composed by annotators of the Linguistic Data Consortium and consists of manually labelled clusters containing news stories discussing a particular topic. A topic is defined as follows:

**Definition.** A topic is an event or activity, along with all directly related events and activities [Lin04].

The topics are selected from a random sample of documents from the corpus. The annotation is search guided, i.e. the related stories are found using a search engine. Furthermore the annotation for the most recently published TDT 5 corpus is incomplete, that is, there will be no guarantee that every story on each topic will have been located [Lin04]: The search for stories related to one particular topic is ceased after 3 hours, in contrast to previous annotations where the annotators decided when all on-topic stories were found.

### 3.1.4 Evaluation metric

The metric used for the old Topic Detection task [FD02] is not suitable for this new hierarchical task. Allan et al [AFB03] discuss various methods for evaluating hierarchical cluster structures. The TDT 2004 HTD task is evaluated by using the minimal cost method described in Allan et al's paper.

The minimal cost metric finds for each annotated topic the system's optimal cluster, having the lowest cost. This cost consists of two component:

- A *detection cost* related to the contents of the cluster and
- a *travel cost* related to the effort to find the cluster.

The detection cost is the same as for previous topic detection tasks and consists of a penalty for false alarms and misses, misses have more impact than false alarms however (see section 2.1.1). The travel cost has been introduced to penalize "powerset" cluster structures, i.e. structures having clusters containing all possible combinations of document sets. The travel cost of a cluster is, independent of its content, related to the shortest path to this cluster from the structure's root cluster. The number of encountered branches and the length of the path are the major components in the travel cost calculation, representing the number of choices a user has to make and the number of cluster titles a user has to read to find the best matching cluster. The score function is parametrized, i.e. the impact of the various cost components is controlled by parameters.

### 3.1.5 Formal domain definitions

Before we discuss how to calculate the minimal cost metric, a more formal definition of the domain is presented.

Assume a document collection $D$ consists of $n$ documents, $d_1$ to $d_n$. For this document collection we have a ground truth $G$, which is a set of $g$ topics $t_1$ to $t_g$. Each topic is a subset of documents from $D$:

$$\forall t \in G \bullet t \subseteq D$$

The ground truth may be incomplete, i.e. the topics might not cover all documents in D:

$$\bigcup_{t \in G} t \quad \subseteq \quad D$$

A system cluster structure $S$ is a directed acyclic graph, which is represented as a pair of clusters $V$ and edges $E$. $V$ is a set of $v$ clusters, $c_1$ to $c_v$. The edges $E$ are a set of $e$ tuples in the form $(c_p, c_c)$, where $c_p$ and $c_c$ are both clusters in $V$. The tuple represents a parent-child relationship between the clusters $c_p$ and $c_c$ respectively. A cluster cannot be a child of itself (the relationship is not reflexive).

$$\begin{aligned} S &= (V, E) \\ V &= \{c_1, c_2, \ldots, c_v\} \\ E &= \{(c_{p_1}, c_{c_1}), (c_{p_2}, c_{c_2}), \ldots, (c_{p_e}, c_{c_e})\} \end{aligned}$$

The parents and children of a cluster $x$ are defined as follows:

$$\begin{aligned} P(x) &= \{p \mid (p, x) \in E\} \\ C(x) &= \{c \mid (x, c) \in E\} \end{aligned}$$

A *cluster path* of length $p$ ($p > 1$) is defined as a series of $p + 1$ (connected) clusters $(c_1, c_2, \ldots, c_{p+1})$, where for each sequence of two clusters $c_k$ and $c_{k+1}$ holds that the tupel $(c_k, c_{k+1})$ is an edge in $V$. $(c_1 \Rightarrow c_2)$ denotes a cluster path between cluster $c_1$ and cluster $c_2$. The set $P$ denotes all the possible paths in the cluster structure. The cluster structure is acyclic, i.e. there is no cluster path between a cluster and itself.

There is one *root cluster* $c_r$, which is the "top" of the cluster structure. This root cluster is no child of any other cluster and there is a path to all other clusters from this root cluster.

The *offspring* of a cluster $c$ is defined as all the clusters which can be reached from $c$:

$$O(c) = \{c_c \mid (c \Rightarrow c_c) \in P\}$$

A cluster $c$ has a set of *directly contained documents* $D_c$. The *documents of a cluster c, $D^*(c)$*, are defined as the directly contained documents of this cluster and the directly contained documents of all of its offspring clusters:

$$D^*(c) = d_c \cup \bigcup_{i \in O(c)} D_i$$

### 3.1.6 Minimal cost (the math)

The minimum cost $\mathrm{C}_{\mathrm{min}}$ can be calculated for a certain cluster $c$ and topic $t$. A lower value indicates a better performance. It is a linear combination of normalized detection cost ($\mathrm{C}^*_{\mathrm{d}}$) and normalized travel cost ($\mathrm{C}^*_{\mathrm{t}}$):

$$\begin{aligned} \mathrm{C}_{\mathrm{min}}(c,t) &= \textsc{wdet} \cdot \mathrm{C}^*_{\mathrm{d}}(c,t) + \\ &\quad (1 - \textsc{wdet}) \cdot \mathrm{C}^*_{\mathrm{t}}(c_r \Rightarrow c) \end{aligned}$$

WDET ($0 \leq$ WDET $\leq 1$) is a constant which sets the relative impact of detection and travel cost.

In the following sections the detection cost and travel cost are discussed.

#### Detection cost

The detection cost penalizes false alarms and misses and is defined as follows for a particular cluster $c$ and topic $t$:

$$\begin{aligned} \mathrm{C}_{\mathrm{d}}(c,t) &= \mathrm{P}_{\mathrm{miss}}(c,t) \cdot \mathrm{P}_{\mathrm{relevant}} \cdot \mathrm{CMISS} + \\ &\quad \mathrm{P}_{\mathrm{fa}}(c,t) \cdot \mathrm{P}_{\mathrm{nonrelevant}} \cdot \mathrm{CFA} \\ \mathrm{P}_{\mathrm{miss}}(c,t) &= \frac{|t \backslash D^*(c)|}{|t|} \\ \mathrm{P}_{\mathrm{fa}}(c,t) &= \frac{|D^*(c) \backslash t|}{|D \backslash t|} \end{aligned}$$

Where:

- $|X|$ is the size of the set $X$.
- $u \backslash v$ is the set $u$ excluding the elements in $v$.
- $\mathrm{P_{miss}}$ and $\mathrm{P_{fa}}$ are the probabilities of a missed detection and a false alarm respectively when comparing a cluster to a ground truth topic.
- CMISS and CFA are the costs of a missed detection and a false alarm respectively.
- $\mathrm{P_{relevant}}$ is the a priori probability of finding a relevant document ($\mathrm{P_{nonrelevant}} = 1 - \mathrm{P_{relevant}}$). This is to compensate for the typical small value of $\mathrm{P_{miss}}$ and large value of $\mathrm{P_{fa}}$, if the average size of a topic is relatively small compared to the size of the total document collection [FD02]. The values of these constants are predetermined for a particular corpus.

Normalization of the detection cost is done by dividing by the minimum of costs if all documents are judged on-topic ($\mathrm{P_{nonrelevant}} \cdot \mathrm{CFA}$) or off-topic ($\mathrm{P_{relevant}} \cdot \mathrm{CMISS}$). The goal of this normalization is to ground the performance to a more meaningful range [FD02]:

$$\mathrm{C_d^*}(c,t) = \frac{\mathrm{C_d}(c,t)}{\min(\mathrm{P_{relevant}} \cdot \mathrm{CMISS}, \mathrm{P_{nonrelevant}} \cdot \mathrm{CFA})}$$

**Travel cost**

The encountered branches on a path, $eb(c_{from} \Rightarrow c_{to})$, are defined as follows:

$$\begin{aligned} eb(c_{from} \Rightarrow c_{to}) &= eb((c_1, c_2, \ldots, c_n)) \\ &= \sum_{i=1..n-1} |C(c_i)| \end{aligned}$$

Where $|C(c)|$ are the number of children of cluster $c$.

The travel cost $\mathrm{C_t}$ consists of a cost for the number of encountered branches and encountered titles when traversing to a cluster $c$ from the root cluster $c_r$.

$$\mathrm{C_t}(c_r \Rightarrow c) = eb(c_r \Rightarrow c) \cdot \mathrm{CBRANCH} + |c_r \Rightarrow c| \cdot \mathrm{CTITLE}$$

Where:

- $c_r \Rightarrow c$ is a path from the root cluster to cluster $c$. This path can be seen as a series of clusters $(c_1, c_2, \ldots, c_n)$ with $c_1 = c_r$ and $c_n = c$.
- $|c_p \Rightarrow c_c|$ is the length of the path from $c_p$ to $c_c$.
- CBRANCH is the cost for "encountering" a branch.
- CTITLE is the cost for "reading" a title.

Normalizing is done by dividing by the expected travelcost in an optimally balanced tree, using an optimal (preferred) branching factor:

$$\begin{aligned}
\mathrm{C}_{\mathrm{t}}^{*}(c_r \Rightarrow c) &= \frac{\mathrm{C}_{\mathrm{t}}(c_r \Rightarrow c)}{\begin{pmatrix}\text{EXPDEPTH} \cdot \text{OPTBRANCH} \cdot \text{CBRANCH} + \\ \text{EXPDEPTH} \cdot \text{CTITLE}\end{pmatrix}} \\
\text{EXPDEPTH} &= \lceil \log_{\text{OPTBRANCH}} n \rceil
\end{aligned}$$

Where

- OPTBRANCH is the optimal branching factor, i.e. the optimal number of child clusters a cluster has. For the HTD task of TDT 2004 this value is 3.
- EXPDEPTH is the expected depth of the balanced tree with as many leaves as the size of the document collection with branching factor of OPTBRANCH.

The evaluation algorithm carries out a top-down tree search to find the best cluster for each ground truth topic. As only the best cluster is required, a trimming algorithm can be used: if the cost of a particular branch cannot generate a better result than the current optimal value, that branch can be trimmed and excluded from further search [AFB03].

The metric and its algorithm are more thoroughly described in Allan et al's paper [AFB03], the TDT evaluation plan [NIS04] and a paper from Fiscus [FD02].

## 3.2 Design

An attempt is made to adapt a common hierarchical agglomerative clustering method for the large TDT dataset consisting of almost 400,000 documents.

The approach is as follows: take a sample small enough so it can still be clustered using hierarchical agglomerative clustering, cluster it and assign the remaining documents to the best matching clusters. Figure 3.2 shows this approach schematically.

The steps are explained in the following sections.

### 3.2.1 Sampling

The first step is to take a random sample from the corpus. The size of this sample is 20,000 documents, its corresponding distance matrix requires an acceptable 800 megabytes of working memory[1].

### 3.2.2 Clustering

The second step is to build a hierarchical cluster structure. Starting point for the clustering method is the cross-entropy reduction scoring function [Kra04]. Suppose we have two documents $D_1$ and $D_2$. Both documents are represented by simple unigram language models $M_{D_1}$ and $M_{D_2}$, a reference unigram model for general English $M_C$ is estimated on the complete document collection. Now the cross-entropy reduction (CER) of $M_{D_1}$ and $M_{D_2}$ compared to $M_C$ is defined as:

$$\begin{aligned} CER(D_1; C, D_2) &= H(D_1, C) - H(D_1, D_2) \qquad (3.1) \\ &= \sum_{i=1}^{n} P(\tau_i|M_{D_1}) \log \frac{P(\tau_i|M_{D_2})}{P(\tau_i|M_C)} \\ &= \sum_{i=1}^{n} \frac{c(D_1, \tau_i)}{\sum_i c(D_1, \tau_i)} \log \frac{(1-\lambda)P(\tau_i|D_2) + \lambda P(\tau_i|C)}{P(\tau_i|C)} \end{aligned}$$

Where:

- $\tau_i$ is an index term.
- $n$ is the number of unique index terms in $C$.
- $c(X, \tau)$ is the term frequency of term $\tau$ in language model $X$.
- $\lambda$ is a smoothing parameter.
- $H(X, Y)$ is the cross-entropy between two language models $X$ and $Y$.

[1] 4 bytes (typical implementation of a floating point value) per comparison in a symmetric matrix

The generative document model $M_{D_2}$ is smoothed by linear interpolation with the background model $M_C$ [SK02]. Normalization of scores (by subtracting $H(D_1, C)$) is essential for adequate performance.

For more information about the cross-entropy scoring function is referred to Kraaij's PhD thesis [Kra04].

The symmetrical version of this scoring function is defined as

$$sim(D_1, D_2) = \frac{CER(D_1; C, D_2) + CER(D_2; C, D_1)}{2} \tag{3.2}$$

A distance matrix is filled using this symmetrical scoring function. For the actual clustering 3 basic hierarchic agglomerative clustering methods are used: single, complete and average pairwise linkage (see section 2.2.3).

### 3.2.3 Optimizing

The result of this clustering process is a, usually unbalanced, binary tree. A skewed cluster, i.e. a cluster which has child clusters containing an unequal number of documents, adds extra travel cost to all of the clusters below this cluster, especially if this cluster is near the root of the tree. Relating to the real world, the 'user' should consider more branches and titles to find clusters. A more balanced tree will reduce the expected travel cost, but how can the structure be rebalanced without losing clustering information? The metric is used to indicate if the changes to the tree have thrown away clustering information: if after rebalancing the detection cost for any 'optimal' cluster grows, the rebalancing has thrown away valuable information from the original structure. The detection cost should remain the same and the travel cost should decrease.

The method used for rebalancing the tree, without large changes to the optimal clusters, is simple. First the clusters are removed which have no documents directly[2] attached and have a dissimilarity higher or equal to a certain threshold. A group of unconnected clusters now remains. These clusters are used to form a better balanced tree with a branching factor of three, suiting the HTD evaluation metric preferring tertiary or quadruple trees[NIS04]. This is done by recursively taking the smallest three (or four) clusters to form a new cluster, until only one root cluster remains.

Figures 3.3(a) and 3.3(b) show the impact of the rebranching operation on a cluster structure of 100 documents. The black squares in the bottom of the visualization represent documents, the rectangles represent clusters grouping documents and clusters. The marked clusters in the first figure will be removed: their dissimilarity is higher than or equal to the chosen threshold

[2] a directly attached document only appears in this cluster and not in child clusters

and they don't directly contain documents. After removing these clusters and corresponding edges, a group of small cluster branches remains. The second figure shows the result of building a more balanced tree with these small branches. The marked clusters now represent newly added clusters. Note the newly added clusters have a branching factor of three and the existing clusters have a branching factor of two.

### 3.2.4 Merging

An index is built from the sample document set. The documents from the corpus which are not in the sample are used as queries on this index returning the best document-likelihood matches. For each document in this complement dataset the best 10 matches are used for merging. The complement document is added to all of the matching documents' clusters.

The new documents which don't have any matching documents are collected in one cluster. This ensures new documents are assigned to at least one cluster.

The result of the merging process is a so called fuzzy cluster structure: a single document can belong to multiple clusters.

## 3.3 Experiments

Experiments were carried out using the English sources from the TDT 3 corpus as a preparation for participation in the trial HTD task of TDT 2004. The size of this dataset is around 35,000 documents, roughly one tenth of the TDT 5 dataset. As a sample we took 10,000 documents from the TDT 3 corpus.

For this set of sample documents a symmetric distance matrix was created, filled with the dissimilarity between each document pair. Using this matrix a cluster structure was built using single, complete and average link methods. The sample structure was scored using the minimal cost metric and TDT 3 ground truth containing 160 topics. Based on the bad results for single linkage was decided to exclude this method from further experiments.

Experiments were carried out with rebranching, varying the cut threshold (0, 0.90, 0.95, 0.96, 0.97 and 0.98) and varying the number of branches (3 and 4) to use when reconstructing the tree. Furthermore experiments were carried out applying rebranching before and after the merging process. Other tree simplifying operations were studied, also changing the structure in the lower parts of tree, but these resulted in similar or worse evaluation results and are not further discussed.

Table 3.1: Comparison of clustering methods

| *Method* | *Minimum cost* | *Detection cost* | *Norm. det. cost* | *Travel cost* | *Norm. travel cost* | *Depth* |
|---|---|---|---|---|---|---|
| Average link | 0.2747 | 0.0074 | 0.3722 | 58.4 | 0.0855 | 11.68 |
| Complete link | 0.6120 | 0.0176 | 0.8778 | 65.7 | 0.0962 | 13.14 |
| Single link | 0.6970 | 0.02 | 1.0003 | 74.0 | 0.1084 | 24 |

The documents in the complement dataset were used as queries for the built sample index. The 20 best matching documents from the sample were searched, using document likelihood. The new documents were assigned to the clusters to which the best matching documents from the sample belong. Two methods were used:

- Adding the new document to the first $n$ (1, 10, 20) matching clusters.
- Adding the new document to the first matching cluster and to to all matching clusters having a document likelihood higher than a certain threshold (0.5, 1, 2)

The minimal cost was calculated over all of the generated cluster structures, including structures only containing sample documents.

The configuration with optimal result for the TDT 3 test collection was used for the TDT 2004 participation. This was constructing a sample cluster structure using average pairwise link for 20,000 documents, applying a rebranch with branching factor 3 and cut threshold 0.96 and finally adding the remaining documents to the clusters of the 10 best matching sample documents. Creating the cluster structure of the TDT 5 corpus took around one complete day of processing time on a 900 Mhz machine having 2 Gb of working memory.

## 3.4 Discussion

In this section the most interesting results from the experiments and participation in TDT 2004 are discussed.

### 3.4.1 Linkage method

First of all the choice of linkage method is discussed. The sample TDT 3 dataset was clustered using complete, average and single linkage methods. For each of the topics in the ground truth, the best cluster, i.e. the cluster having the minimum cost, was calculated. The rows in table 3.1 show the average characteristics of these best clusters. Without rebranching average pairwise linkage gave the best results by far. Average pairwise linkage scored 55% lower than complete linkage and 71% lower than single linkage clustering methods.

Further investigation showed that single linkage, as expected, performed bad because of its chaining behaviour [JMF99, SBCQ98]. A smaller sample of 100 documents was taken, clustered using single linkage and visualized in a tree (figure 3.4). The figure shows how, especially in the upper part of the tree, new clusters are created by merging an existing cluster and a single document cluster. As a result, the travel cost to reach a more meaningful cluster, i.e. a cluster more closely resembling topics from the ground truth residing at the bottom of the structure, is very high. The travel cost overshadows the detection cost in such a way that the cluster having the lowest overall cost (consisting of travel cost and detection cost) is in the upper part of the structure, although its detection cost is much higher.

The visualization of a cluster structure with 100 documents obtained by using complete linkage (figure 3.5) does not clearly show any chaining behaviour. However, details of the experiment outcome showed that a few clusters were chosen frequently as best matching cluster for a topic, just like the single link cluster structure. A screen shot of a cluster structure browser (Figure 3.6) shows the complete linkage structure also suffers from some kind of "chaining" behaviour. At the root the document set is divided in two clusters: one tight, relatively small cluster with a dissimilarity little less than 1, and one large cluster with a dissimilarity equal to 1. The large cluster is again divided in one small cluster and one large cluster. This continues downwards the tree. The visualization of the structure of 100 documents did not show this behaviour, simply because the dataset is too small. The result, just like the single linkage structure, does not allow the best clusters to be found deep down the clustering tree because of the high travel cost to get there. Some of the best matching clusters found (with a smaller travel cost less influencing the complete cost) were promising however. Table 3.2 gives a sample of the best clusters found for particular topics and its score. The clusters found at depth 2 can be considered as chosen under influence of travel cost - most probably a cluster with a lower detection cost can be found further down the tree. The other clusters have a more promising detection cost, with a recall close to 100% and a precision of around 25%.

The structure obtained by using average linkage seems to be more balanced, naturally enabling more clusters to be considered, not being limited by travel cost. This is one of the major reasons average linkage performs much better when evaluating with the minimal cost metric.

Table 3.2: Sample of best matching clusters using complete linkage

| *System cluster* | *Minimum cost* | *Norm detect. cost* | *Norm travel cost* | *#Ref* | *#Sys* | *#Union* | *Depth* |
|---|---|---|---|---|---|---|---|
| v7102 | 0.6656 | 1.001 | 0.0146 | 5 | 2 | 0 | 2 |
| v7102 | 0.6656 | 1.001 | 0.0146 | 33 | 2 | 0 | 2 |
| v7102 | 0.6656 | 1.001 | 0.0146 | 60 | 2 | 0 | 2 |
| v8514 | 0.2333 | 0.1686 | 0.3588 | 30 | 29 | 25 | 49 |
| v7102 | 0.6656 | 1.001 | 0.0146 | 1 | 2 | 0 | 2 |
| v7102 | 0.6656 | 1.001 | 0.0146 | 4 | 2 | 0 | 2 |
| v7102 | 0.6656 | 1.001 | 0.0146 | 9 | 2 | 0 | 2 |
| v7102 | 0.6656 | 1.001 | 0.0146 | 1 | 2 | 0 | 2 |
| v8933 | 0.5553 | 0.0152 | 1.6036 | 10 | 41 | 10 | 219 |
| v7102 | 0.6656 | 1.001 | 0.0146 | 3 | 2 | 0 | 2 |
| v8500 | 0.1387 | 0.0064 | 0.3954 | 8 | 21 | 8 | 54 |
| v5701 | 0.1454 | 0.0015 | 0.4247 | 1 | 4 | 1 | 58 |
| v7102 | 0.6656 | 1.001 | 0.0146 | 41 | 2 | 0 | 2 |
| v7102 | 0.6656 | 1.001 | 0.0146 | 5 | 2 | 0 | 2 |
| v7102 | 0.6656 | 1.001 | 0.0146 | 9 | 2 | 0 | 2 |
| v7102 | 0.6656 | 1.001 | 0.0146 | 17 | 2 | 0 | 2 |
| v8013 | 0.2758 | 0.1765 | 0.4686 | 12 | 30 | 10 | 64 |

### 3.4.2 Influence of rebranching

The preference of the minimal cost metric for clusters closer to the root of the tree in combination with an unbalanced tree resulted in bad results for complete and single linking. The tree should be balanced without modifying important cluster information. This is done using the rebranching method described before. Table 3.3 shows the minimum cost of the rebranched structures. The dissimilarity thresholds used are adapted to the various methods. The complete linkage causes the dissimilarity to reach 1 quickly for clusters higher in the tree. The threshold is set close to 1 correspondingly. Average linkage will not quickly reach a dissimilarity close to 1, so a threshold value smaller than 1 is chosen. Single linkage suffered so badly under the chaining effect, no threshold value resulted in a cluster structure with its best clusters having a lower minimal cost. Therefore it was decided not to continue further experiments using the single link clustering technique. The rebranching action did not have a large (-6%) impact on the minimum cost for the structure built with average linking. The normalized travel cost however decreased significantly (-65%). As expected the minimum cost for the cluster structure built using complete linking decreased (-50%)as a result of rebranching. A more balanced tree will be searched more thoroughly, i.e. more clusters down the tree are considered, enabling all of the compact clusters to be picked as optimal clusters. As a result the travel cost *and* the detection cost decreased after rebranching the structures built using complete linkage.

### 3.4.3 Influence of matching

It was expected that the complete cluster structures, i.e. the structures obtained by adding the rest of the document dataset to the sample structure, would increase the average minimum cost of the optimal clusters. The contrary was true: adding the new documents to multiple clusters, resulting in a fuzzy cluster structure, improved the results! Table 3.4 shows the cost before and after the matching process. It's particularly interesting that the average detection costs for the TDT 3 and TDT 5 dataset are so different. For the TDT 5 dataset the normalized detection cost and normalized travel cost are in the same order, whereas for the TDT 3 the detection cost is much higher. This might be caused by differences in the dataset, or the way the ground truth was composed.

Furthermore it is noteworthy that the detection cost for the TDT 5 after matching has decreased. The recall has improved (lower $P_{\text{miss}}$ rate) but the precision has gone down (higher $P_{\text{fa}}$). The fuzzy matching is causing this. By adding new documents to multiple clusters, recall will often increase for these clusters. The cost of a false alarm is very low because the dataset is quite large and topics are quite small[3]. Simply guessing related documents for a particular topic using this matching method pays off: the chance an on-target document is guessed is quite large, resulting in a high chance to increase recall, while the cost of a false alarm is low.

### 3.4.4 Intuitiveness

The results do raise questions about the intuitiveness of the metric. For example consider the cluster named 'v18100' in table 3.5. It represents a topic supposedly to have 81 new items, but it actually contains 2826 items, of which 80 actually overlap with the truth cluster. The penalty for missing 1 of the 81 documents is calculated as 0.0123, whereas the false alarm of 2746 items only adds 0.0099 of cost. Just imagine a user trying to find its way through a 'topic' polluted with so many unrelated items. The averages in table 3.5 show the clusters do have a good recall, but its precision is very low. This phenomenon is also apparent in the results of table 3.4: after the fuzzy matching of new documents the the misses decrease but the false alarm rate goes up. The metric allows a large increase of the recall by

[3] the cost of a false alarm is normalized by the chance a random document does not belong to a topic, which is low if the dataset is large and the topics small

adding documents to multiple clusters - the loss in precision is not penalized.

Although the idea behind the introduction of the travel cost is understandable, it does not really penalize 'powerset' structures as it was intended. The travel cost penalizes structures not having the desired branching factor or which are not balanced very well, although the ground truth does not provide any information about this. Another cost component should be introduced to penalize scattering documents over a large number of unrelated clusters as is the case when constructing powerset cluster structures.

## 3.5 Conclusion & further research

In this chapter the results of a prototype HTD system were presented. The usage of conventional agglomerative clustering techniques combined with dissimilarity measurement using language modelling looks promising. The structures built with complete linkage using this distance measurement do need restructuring to be effective however.

The system has been used for participation in the newly introduced HTD evaluation task of TDT 2004 and achieved best results. The intuitive quality of the clusters is questionable however. At this time the results have too little precision to be really useful. The results give thought about the metric used for HTD evaluation. The metric should be studied in more depth to investigate and improve the usefulness of the cluster structures.

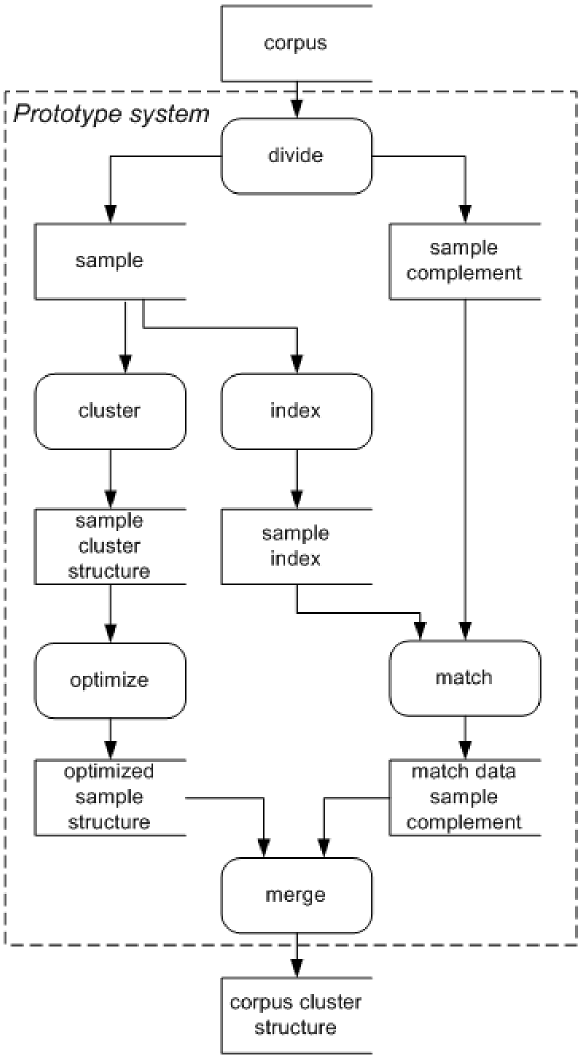
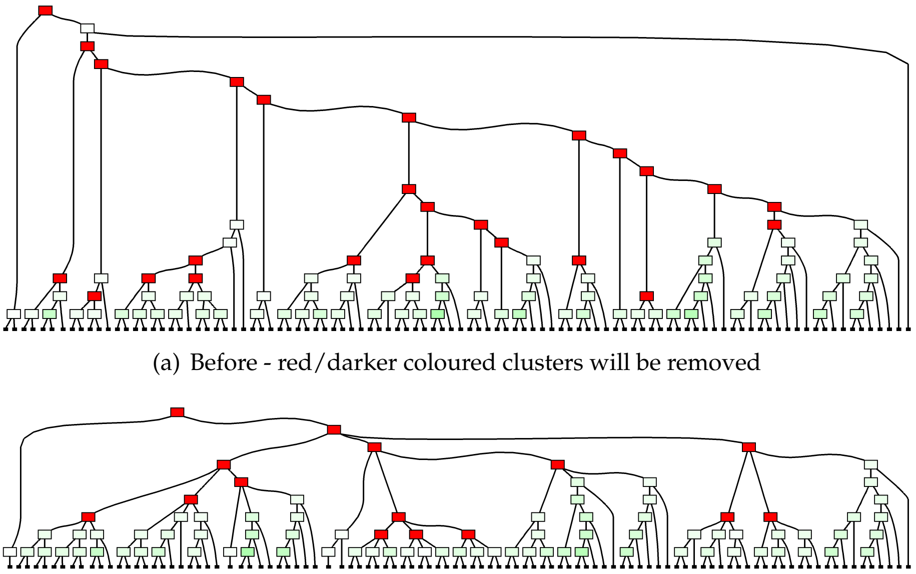


Figure 3.2: Data Flow Diagram

Table 3.3: Influence of rebranching

| *Cluster method (size)* | *Minimum cost* | *Norm. detection cost* | *Norm. travel cost* | *Depth* |
|---|---|---|---|---|
| Average linkage | 0.2747 | 0.3722 | 0.0855 | 11.68 |
| . . . after rebranching (threshold 0.97) | 0.2579 | 0.3620 | 0.0559 | 6.11 |
| Complete linkage | 0.612 | 0.8778 | 0.0962 | 13.14 |
| . . . after rebranching threshold 1.0) | 0.3497 | 0.5006 | 0.0567 | 5.89 |



(a) Before - red/darker coloured clusters will be removed

(b) After - red/darker coloured clusters are new

Figure 3.3: The result of rebranching

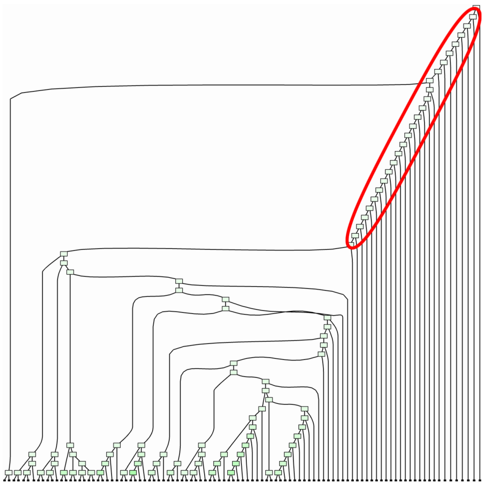


Figure 3.4: Single link clustering suffers from chaining

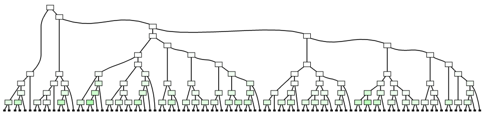
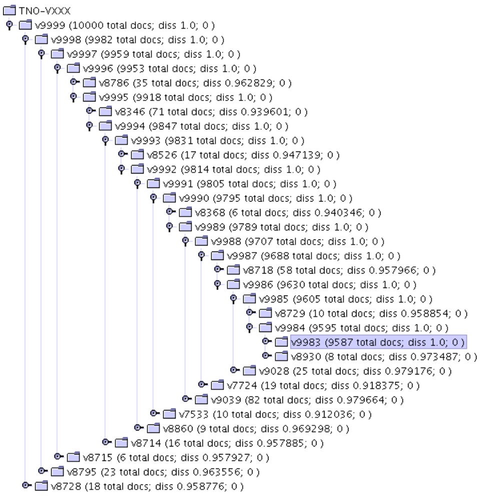


Figure 3.5: Complete link clustering



```
TNO-VXXX
  v9999 (10000 total docs; diss 1.0; 0 )
    v9998 (9982 total docs; diss 1.0; 0 )
      v9997 (9959 total docs; diss 1.0; 0 )
        v9996 (9953 total docs; diss 1.0; 0 )
          v8786 (35 total docs; diss 0.962829; 0 )
          v9995 (9918 total docs; diss 1.0; 0 )
            v8346 (71 total docs; diss 0.939601; 0 )
            v9994 (9847 total docs; diss 1.0; 0 )
              v9993 (9831 total docs; diss 1.0; 0 )
                v8526 (17 total docs; diss 0.947139; 0 )
                v9992 (9814 total docs; diss 1.0; 0 )
                  v9991 (9805 total docs; diss 1.0; 0 )
                    v9990 (9795 total docs; diss 1.0; 0 )
                      v8368 (6 total docs; diss 0.940346; 0 )
                      v9989 (9789 total docs; diss 1.0; 0 )
                        v9988 (9707 total docs; diss 1.0; 0 )
                          v9987 (9688 total docs; diss 1.0; 0 )
                            v8718 (58 total docs; diss 0.957966; 0 )
                            v9986 (9630 total docs; diss 1.0; 0 )
                              v9985 (9605 total docs; diss 1.0; 0 )
                                v8729 (10 total docs; diss 0.958854; 0 )
                                v9984 (9595 total docs; diss 1.0; 0 )
                                  v9983 (9587 total docs; diss 1.0; 0 )
                                  v8930 (8 total docs; diss 0.973487; 0 )
                              v9028 (25 total docs; diss 0.979176; 0 )
                          v7724 (19 total docs; diss 0.918375; 0 )
                        v9039 (82 total docs; diss 0.979664; 0 )
                    v7533 (10 total docs; diss 0.912036; 0 )
                  v8860 (9 total docs; diss 0.969298; 0 )
              v8714 (16 total docs; diss 0.957885; 0 )
        v8715 (6 total docs; diss 0.957927; 0 )
      v8795 (23 total docs; diss 0.963556; 0 )
    v8728 (18 total docs; diss 0.958776; 0 )
```

Figure 3.6: 'Chaining' behaviour of complete linkage clustering

Table 3.4: Influence of matching on average costs

| *Cluster method (size)* | *Minimum cost* | *Norm. detection cost* | *Norm. travel cost* | *P(miss)* | *P(fa)* |
|---|---|---|---|---|---|
| TDT3 sample (10,000) | 0.2579 | 0.3620 | 0.0559 | 0.3069 | 0.0112 |
| . . . after matching (35,000) | 0.2430 | 0.3581 | 0.0195 | 0.2681 | 0.0184 |
| TDT5 sample (20,000) | 0.0565 | 0.0629 | 0.0441 | 0.0493 | 0.0028 |
| . . . after matching (278,000) | 0.0282 | 0.0406 | 0.0041 | 0.0224 | 0.0037 |

Table 3.5: Sample results from one complete TDT 5 cluster structure

| *System cluster* | *Minimum cost* | *Norm detect. cost* | *Norm travel cost* | *#Ref* | *#Sys* | *#Union* | *P(miss)* | *P(fa)* |
|---|---|---|---|---|---|---|---|---|
| v13965 | 0.0039 | 0.0045 | 0.0028 | 5 | 261 | 5 | 0 | 0.0009 |
| v15445 | 0.0023 | 0.0023 | 0.0022 | 1 | 133 | 1 | 0 | 0.0005 |
| v14140 | 0.0024 | 0.0019 | 0.0035 | 27 | 133 | 27 | 0 | 0.0004 |
| v16401 | 0.0095 | 0.0131 | 0.0025 | 13 | 759 | 13 | 0 | 0.0027 |
| v18100 | 0.0411 | 0.0607 | 0.0031 | 81 | 2826 | 80 | 0.0123 | 0.0099 |
| v3969 | 0.0013 | 0.0004 | 0.0031 | 1 | 24 | 1 | 0 | 0.0001 |
| v5859 | 0.0019 | 0.0012 | 0.0032 | 2 | 71 | 2 | 0 | 0.0002 |
| v1076 | 0.0029 | 0.0018 | 0.0051 | 1 | 104 | 1 | 0 | 0.0004 |
| v3440 | 0.0019 | 0.0013 | 0.0031 | 2 | 76 | 2 | 0 | 0.0003 |
| v9072 | 0.0094 | 0.0117 | 0.0050 | 21 | 683 | 21 | 0 | 0.0024 |
| v2590 | 0.0017 | 0.0005 | 0.0042 | 1 | 28 | 1 | 0 | 0.0001 |
| v8772 | 0.0448 | 0.0664 | 0.0030 | 63 | 223 | 59 | 0.0635 | 0.0006 |
| v17828 | 0.0016 | 0.0009 | 0.0030 | 1 | 50 | 1 | 0 | 0.0002 |
| v15435 | 0.0065 | 0.0073 | 0.0051 | 2 | 417 | 2 | 0 | 0.0015 |
| v17092 | 0.0037 | 0.0042 | 0.0028 | 5 | 241 | 5 | 0 | 0.0008 |
| … | … | … | … | … | … | … | … | … |
| average | 0.0282 | 0.0406 | 0.0041 | 43.15 | 1073.1 | 42.05 | 0.0224 | 0.0037 |

Chapter 4

# Evaluation metric

As became clear in the discussion of the prototype, evaluating a hierarchical cluster structure is not straightforward. In this chapter the expectations of a hierarchical topic structure and the minimal cost metric used for the TDT evaluation are discussed. Improved metrics and a visualization method are proposed useful for evaluating a hierarchical cluster structure and a flat ground truth.

## 4.1 Expectations of hierarchical topic structures

The HTD task of the TDT programme does not precisely describe what is intended with a hierarchical topic structure. It allows "clusters to be defined at different levels of granularity" [NIS04]. How this corresponds to the definition of a topic (see page 21), remains unanswered. A topic is based on a seminal event and contains all documents describing this and directly related events or activities. A larger corpus (for example spanning a larger time frame) in general will describe more seminal events and this will result in more (TDT) topics. Following the rules of interpretation of the LDC this results in a fuzzy, but flat structure of topics. How this flat structure becomes hierarchical is left unanswered by the TDT task. Allowing a hierarchical structure at this point looks like allowing more opportunities or guesses to approximate the ground truth. A system has some method of clustering documents, at some level documents are joined, but as it doesn't "know" where a topic begins or ends it may show its path of "reasoning" so the evaluation can show the system is on the right track or not. Such an evaluation shows the potential of the system, but the hierarchy will not be of use to a user who wants to browse a large collection of documents. The hierarchy is then a means of evaluation and not a browsable structure, as the evaluation metric does suspect.

Assuming it is possible to cluster documents in TDT topic clusters, it would be desirable to group these clusters at some higher level. A natural continuation of the cluster process would be grouping the topic clusters in clusters representing some higher notion of topic. The clusters at a higher level bring some abstraction by grouping a reasonable number of clusters which have some characteristic in common. If a significant number of topics discuss accidents, a valuable cluster might group these various accidents, abstracting from the type of accidents or the people involved.

Not only is this form of abstraction hard to realize – ask two different persons to organize a set of documents in a hierarchy and the resulting structures are bound to differ – fully automated evaluation of these abstract hierarchies seems almost impossible.

It would be useful if the quality of a cluster structure at least partially can be assessed, given a flat ground truth provided by the LDC. The evaluation then becomes answering the following question:

> To what extent does this topic structure represent particular topics?

In the following sections the minimal cost metric is discussed to find the shortcomings of this evaluation method. After that improvements are presented.

## 4.2 Minimal cost metric

### 4.2.1 Construction of the ground truth

Before describing how the ground truth is put to work, the annotation method used by the LDC is mentioned [Lin04].

The annotation manual describes the rules of interpretation stating thirteen types of seminal events, e.g. "crimes" or "natural disasters". For each of these types is described what is the scope of related events and activities. Furthermore the manual stresses that a TDT topic discusses only one specific event, whereas in normal discourse you might expect a topic to be more general.

The ground truth is only a partial annotation of the corpus. Random documents are used as seed for a particular topic. This seed story is used to determine the seminal event and a search engine is used to find related stories. The resulting topics strongly vary in size and detail level.

An important consequence of this method is that the chosen seed strongly influences the resulting topic. If a random document from an annotated topic is used as a new seed, this can result in a different (but probably overlapping) topic. It's not clear how particular document seeds are treated, for example a news item which discusses two distinct seminal events.

### 4.2.2 Motivation of the minimal cost metric

Fiscus [FD02] describes the TDT as a detection task in which "the system is presented with input data and a hypothesis about the data, and the system's task is to decide whether the hypothesis about this data is true". The hypothesis is true when a document belongs to a particular topic and false if it does not. This leads to a number of target and non-target *trials*, i.e. a test to determine if a document is on- or off-topic respectively. The detection cost of a particular cluster for a particular topic indicates how well the cluster represents the topic.

Allan et al [AFB03] note the new fuzzy hierarchical structure introduces some new challenges. The fuzzy structure allows documents to be assigned to multiple clusters. A "powerset" structure, i.e. a structure in which each or many possible sets of documents have its own cluster, would achieve a perfect detection cost when searching for the best matching cluster. The hierarchy allows to define relationships across clusters and allows splitting a topic into multiple clusters. The metric should incorporate this hierarchy in a meaningful fashion. The travel cost is introduced to deal with the hierarchy and fuzziness of the cluster structure.

### 4.2.3 Shortcomings of the minimal cost metric

At a first glance the evaluation methodology of the TDT seems clear and intuitive: human annotators build a ground truth and system cluster structures are evaluated by scoring its resemblance. By following this approach and in particular the minimal cost metric as an evaluation mechanism several hypotheses about the desired structure and usage are adopted. The minimal cost metric is parametrized, i.e. the impact of the components can be adjusted using parameter values. Using a particular set of parameters influences the properties of the desired cluster structure and evaluates a particular task.

The travel cost component is introduced as some kind of usability cost: it's attractive to have a hierarchical structure which brings you quickly from the root to the desired cluster. It implies the cluster structure is used for some kind of browsing: a user starts at the root cluster and descends down the tree towards the best cluster.

Looking for the best cluster can be compared to a user with a very specific information need and finding out if a cluster exactly meets this need. If a hierarchical structure is useful for finding very specific information is questionable. Pirolli et al [PSHD96] showed that the Scatter/Gather cluster hierarchy did not increase the speed of finding specific information compared to a 'common word-based search'. Having a small topic in mind and finding this information in a collection of $400,000$ documents by browsing the hierarchy from the root cluster, would be like finding a needle in a haystack. On the other hand it would be interesting if the structure allows finding such a topic or clearly outlines such a topic.

In the following sections some of the shortcomings of the current approach are discussed.

### Combined cost

The best cluster is defined as the cluster with the lowest combined detection and travel cost: a user considers both the quality of the cluster found and the effort of finding it. The combined cost is a linear combination of the two components: a high combined cost is caused by the high contributions of one or both cost components. A cluster having a high travel cost will result in a high combined cost and influences if this cluster is the best cluster for a particular topic. The contribution of the travel cost can cause a cluster not to be chosen as best cluster, despite a possibly low detection cost for a particular topic. This can be seen as a user reluctant of finding a better cluster, after trying all clusters with a travel cost below a certain effort. If this reluctant behaviour is desirable in an evaluation is questionable. The evaluation should tell if a cluster structure effectively outlines a topic and besides that note that finding the most effective cluster required a certain effort, translated into some (travel) cost. Not even considering particular clusters "as they require too much effort" restrains a possibly very effective cluster structure from performing well. As a result the evaluation cannot discriminate between an ineffective cluster structure and an effective but "high-effort" cluster sructure.

### Shortest path representing effort

During evaluation the cluster structure is thoroughly searched for a cluster with the lowest combined cost for a particular topic. The path from the root which leads directly to the best cluster is used for calculating the travel cost, representing the effort of finding the cluster. It's questionable if this effort is directly related to the shortest path. A shorter path to reach the same cluster does not always imply the effort to find that cluster is smaller as the shortest path may not be the actual search path of a user. This effort might be stronger related to the informative value of the labels of the clusters and a clear categorisation.

### Hierarchy preferences

The travel cost favours balanced cluster structures with a particular branching factor, although no evidence is provided that a balanced hierarchical structure indeed is better. It's quite likely that the documents in a news archive are not equally divided in high level topics. This requirement encourages artificial approaches to achieve a better score, such as the rebalancing procedure used for the prototype. The structure obtains a better score by applying this operation, without using any knowledge of the document content. The following example shows how a suboptimal solution increases performance according to the metric.

**Example.** Assume a cluster structure $S$ and a ground truth $G$. The incomplete ground truth contains $g$ topics. $S$ has one root cluster $c_r$, which has $g+1$ child clusters, $c_1$ to $c_{g+1}$. Clusters $c_1$ to $c_g$ exactly match the topics $t_1$ to $t_g$:

$$\forall n \in 1..g \bullet t_n = c_n$$

Cluster $c_{g+1}$ contains the documents not annotated by the ground truth, as a result $S$ covers the complete document collection:

$$c_{g+1} = D \backslash \bigcup_{t \in G} t$$

The detection cost for each of the best clusters is $0$, as for each topic an exact match can be found. The travel cost for each of these clusters is:

$$(g+1) \cdot \text{CBRANCH} + \text{CTITLE}$$

In general the number of topics $g$ is much larger than the preferred branching factor (3 for TDT 2004). The clusters $c_1$ to $c_{g+1}$ can be used as leaf clusters in a balanced tree structure with a branching factor of 3. As this structure has a depth of $\lceil \log_3(g+1) \rceil$ the travel cost for the optimal clusters (still having a detection cost of $0$) is:

$$\lceil \log_3(g+1) \rceil \cdot (3 \cdot \text{CBRANCH} + \text{CTITLE})$$

For CBRANCH $= 2$ and CTITLE $= 1$ (the values used for the TDT evaluation) this gives for the restructured example:

$$7 \cdot \lceil \log_3(g+1) \rceil$$

For the original structure this yields a cost of:

$$(g+1)\cdot \text{CBRANCH} + \text{CTITLE} = 2\cdot g + 3$$

For all $g > 5$ this gives a lower or at most equal travel cost for the restructured sample. This shows that by adding dummy clusters to an optimal solution a lower score can be obtained, which is clearly not desirable.

We doubt if a hierarchy should have a preferred branching factor, or should be balanced. The Open Directory Project (DMOZ), a hierarchical categorisation of a collection of websites created by human editors, does not have a certain 'balancedness' or branching factor (see appendix A). Assuming the news items in the corpus are as diverse as the websites in the DMOZ, a preference for a particular branching factor or "balancedness" is not justified.

**Linear travel cost**

For the moment we will assume the user is capable of descending the cluster structure towards the best cluster directly. Furthermore we will assume the cluster structure is, as desired by the metric, perfectly balanced and has the desired branching factor.

Starting from the root cluster, a user has to choose from a limited number of clusters; choosing one cluster brings the user one step closer in finding the desired cluster. By choosing this cluster, the other clusters and its child clusters are discarded. So with a branching factor of 3, a selection is made of one third of the documents contained by the root cluster. Two third of the documents contained by the root cluster are discarded. Every next choice for a particular cluster reduces the number of 'selected' documents with a factor 3 (the branching factor). For each choice in this search path a constant amount of travel cost is added.

The revenue (the number of discarded documents) of later choices in the search path is smaller, but the cost (a choice) remains the same. So it can be argued that these choices are more expensive and that the travel cost should increase stronger for later choices in the search path.

This can also be explained from the viewpoint of possible number of clusters at a certain depth. Assuming a fixed branching factor of $b$, the root cluster has $b$ possible clusters at depth 1. These child clusters all can have $b$ children, so the structure has $b^2$ possible clusters at depth 2. A cluster structure with depth $n$ has $b^n$ possible clusters at depth $n$.

The travel cost of a cluster found at depth $n$ is:

$$\mathrm{C_t}(n) = (\text{CTITLE} + b\cdot \text{CBRANCH})n$$

When the depth of the cluster structure is increased, the number of possible clusters and the travelcost change as follows:

$$\begin{aligned} \Delta b^n &= b^{n+1} - b^n = b^n \cdot b - b^n = (b-1) \cdot b^n \\ \Delta\, \mathrm{C_t}(n) &= \mathrm{C_t}(n+1) - \mathrm{C_t}(n) = (\textsc{ctitle} + b \cdot \textsc{cbranch}) \end{aligned}$$

So increasing the depth of a cluster structure with one, especially for clusters which are already deep, pays off. Many more 'guesses' are allowed with only a small linear increase in travel cost.

### Discriminating powerset and skewed cluster structures

The travel cost is introduced to penalize powerset cluster structures [AFB03], but it actually cannot determine this. A cluster structure tending more towards a powerset solution does indeed have a higher travel cost, as more clusters also require a larger width and height with corresponding branching cost and title cost respectively. But as the travel cost only looks at the structure of the shortest path from root to cluster, i.e. the length of the path and the branching factor of the clusters in the path, it cannot actually determine the difference between a skewed cluster structure and powerset cluster structure. The following example illustrates this.

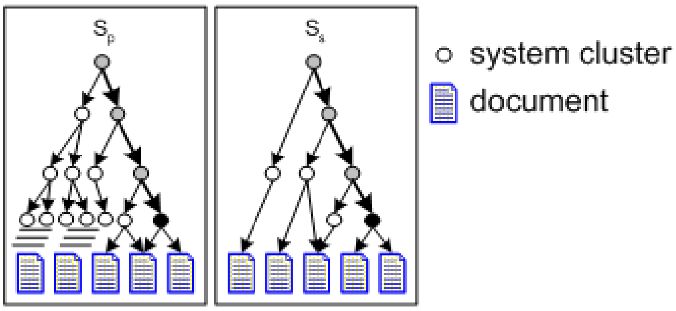


Figure 4.1: Travel cost of a powerset and skewed cluster structure

**Example.** Assume two cluster structures $S_p$ (a "powerset" cluster structure)and $S_s$ (a skewed cluster structure) for a particular document collection $D$. Both structures are evaluated for a particular topic $t$ from a ground truth. Figure 4.1 shows the best two clusters found for $S_p$ and $S_s$, indicated as a black dots. The gray dots indicate the clusters on the path to the best cluster. Both paths have a length of 3 (resulting in $3 \cdot$ CTITLE) and 6 branches are encountered (resulting in $6 \cdot$ CBRANCH). Although the resulting travel cost for both clusters is the same, the powerset cluster structure $S_p$ allows far more possible clusters, far more "guesses" for each of the topics.

### 4.2.4 Conclusion

Modelling the user effort of finding a cluster seems intuitive but the implementation in the minimal cost metric doesn't work out as intended. The travel cost imposes strong restrictions or preferences on the structure, which do not always correspond to the expectations of a good cluster structure. Using a combined cost as best cost limits the information provided by the evaluation; possibly effective cluster structures can be judged as not valuable if the travel cost of the effective clusters is too high.

Major difficulties in defining an evaluation metric for hierarchical topic detection are the unclear purpose and following expectations of the cluster structures, as discussed in section 4.1. Maybe this is because it's also unclear what can be expected from current HTD techniques.

Defining a very strict evaluation with a preference for particular cluster structures does not seem to contribute to solving these difficulties. The evaluation should aid in exploring what is technically possible and stimulate discussion about what is desired and what is expected of hierarchical topic structures. A vague task should have a "vague" metric accordingly. Therefore is argued to use separate indicators to evaluate cluster structures. The detection cost and travel cost should be used and evaluated separately. Other indicators are required to evaluate the true "powerset" tendency and complexity of the cluster structure. In the following section these additions and improvements are further discussed.

## 4.3 Revised evaluation metric

During the evaluation a topic is compared to a cluster structure. A cluster structure might have a single cluster exactly matching a topic cluster, or multiple clusters partially matching the topic cluster.

We are primarily interested in the *topicrelevant clusters* of a topic.

**Definition.** *Topicrevelant clusters* for a particular topic, $R(t)$, are the clusters from a cluster structure $V$ which (directly or indirectly) contain one or more documents from the topic $t$:

$$R(t) \quad = \quad \{c \mid c \in V \land \exists d \in t \bullet d \in D^*(c)\}$$

Note that the root cluster is topicrelevant for all topics, as it contains all documents including at least one document from a topic. Furthermore all topicrelevant clusters are connected via some path to the root cluster. The topicrelevant clusters and their relationships form a *natural subgraph* of the original structure.

The group of topicrelevant clusters can be examined to see how the cluster structure represents a particular topic. If a topic has many topicrelevant clusters, the topic might not be distinguishable in the structure. Multiple aspects are of interest, which will be described in the following sections.

### 4.3.1 Effectiveness

The effectiveness of each topicrelevant cluster is of interest, which is based on the overlap between the cluster and topic. The root cluster will have a recall of 100% but a low precision (typically even approaching 0%) as the topic will not be the complete document collection. Other topicrelevant clusters further down the structure will have a higher precision, but probably at the expense of recall. Following the current evaluation of TDT a preference can be adopted for a particular trade off between recall and precision.

### 4.3.2 Fuzziness

The new hierarchical cluster structure allows documents to be part of multiple clusters and clusters to be children of multiple parents. This “fuzziness” is an indicator of the complexity of a cluster structure. The complexity of a topic can be related to the paths from the root cluster to each of the individual topicdocuments.

**Definition.** The *documentfuzziness* of a document, $df(d)$, is the number of paths from the root cluster $c_r$ to a document $d$.

The documentfuzziness is at least 1; a higher value indicates the document can be reached in multiple ways.

**Definition.** The *topicfuzziness*, $tf(t)$, is the average documentfuzziness of the documents in a topic $t$.

The topicfuzziness is at least 1; a higher value indicates the documents in the topic can be found in multiple ways.

### 4.3.3 Hierarchy above and below

A larger number of topicrelevant clusters already signals the topic is not represented straightforward, but also the structures above and below topicrelevant clusters are of interest. A cluster might be very effective, but if the cluster or its topicdocuments can only be reached via a complex web of clusters, this might not be the desired situation.

As indicators of the complexity of the hierarchy above a cluster, we use the *average root path length* and the *average encountered branches* of the cluster. Note this indicators were also used to calculate the travel cost.

**Definition.** The average root path length of a cluster, $ap(c)$, is the average length of the paths from the root cluster $c_r$ to cluster $c$.

**Definition.** The average encountered branches of a cluster, $eb(c)$, is the average number of encountered branches of all the paths from the root cluster to cluster $c$.

As indicators of the complexity of the hierarchy below a cluster, we can look at the number of steps needed to reach each of the topicdocuments. Furthermore the number of distinct paths to a topicdocument indicates the complexity of the hierarchy below a cluster. We use the *average path length* to the topic documents of that cluster and the number of paths from this cluster to the topicdocuments as indicators of the complexity below a cluster.

## 4.4 Visualizing topic evaluations

Having a table of the previously mentioned indicators for a number of topicrelevant clusters is useful, but a visualisation of this data can bring more insight how the clusters are actually connected.

The natural subgraph spanned by the topicrelevant clusters can be displayed to provide this information. In this graph the relationship between topicdocuments and clusters can be visualized. Furthermore the relationship between parent and child clusters can be shown. As the structure can be fuzzy, a topicdocument can be directly contained in multiple clusters. This fuzziness can also be displayed by connecting the topicdocuments contained by topicrelevant clusters.

As the paths can become quite long, we need some way to group less significant clusters. Significant clusters are defined as follows:

**Definition.** A significant topicrelevant cluster has a topicdocument directly attached, or has at least two child clusters which are topicrelevant, i.e. the cluster merges two or more topicrelevant clusters. The root cluster is always significant.

Chains of not significant topicclusters can be represented as a single node to save space.

Figure 4.2 shows an example of such an visualization:

- The elipses represent relevant topicclusters, identified by their name (and dissimilarity), total number of documents and the number of matching topicdocuments (union). The colour indicates the effectiveness of the topiccluster; a greener/darker colour implies a lower detection cost.
- The rectangles represent (groups of) topicdocument(s) identified by their name.
- The triangles represent groups of irrelevant topicclusters.
- The edges indicate parent child relationships between clusters.
- The dotted edges between rectangles containing topicdocuments indicate there is overlap between the two groups.

### 4.4.1 Algorithm

One of the advantages of the algorithm used for the minimal cost metric, is the pruning of unnecessary searches (see page 25). The proposed indicators cannot benefit from such an approach, as *all* topicrelevant clusters need to found. Therefore a bottom-up algorithm is suggested. Starting from the topic documents, the topicrelevant clusters can be found which directly contain one or more topic documents. This set of clusters can repeatedly be expanded with the parents of the clusters in the set, until no more parents can be added. This set of clusters then contains all the topicrelevant clusters. Experiments showed this algorithm outperformed the pruning tree search used for the minimal cost metric for the cluster structures created using the prototype [1].

## 4.5 Conclusion

Evaluating hierarchical topic structures is difficult with only a flat ground truth available; the value of higher level clusters is hard to determine. Imposing strong restrictions as in the minimal cost metric results in a bias for particular suboptimal solutions. Looking for only one particular “best” cluster seems a limited approach when the structure offers far more potential.

We argue for evaluating a selection of clusters when comparing a hierarchical cluster structure to a ground truth topic. This selection of clusters

[1] Running the minimal cost evaluation algorithm on the prototype TDT 5 cluster structures took around one complete day; the bottom-up approach took around 20 minutes on a GHz workstation with 1 GB of working memory

can be evaluated using multiple but seperated indicators which give an extensive view of the performance of the cluster structure. We have chosen not to combine these indicators in a new single value score, as it again will present a preference for a very particular type of cluster structure while at this time the requirements for a hierarchical topic structure seems unclear. As a result it is not possible to tweak the structure for a minimal overall cost and forces a discussion of the obtained indicator values and a desired cluster structure.

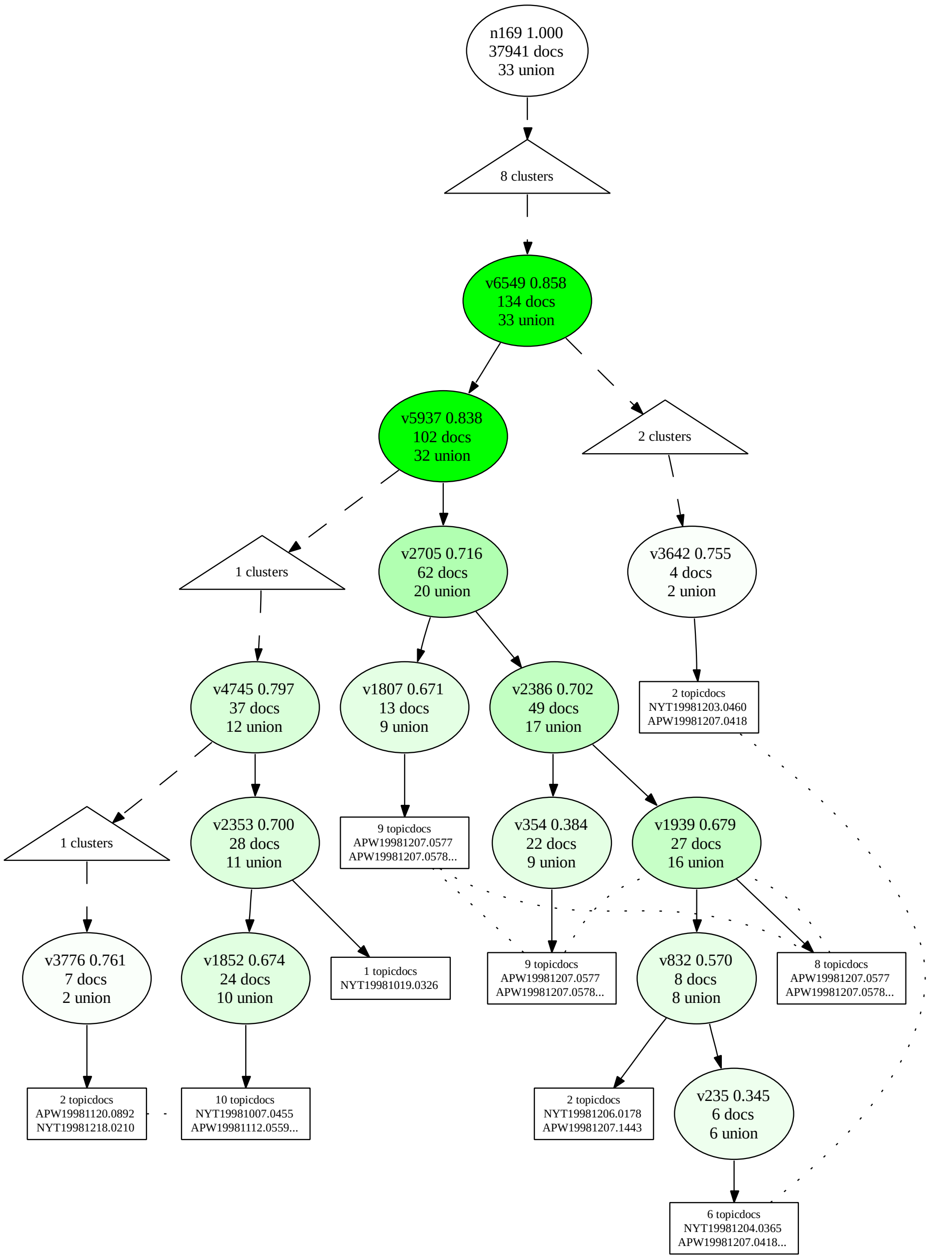


Figure 4.2: Visualisation of topicclusters

Chapter 5

# Experiments

In this chapter the indicators and visualization introduced in the previous chapter are used to underline the results obtained during development of the prototype (chapter 3). The chapter serves as an example how to use the indicators for an evaluation of a cluster structure. The evaluation results have been added in Appendix D.

## 5.1 Linkage methods

During development of the prototype system was observed that according to the minimal cost metric, single linkage performed worst, followed by complete linkage and the best results were obtained by using average pairwise linkage. It is expected this order will not change, but why does average linkage outperform the others?

### 5.1.1 General

Before the three different methods are discussed, a note is made on the result of a HAC method in general.

**Effectiveness**

As the result of HAC is a binary tree, a cluster structure of $n$ documents requires $n - 1$ clusters to merge them to one root cluster. A cluster has two children: either a pair of two document clusters, a pair of two clusters, or a combination of one cluster and one documentcluster. Singleton topics, i.e.

topics containing only one document will always have a matching document cluster and "perform well" judging from the effectiveness indicators.

### Fuzziness

Documents can only be assigned to one cluster, so there is no fuzziness at all in a cluster structure obtained from applying HAC. The matching process does introduce fuzziness, which will be further discussed in section 5.3.

### Hierarchy

The hierarchy above effective topicrelevant clusters has paths of varying lengths to the root cluster. Each cluster has only one path to the root cluster. If the cluster structure is completely balanced, the path to reach the root will be of length $\lceil \log_2(n/m) \rceil$, where $n$ is the size of the collection and $m$ is the size of the cluster. The path length to the root can vary between 1 (a cluster as a directly under the root) to $n - m$ (a completely unbalanced cluster structure). The number of encountered branches is, as the cluster structure is a binary tree, the double of the path length.

The value of the indicators of the hierarchy below topicrelevant clusters depends on the size of the cluster. A cluster containing $n$ documents has $n-1$ (grand)child clusters to group the $n$ documents. The average path length to reach topicdocuments from this cluster (dependent on its balancedness) will grow with the size of the cluster. The average path length to reach topicdocuments will also grow as a result of a less effective cluster; more documents require more clusters in a binary structure and the paths to reach topic documents will grow accordingly.

Without looking at the actual result of a HAC, we can already note the hierarchy will not be very useful for browsing. Assuming HAC is capable of grouping documents in topics effectively, the structure cannot clearly outline a topic as it is bound to the binary branching.

Figure 5.1 shows a cluster structure which contains a cluster with an almost perfect effectiveness. The chain of 2609 clusters between the root cluster and the first significant topicrelevant cluster make identification of this distinct topic hard, although this cluster effectively matches the topic.

## 5.1.2 Single linkage

The visualization of a 100 documents sample clustered using single linkage already showed single linkage suffered from chaining (see figure 3.4 on page 35). Visualization of a larger cluster structure (with 20,000 documents) is not possible in the same way however.

Only visualizing the subgraph containing topicrelevant clusters for each of the topics from the TDT 3 evaluation does show this behaviour. The clusters form a connected chain, separated by long paths of non significant clusters (paths represented by triangles). Some characteristic visualizations are shown in figure 5.2.

The chaining behaviour is noticeable from the proposed indicator values (see Appendix D). The average number of encountered branches and average path length to reach the best topicrelevant clusters are very high if a user has to browse the structure (an average length of almost 3000 clusters and 6000 branches to consider for one topic). Also the number of topicrelevant clusters for a topic is high: with an average of almost 3500 clusters per topic with an average size of 30.

### 5.1.3 Complete linkage

The visualization of the sample clustered using complete linkage shows the method produces tight groups of clusters, which are combined at a higher level. Complete linkage defines the distance between the newly created cluster and remaining clusters as the minimal distances between the merged clusters and the remaining clusters (see section 2.2.3). As a result the relationship between newly created clusters and remaining clusters quickly diminishes. Clusters high in the tree have a dissimilarity with the maximum value of 1; clusters at this level were not "motivated" by some similarity, but there simply is not a more similar pair at that time to merge.

Figure 5.3 shows some typical visualizations of subgraphs containing topicrelevant clusters. The topics used for these visualizations correspond to the ones used for figure 5.2.

The indicator values show the average recall and precision of the most effective clusters are both around 50%, resulting in an average detection cost of 0.01 (see Appendix D). This gives a different view of the effectiveness compared to the TDT evaluation (see average results in table 3.1 on page 29): during the minimal (combined) cost evaluation the average detection cost component was 0.0176. Comparing to the single link cluster structure, many of the indicator values are lower (the average root path length is 90% shorter and the number of topicrelevant clusters 78% lower). Surprisingly the average effectiveness of the best clusters from the complete link cluster structure is comparable (4% lower) to the clusters from the single link cluster structure. The evaluation using the minimal cost metric did not show this resemblence. With modification the obtained cluster structures using single linkage, might achieve similar results as complete linkage.

### 5.1.4 Average linkage

The visualizations of average pairwise link look like a combination of single and complete linkage: compact groups of clusters of various sizes are linked in chains (see figure 5.4).

Interestingly the visualization shows the average link cluster structures contains more effective topicrelevant clusters (more clusters are coloured stronger; this indicates a lower detection cost).

Average pairwise linkage outperforms single and complete linkage in almost all indicator values. Only the precision of the average pairwise linkage method (30%) is worse (20%. lower than complete linkage and 14%. lower than single linkage). With an average path length of 15 from root cluster to most effective topicrelevant clusters, the indicator value shows the average pairwise link cluster structure is more naturally balanced for the evaluated topics. If a path of 15 clusters and 30 branches to consider is still userfriendly is questionable. The hierarchy below the best clusters might be too complex, as the paths to topic documents are long.

## 5.2 Rebranching

As intended with the rebranching operation, only the top of the cluster structure is changed. As a result the average path lengths from the root to topicrelevant clusters decreases as well as the average number of encountered branches. The visualization of the rebranching operation does not bring any new information – it merely decreases a chain of non significant topicclusters (a triangle shaped node gets a different label).

## 5.3 Matching

The matching method influences fuzziness of the cluster structure, as new documents can be added to multiple sample clusters, resulting in a higher documentfuzziness for these documents. If a topicdocument is added to multiple (grand)child clusters of a topicrelevant cluster, the document can be reached via multiple paths. The visualization of the subgraph containing topicrelevant shows that sometimes this matching process leads to bizarre results. Figure 5.5 shows a very fuzzy cluster structure.

## 5.4 Conclusion

The separated indicators and visualization introduced in the previous chapter give a new perspective on the prototype cluster structures previously evaluated using the minimal cost metric. Average pairwise linkage still outperforms complete and single linkage in the complexity of the hierarchy above the best topicclusters: the paths from the root cluster are shorter and less branches have to be considered. Also in effectiveness the average link method outperforms the others, but it should be noted only a particular trade off between recall and precision was evaluated in which recall was preferred over precision. A different trade off might give a different result.

The evaluation shows the cluster methods have a tendency to group documents in topiccoherent groups, although especially precision is not very high. The cluster structure above and below do not seem very useful however; independent of clustering method the topicrelevant clusters require long paths (on average longer than 15 clusters) from the root and long paths to topicdocuments from these clusters. The increased documentfuzziness does not seem to bring more clarity in topicrelevant clusters: they provide multiple paths to locate the same documents.

One important next step which should be explored is how the binary cluster structure can be made useful by restructuring the upper and lower cluster structures. Interesting would be to find out if the documentfuzziness might gives information about which clusters are too detailed and should be merged with its parent.

Another interesting question is the effect of the document modelling and distance metric. Does a system based on the Dice or Cosine distance measure give topicrelevant clusters with the same effectiveness?

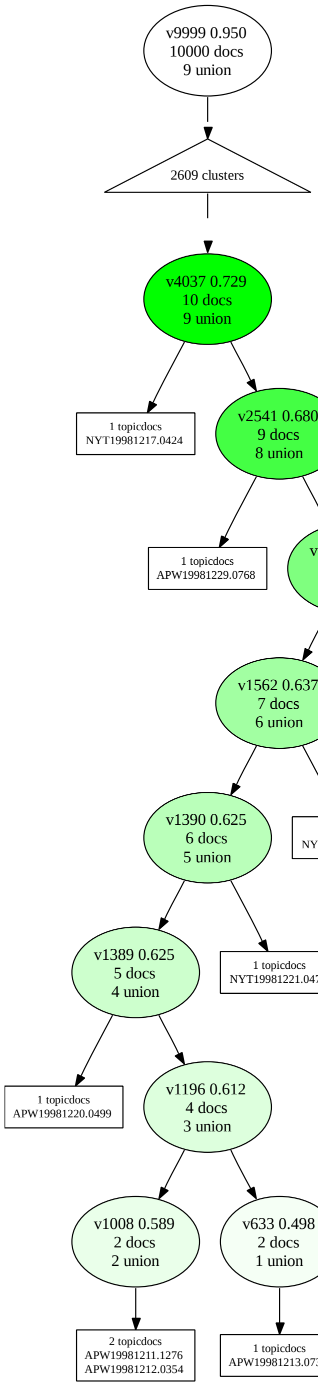
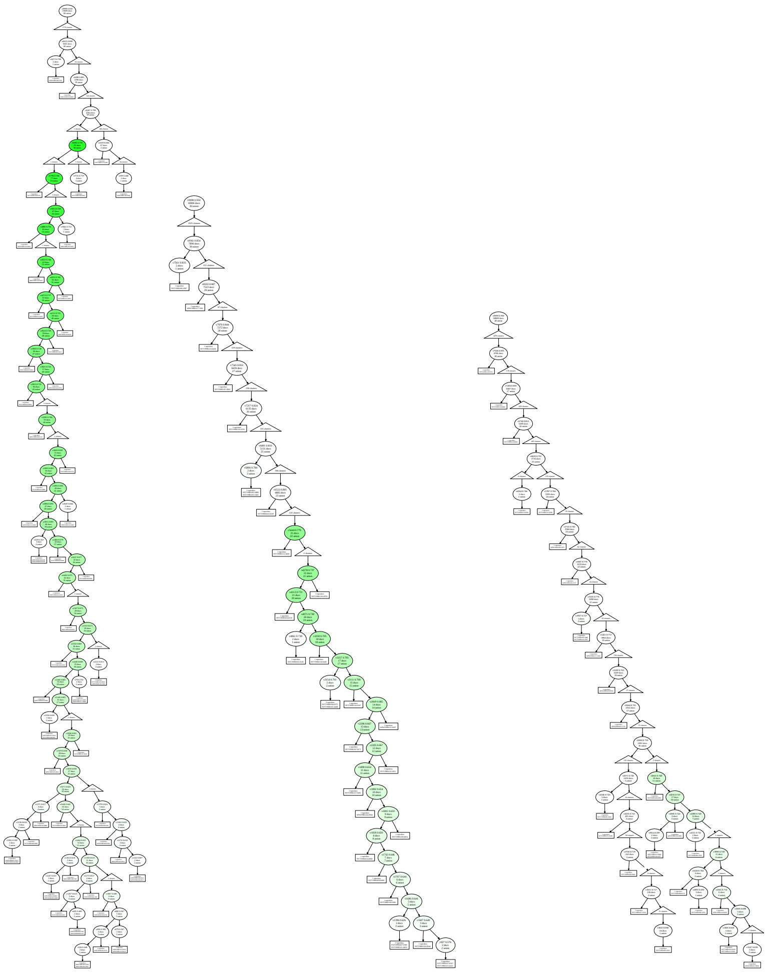
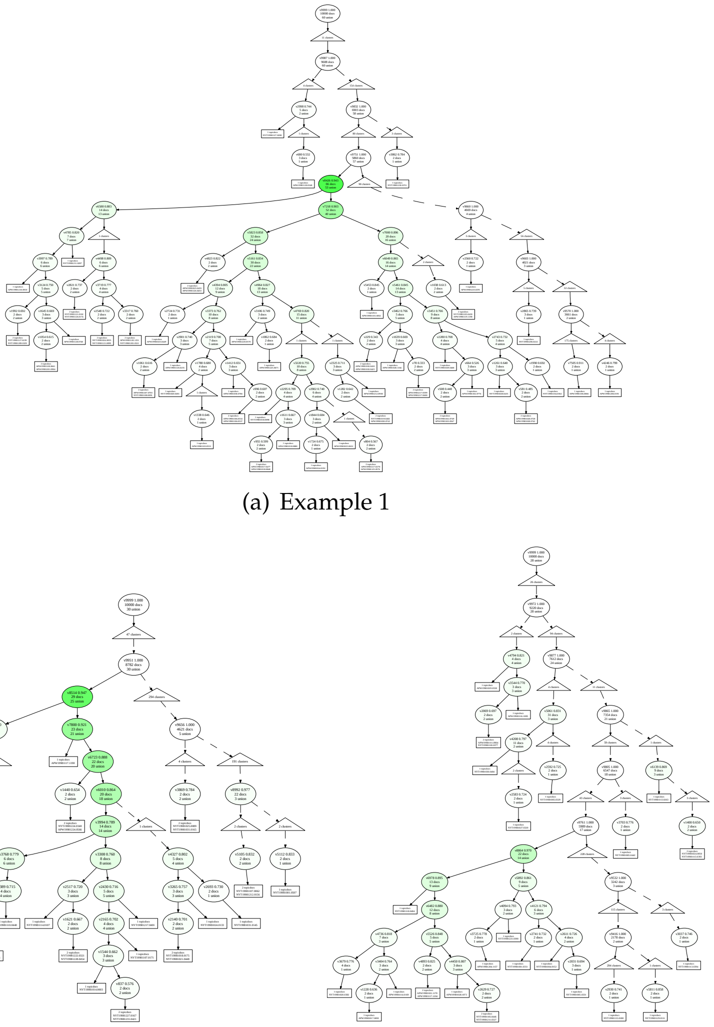


Figure 5.1: A nearly "perfect" cluster structure



(a) Example 1 (b) Example 2 (c) Example 3

Figure 5.2: Characteristic visualizations of single linkage

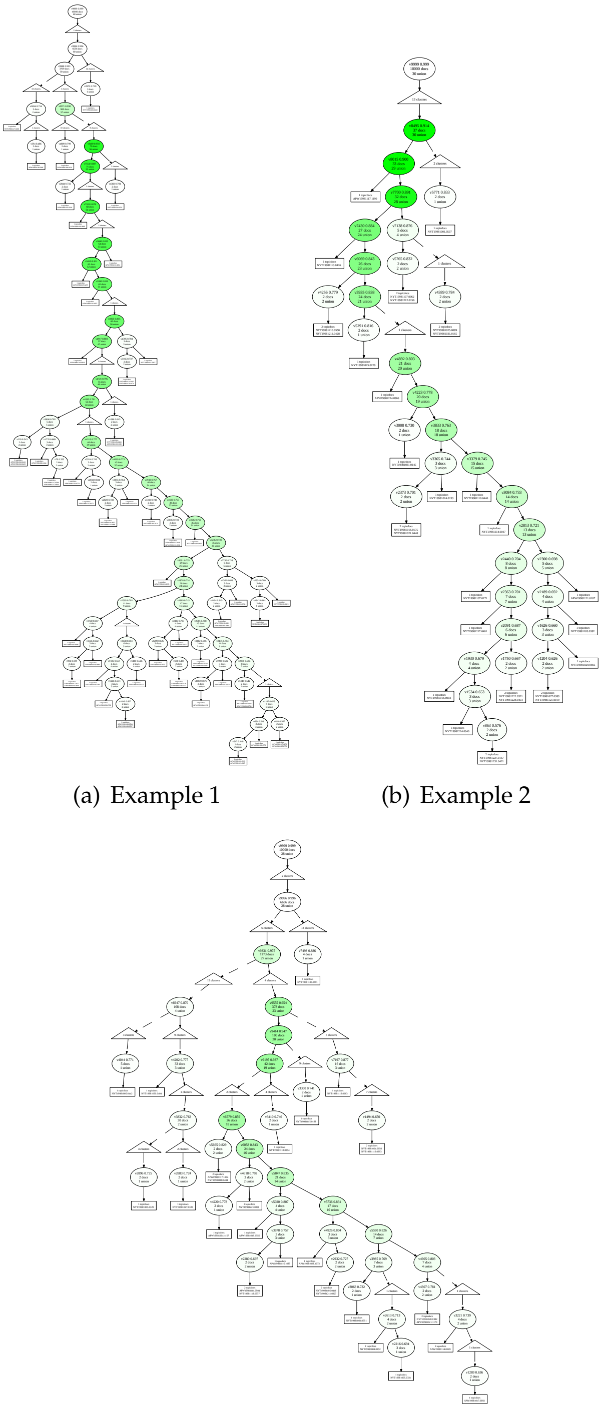


(a) Example 1

(b) Example 2

(c) Example 3

Figure 5.3: Characteristic visualizations of complete linkage



(a) Example 1

(b) Example 2

(c) Example 3

Figure 5.4: Characteristic visualizations of average linkage

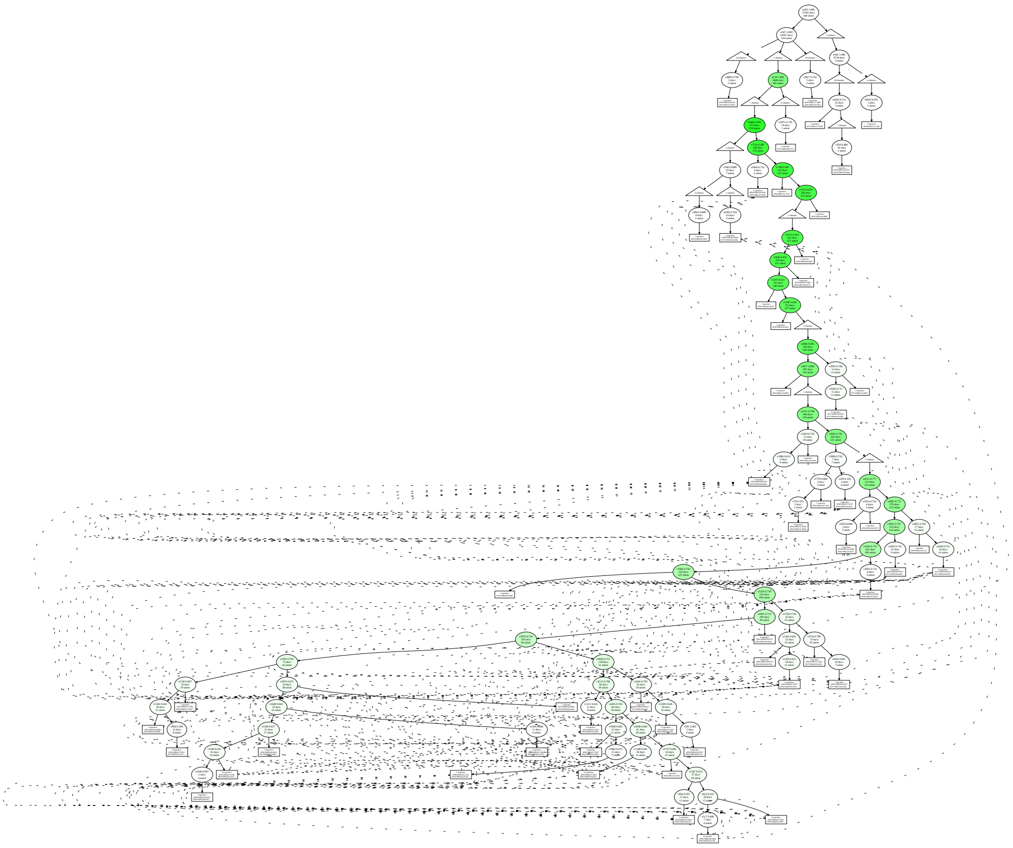


Figure 5.5: A fuzzy cluster structure after matching

Chapter 6

# Evaluation and discussion

In this chapter we will try to answer the research questions posed in the introduction chapter:

> How can an automated Hierarchical Topic Detection system be configured effectively to improve exploration and navigation of news archives?

With the following sub questions:

- How can a very large collection of news items be clustered effectively?
- How can the quality of an hierarchical cluster structure be measured?

In the following sections these questions will be discussed. Paragraph 6.3 concludes this work and deals with the main research question, followed by possible future research.

## 6.1 Scalable hierarchical topic clustering

The onset of this work was the participation in the Hierarchical Topic Detection task proposed by the TDT 2004. This trial evaluation task requires a large corpus of multilingual news items to be grouped in a structure of topic coherent clusters. The large size of the corpus and the limited amount of available processing time pose a challenge for the participants. The suggested directed acyclic graph structure offers rich possibilities for organizing news items: news items can be assigned to multiple clusters and clusters may be a subset of other clusters.

The used approach is a combination of classic hierarchical clustering methods with more recent language modelling techniques. The classic hierarchical clustering methods produce robust and deterministic cluster structures, but are in general not applicable for large document collections because of their complexity (see 2.2.3). Therefore the clustering methods are applied to a corpus sample of feasible size, resulting in a binary sample cluster structure. The remaining documents are added to multiple sample clusters, resulting in a fuzzy cluster structure. The cluster structure is optimized for the evaluation metric by applying a rebranching algorithm which rebuilds the upper part of the cluster structure.

During the evaluation the obtained cluster structure is compared to topics from a manually composed ground truth. The evaluation determines to which extent a cluster structure represents the ground truth and calculates a "lowest cost cluster" for each truth topic.

The evaluation shows the approach can, to a certain extent, group topically related news items. The quality of the approximation varies from topic to topic, some topics are outlined better by the best clusters found than others. The evaluation shows the recall of the best clusters is high (close to 97%), but the average precision is low (around 5%). Although the precision should be improved, it shows the method is capable of roughly outlining topically related groups of news items in a large collection.

Three different hierarchical agglomerative clustering methods were studied. The metric indicated average pairwise link clustering method outperformed single and complete link methods.

The sampling method seems promising in making hierarchical clustering scalable. The evaluation did not indicate the quality of the cluster structure deteriorated after adding the remaining documents to the sample cluster structure. Further research should point out if this indeed is true and to which extent this sampling method remains usable (see section 6.4).

The proposed rebranching method as intended decreases the cost of the best clusters judged by the evaluation metric, but if this approach also increases the quality of the clusters is questionable. The method only artificially rebuilds the top of the cluster structure, without using knowledge of the document collection (see 3.2.3).

Adding the remaining news items to more than one sample cluster increased the performance measured by the evaluation metric. It is doubted if this increased fuzziness also improves the quality of the cluster structure. By adding news items to multiple matching clusters, news items can be added to neighbouring clusters. If this situation occurs, the cluster structure may become confusing to understand: if neighbouring clusters overlap, should they exist anyway?

Although the proposed method leaves room for improvement (see further research in a later section), the evaluation raises several questions about the used metric. A better metric score does not always seem to correspond with an increase of the intuitive quality of the cluster structure. The metric sets a measurable goal for improvements in this area and a wrong evaluation metric can stimulate 'improvements' in the wrong direction. Therefore it was decided to study and improve the evaluation metric.

## 6.2 Evaluation of hierarchical cluster structures

The minimal cost metric used for the evaluation compares the clusters in a cluster structure with the topics from a ground truth and calculates a cost for each cluster-topic pair. This cost consists of a detection and a travel cost component. The detection cost measures how effective a particular cluster represents a topic (in terms of recall and precision). The travel cost expresses the "effort" to reach the cluster when traversing to this cluster from the root cluster. For each topic the cluster with the lowest cost is sought and this cluster is appointed best cluster for this topic.

A metric considering both effectiveness of the clusters and user effort to find these clusters seems useful: it keeps an eye on both the quality and the usefulness of the cluster structure. The modelling in the minimal cost metric seems to overshoot this goal however.

The best cluster has a combined score of detection and travel cost. The travel cost increases for clusters further from the root cluster and it limits the choice of best clusters. Clusters far down the cluster structure may not be considered as best cluster although its effectiveness for particular topics is high (see 4.2.3). This can lead to the false conclusion a cluster method is not capable of outlining these topics, when only the hierarchy does not meet the requirements of a low travel cost.

The travel cost to reach a cluster is calculated using the characteristics of the shortest search path from the root cluster to this cluster. A longer search path leads to a higher cost, indicating a lower quality. The travel cost implies a relationship between the characteristics of the shortest path to reach a cluster and the effort to find that cluster. It's questionable if this relationship is always correct: a shorter search path does not always imply the cluster is easier to locate. Removing a valuable layer of abstraction from a cluster structure, decreasing the travel cost for many clusters, will not likely decrease the effort to find clusters.

The travel cost should penalize 'powerset' structures, structures which contain (a large subset of) all the possible clusters. It in fact cannot discriminate between powerset and skewed cluster structures (see section 4.2.3).

Furthermore the travel cost prefers a particular branching factor of the clusters and favours balanced cluster structures (see section 4.2.3).

The travel cost imposes strong, but questionable preferences for particular cluster structures. In some cases these preferences are correct: a cluster structure which is very unbalanced might require more effort to reach clusters than a *comparable* structure which is more balanced. On the other hand it seems strange to impose a particular preferred branching factor if the documents don't give a reason to structure them in such a way.

A major cause in the problems of evaluating hierarchical cluster structures is the lack of a clearly defined goal. The HTD task does not clearly indicate what is expected from the cluster structure with "multiple levels of granularity". Furthermore the application of the hierarchical structure is unclear: is it used for for example browsing or maybe for some kind of cluster based retrieval? Not having a clear goal makes it difficult to determine a suitable metric or metric parameters. Despite an unclear task description a metric has been used which does clearly define desired properties.

Organizing an evaluation in an unclear setting should provide a playground for discussion about what can be achieved in HTD and how this can be measured. The current evaluation metric only presents a narrow view on the structures. The metric does define the desired cluster structure when these characteristics are still unclear. Furthermore the provided indicators do not cover the rich possibilities the new hierarchical structure offers.

Therefore it is suggested to use multiple separate indicators to describe the quality of a cluster structure. Separate indicators, if clearly defined, give a clear view of particular characteristics of a cluster structure. The value should always be assessed in combination with the values of other indicators. New indicators are needed to assess the possibilities the cluster structure has to offer. These should primarily aim at pointing out the complexity of the cluster structure: the 'fuzziness' of the news items and the connections between the clusters in the structure. A visualization of the evaluation of a particular topic can give valuable information about how a cluster structure outlines a topic.

Rather than evaluating a hierarchical cluster structure using a ground truth, a more theoretical methodology should be sought to assess the originating cluster method. By using a hierarchical agglomerative clustering method it was already clear a binary cluster structure would be the result. It's already questionable if a binary cluster tree is useful for browsing if the collection (and the resulting structure) is large. Assigning the remaining news items to multiple clusters indeed allows news items to cover multiple topics, as a news item can discuss two particular topics, but if done too excessively blurs the user's comprehension of the cluster structure. Although the rebranching operation partially brings down the top of the binary tree to a shallower tree with a higher branching factor, it does not introduce any useful levels of abstraction. These and more intrinsic properties of HTD methods might be collected in a framework to compare methods.

## 6.3 Conclusion

Fully automatic hierarchical topic detection operates between high expectations of flexible, intuitive taxonomies on one hand and limited computable models on the other.

The obtained cluster structures do not bring us the clear levels of abstraction we hoped for, but its tendency to group topically related news items can be made useful for exploration of a large unknown set of news items. For such a purpose the lack of precision is acceptable. The hierarchical structure as it is does not seem directly useful for browsing: especially near the root cluster, the clusters serve as "glue": they are simply used to collect the loose clusters. A method should be developed to make the top of the cluster structure more useful and to combine superfluous clusters further down the cluster structure (see future work). At this moment the clusters are unlabelled bags of documents and child clusters. A labelling algorithm should be applied to disclose the content of a cluster in a single glance.

The contributions of this work are a twofold.

First of all a simple scalable hierarchical clustering method has been proposed and used to participate in the TDT. With some adjustments the resulting cluster structure can improve exploration of a large unlabelled collection of news items.

The second contribution is a number of insights in the evaluation of hierarchical topic structures using a flat ground truth. A loose evaluation using multiple separated indicators is proposed to stimulate a discussion about the desired properties of a hierarchical topic structure.

## 6.4 Future work

During the previous paragraphs multiple suggestions were made for future work. This future work is divided in two sections: the first suggests further research on scalable HTD and to improve its usefulness; the second suggests research on the evaluation of HTD.

### 6.4.1 Scalable HTD

The proposed cluster method does not yet produce directly usable cluster structures. Future work should aim at modifying the cluster structures to make them directly usable for browsing. This could be done by for example subsuming clusters which do not make a clear distinction between groups of news items. The matching process might provide valuable information about documents belonging to one single cluster: if documents are contained by neighbouring clusters, these clusters might be merged.

Another approach can be merging the cluster structure with an existing news taxonomy. This improves the understandability of the upper part of the cluster structure. Appendix E shows some results of merging the obtained cluster structure in the Reuters news categorization.

Future work can include a study to improve the effectiveness of the clusters. This can include a study of the influence of the document representation and the influence of the used similarity measure. This work showed that average pairwise linkage outperformed complete and single linkage. Interesting would be to find out if and how the average pairwise link might be improved.

The sample based approach seems a promising method for clustering large collections. Further research might point out how the size and choice of this sample influences the quality of the overall cluster structure.

During manual evaluation of the obtained cluster structures became clear how important the labelling and visualization of the structure is. Further research could involve development of a graphical user interface for the cluster structures and labelling algorithm. The labelling process could be integrated with the clustering process to assure the labels remain informative and might even serve as feedback to find out if a clustering step is useful.

### 6.4.2 Evaluation of HTD systems

Modelling the user effort of finding a cluster to evaluate the user friendliness of a cluster structure would give valuable feedback on a cluster method. A user experiment could be set up to find out what really influences the usability of a cluster structure. This might be translated to particular values of evaluation indicators or lead to development of new indicators.

The development of a more theoretical evaluation methodology for hierarchical topic detection methods would be useful to step back from the experimental methodologies. This could include a framework of important properties of HTD methods.

This study has shown the importance of understanding the created cluster structures. This comprehension is not only of importance for evaluating a cluster structure but maybe even more for using it. By further improving this comprehension in both construction and evaluation of cluster structures, hierarchical topic detection can aid more in exploring and navigation of news archives.

# Bibliography

[AFB03] James Allan, Ao Feng, and Alvaro Bolivar. Flexible intrinsic evaluation of hierarchical clustering for TDT. In *Proceedings of the twelfth international conference on Information and knowledge management*, pages 263–270. ACM Press, 2003.

[BYRN99] Ricardo Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley Longman Publishing, 1999.

[CKPT92] Douglass R. Cutting, David R. Karger, Jan O. Pedersen, and John W. Tukey. Scatter/gather: a cluster-based approach to browsing large document collections. In *Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 318–329. ACM Press, 1992.

[DL01] Manoranjan Dash and Huan Liu. Efficient hierarchical clustering algorithms using partially overlapping partitions. *Lecture Notes in Computer Science*, 2035, 2001.

[DMO05] DMOZ. The open directory project. `http://www.dmoz.org/`, 2005.

[EHW86] A. El-Hamdouchi and P. Willett. Hierarchic document classification using ward's clustering method. In *SIGIR '86: Proceedings of the 9th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 149–156. ACM Press, 1986.

[EHW87] Abdelmoula El-Hamdouchi and Peter Willett. Techniques for the measurement of clustering tendency in document retrieval systems. *Journal of Information Science*, 13(6):361–365, 1987.

[FBY92] William B. Frakes and Ricardo Baeza-Yates. *Information retrieval: data structures and algorithms*. Prentice-Hall, 1992.

[FD02] Jonathan G. Fiscus and George R. Doddington. Topic detection and tracking evaluation overview. pages 17–31, 2002.

[HP96] Marti A. Hearst and Jan O. Pedersen. Reexamining the cluster hypothesis: scatter/gather on retrieval results. In *SIGIR '96: Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 76–84. ACM Press, 1996.

[JMF99] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. *ACM Computing Surveys*, 31(3):264–323, 1999.

[Kra04] Wessel Kraaij. *Variations on language modeling for information retrieval*. PhD thesis, University of Twente, May 2004.

[Lin04] Linguistic Data Consortium. TDT 5: 2004: Project resources for 2004 evaluation. `http://www.ldc.upenn.edu/Projects/TDT2004`, 2004.

[MW05] Merriam-Webster. Merriam-Webster's collegiate dictionary. `http://www.webster.com/`, 2005.

[NIS04] NIST. The 2004 Topic Detection and Tracking (TDT2004) task definition and evaluation plan. `http://www.nist.gov/speech/tests/tdt/index.htm`, 2004.

[PL02] Patrick Pantel and Dekang Lin. Document clustering with committees. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 199–206. ACM Press, 2002.

[PSHD96] Peter Pirolli, Patricia Schank, Marti Hearst, and Christine Diehl. Scatter/gather browsing communicates the topic structure of a very large text collection. In *CHI '96: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 213–220. ACM Press, 1996.

[Sal68] Gerard Salton. *Automatic Information Organisation and Retrieval*. McGraw-Hill, New York, 1968.

[Sal91] Gerard Salton. Developments in automatic text retrieval. *Science*, 253(5023):974–980, 1991.

[SBCQ98] Alan F. Smeaton, Mark Burnett, Francis Crimmins, and Gerard Quinn. An architecture for efficient document clustering and retrieval on a dynamic collection of newspaper texts. In *Proceedings of the 20th BCS-IRSG Annual Colloquium*, 1998.

[SC99] Mark Sanderson and Bruce Croft. Deriving concept hierarchies from text. In *SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 206–213. ACM Press, 1999.

[SK01] Martijn Spitters and Wessel Kraaij. TNO at TDT2001: Language model-based topic detection, 2001.

[SK02] Martijn Spitters and Wessel Kraaij. Unsupervised event clustering in multilingual news streams. *Proceedings of the LREC2002 Workshop on Event Modeling for Multilingual Document Linking*, pages 42–46, 2002.

[SKK00] M. Steinbach, G. Karypis, and V. Kumar. A comparison of document clustering techniques, 2000.

[Voo85] Ellen Marie Voorhees. The cluster hypothesis revisited. In *Proceedings of the 8th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 188–196. ACM Press, 1985.

[vR79] C. J. van Rijsbergen. *Information Retrieval*. Butterworths, London, second edition, 1979.

[WF00] Wai-chiu Wong and Ada Wai-chee Fu. Incremental document clustering for web page classification. In *Proceedings of the International Conference on Information Society in the 21st Century IS2000*, 2000.

[Wil88] Peter Willett. Recent trends in hierarchic document clustering: a critical review. *Information Processing Management*, 24(5):577–597, 1988.

[ZRL96] Tian Zhang, Raghu Ramakrishnan, and Miron Livny. BIRCH: an efficient data clustering method for very large databases. In *SIGMOD '96: Proceedings of the 1996 ACM SIGMOD international conference on Management of data*, pages 103–114. ACM Press, 1996.

Appendix A

# Cluster structure DMOZ

As a sidestep we will take a look at DMOZ (an acronym for Directory Mozilla), also known as the Open Directory Project. “The Open Directory Project is the largest, most comprehensive human-edited directory of the Web. It is constructed and maintained by a vast, global community of volunteer editors” [DMO05]. The DMOZ is a hierarchical categorization of over 4 million websites. At the top level abstract categories are defined as “Sports”, “Arts” and “News” but further down one can find “Criticism” (as a subcategory of Computers: Programming: Methodologies: Object-Oriented).

Figure A.1 shows the cluster structure of DMOZ concerning the number of clusters having child clusters and the average number of child clusters (the branching factor) per cluster (if such a cluster has children). At the top level the structure has a branching factor of 17; the top level cluster has 17 children. These 17 cluster all have child clusters, with an average of 37.5 child clusters. The point we would like to make is that the DMOZ cluster structure is not balanced at all; some clusters don’t have child clusters below depth two, where some clusters have child clusters at depth thirteen! We argue that, as the articles from a news archive might be as diverse as websites from the DMOZ directory, a cluster structure for a news archive does not need to be balanced nor do the clusters need to kept to a fixed branching factor.

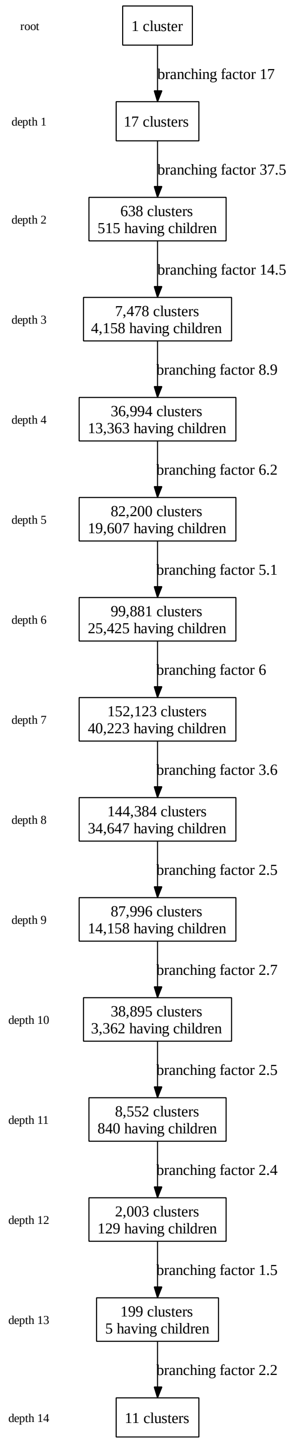


Figure A.1: Cluster structure of DMOZ

Appendix B

# HAC methods and document discrimination

In this appendix a small example will show how the distances between documents change by clustering using single, complete and average pairwise link.

Assume five documents of equal length, $D_1$ to $D_5$. They all contain three terms, a subset of $\{a \ldots h, p \ldots r\}$:

- $D_1 = \{c, d, e\}$
- $D_2 = \{d, e, f\}$
- $D_3 = \{a, b, c\}$
- $D_4 = \{f, g, h\}$
- $D_5 = \{p, q, r\}$

Or in a term vector ('-' marks no occurrences):

| | a | b | c | d | e | f | g | h | p | q | r |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $D_1$ | - | - | 1 | 1 | 1 | - | - | - | - | - | - |
| $D_2$ | - | - | - | 1 | 1 | 1 | - | - | - | - | - |
| $D_3$ | 1 | 1 | 1 | - | - | - | - | - | - | - | - |
| $D_4$ | - | - | - | - | - | 1 | 1 | 1 | - | - | - |
| $D_5$ | - | - | - | - | - | - | - | - | 1 | 1 | 1 |

As a similarity function we will simply use the number of overlapping terms. The following distance matrix can be created:

| | $D_1$ | $D_2$ | $D_3$ | $D_4$ |
|---|---|---|---|---|
| $D_2$ | 2 | - | - | - |
| $D_3$ | 1 | 0 | - | - |
| $D_4$ | 0 | 1 | 0 | - |
| $D_5$ | 0 | 0 | 0 | 0 |

All three methods will choose $D_1$ and $D_2$ as first pair of documents to cluster, as the overlap is the largest. This call newly created cluster is $D_x$. The distance between the remaining (document)clusters and $D_x$ has to be calculated and this calculation depends on which HAC method is used. The following table shows the new distances using single, complete and average pairwise link.

| | Single | Complete | Average |
|---|---|---|---|
| | $D_x$ | $D_x$ | $D_x$ |
| $D_3$ | 1 | 0 | 1 |
| $D_4$ | 1 | 0 | 1 |
| $D_5$ | 0 | 0 | 0 |

Notice that the complete linkage method throws away *all* distance knowledge from the children of $D_x$: $D_1$ and $D_2$ don't have overlapping terms overlapping with all the remaining documents.

Likewise can be shown how single linkage 'chains' two at first sight unrelated documents.

Average pairwise link doesn't suffer from these extremes.

Appendix C

# Developed software

Software has been written in Java for this project. In this appendix the most important components are briefly mentioned; the Javadoc found in the source code gives further details.

## Package nl.tno.htd.structures

Contains the main data structures: Clustering, Cluster, Document). Contains implementations for a cluster structure stored in an xml-file and a database.

## Package nl.tno.htd.clusterbrowser

Contains classes for the ClusterBrowser user interface. The main program is started with:

```
java nl.tno.htd.clusterbrowser.ClusterBrowser
```

### Subpackage clusteringtree

Contains classes for the clustering tree which displays a Clustering in a tree structure.

### Subpackage lucene

Contains classes for a searchframe using Lucene.

## Package nl.tno.htd.evaluation

Contains classes for evaluating a cluster structure using a ground truth and which creates *Dot* files for visualizing the evaluation of a topic. The evaluation can be run with:

```
java nl.tno.htd.evaluation.Evaluate
```

## Package nl.tno.htd.tools

Contains a number of commandline tools for modifying and exporting cluster structures.

- `ClusterTransformer` can rebranch a clustering or add matching documents.
- `DetectClusterPlagiaat` uses legacy software to indicate if a cluster contains near-duplicates.
- `DotExporter` exports a cluster structure for visualization using Dot.
- `GroundTruthAnalyzer` creates a LaTeX file with histograms of the time distributions of ground truth topics.
- `TreeMapExporter` exports a cluster structure to a file readable by TreeMap (see `www.cs.umd.edu/hcil/treemap/`).

## Package nl.tno.htd.lucene

Contains classes for indexing a database containing text assets using the Lucene open source search engine. The main program can be run with:

```
java nl.tno.htd.lucene.IndexDbAssets
```

## Package nl.tno.htd.reuters

Contains classes for indexing the Reuters text collection with Lucene and for "annotating" a cluster structure with the topics and countries used for Reuters. Main programs:

```
java nl.tno.htd.reuters.AnnotateClusteringDocuments
java nl.tno.htd.reuters.ReuterizeClustering
java nl.tno.htd.reuters.index.CreateReutersIndex
```

## Package nl.tno.htd.dmoz

Contains classes to visualize the dmoz cluster structure (see appendix A).

## Package nl.tno.htd.utils

Some useful (non-specific) data structures.

Appendix D

# Evaluation results

In this appendix the evaluation of various cluster structures of the TDT 3 corpus can be found.

The tables show for each topic the topicfuzziness, the number of topicrelevant clusters and the indicator values of the most effective topicrelevant cluster (i.e. having the lowest detection cost).

## D.1 TDT3 sample cluster structure single linkage

| *Topic* | | | | *Cluster* | | | *Effectiveness* | | | | | | *Hierarchy above* | | | *Hierarchy below* | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *Id* | *Size* | *Fuzz.* | *#Relevant* | *Id* | *Size* | *Diss.* | *#Union* | *#FA* | *#Miss* | *Recall* | *Precision* | *Det.cost* | *#Root paths* | *Avg enc. branches from root* | *Avg path length from root* | *#Topicdoc paths* | *Avg path length to topicdocs* |
| 30001 | 5 | 1 | 3104 | v2467 | 44 | 0.678 | 5 | 39 | 0 | 1 | 0.1136 | 0.0004 | 1 | 6168 | 3084 | 5 | 6.4 |
| 30002 | 33 | 1 | 3651 | v4806 | 30 | 0.753 | 10 | 20 | 23 | 0.303 | 0.3333 | 0.0141 | 1 | 6660 | 3330 | 10 | 5.7 |
| 30003 | 60 | 1 | 3625 | v6085 | 83 | 0.793 | 56 | 27 | 4 | 0.9333 | 0.6747 | 0.0016 | 1 | 5956 | 2978 | 56 | 32.68 |
| 30006 | 30 | 1 | 3020 | v5644 | 25 | 0.779 | 22 | 3 | 8 | 0.7333 | 0.88 | 0.0054 | 1 | 5928 | 2964 | 22 | 12.68 |
| 30014 | 1 | 1 | 1591 | d5750 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 3180 | 1590 | 1 | 0 |
| 30016 | 4 | 1 | 2210 | v5917 | 18 | 0.788 | 4 | 14 | 0 | 1 | 0.2222 | 0.0001 | 1 | 4398 | 2199 | 4 | 3.75 |
| 30017 | 9 | 1 | 3226 | v4046 | 46 | 0.73 | 5 | 41 | 4 | 0.5556 | 0.1087 | 0.0093 | 1 | 6388 | 3194 | 5 | 7.2 |
| 30018 | 1 | 1 | 2600 | d7084 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 5198 | 2599 | 1 | 0 |
| 30022 | 10 | 1 | 2896 | v3966 | 27 | 0.727 | 10 | 17 | 0 | 1 | 0.3704 | 0.0002 | 1 | 5738 | 2869 | 10 | 9.3 |
| 30030 | 3 | 1 | 3187 | v4305 | 4 | 0.737 | 3 | 1 | 0 | 1 | 0.75 | 0 | 1 | 6362 | 3181 | 3 | 2 |
| 30031 | 8 | 1 | 2954 | v5372 | 20 | 0.77 | 8 | 12 | 0 | 1 | 0.4 | 0.0001 | 1 | 5862 | 2931 | 8 | 5.75 |
| 30032 | 1 | 1 | 3095 | d8296 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 6188 | 3094 | 1 | 0 |
| 30033 | 41 | 1 | 3878 | v4931 | 32 | 0.757 | 20 | 12 | 21 | 0.4878 | 0.625 | 0.0104 | 1 | 7160 | 3580 | 20 | 8.2 |
| 30034 | 5 | 1 | 2653 | v5808 | 90 | 0.784 | 5 | 85 | 0 | 1 | 0.0556 | 0.0008 | 1 | 5232 | 2616 | 5 | 11.4 |
| 30038 | 9 | 1 | 2628 | v4037 | 10 | 0.729 | 9 | 1 | 0 | 1 | 0.9 | 0 | 1 | 5220 | 2610 | 9 | 5 |
| 30040 | 17 | 1 | 3918 | v4139 | 330 | 0.732 | 13 | 317 | 4 | 0.7647 | 0.0394 | 0.0078 | 1 | 7522 | 3761 | 13 | 57.85 |
| 30042 | 12 | 1 | 3574 | v2822 | 24 | 0.691 | 10 | 14 | 2 | 0.8333 | 0.4167 | 0.0035 | 1 | 7082 | 3541 | 10 | 9 |
| 30043 | 3 | 1 | 2999 | v3876 | 11 | 0.724 | 2 | 9 | 1 | 0.6667 | 0.1818 | 0.0068 | 1 | 5978 | 2989 | 2 | 4 |
| 30046 | 17 | 1 | 3849 | v4209 | 161 | 0.735 | 8 | 153 | 9 | 0.4706 | 0.0497 | 0.0121 | 1 | 7510 | 3755 | 8 | 38.38 |
| 30047 | 10 | 1 | 2956 | v5372 | 20 | 0.77 | 10 | 10 | 0 | 1 | 0.5 | 0.0001 | 1 | 5862 | 2931 | 10 | 5.8 |
| 30048 | 28 | 1 | 3875 | v4641 | 18 | 0.748 | 11 | 7 | 17 | 0.3929 | 0.6111 | 0.0122 | 1 | 7138 | 3569 | 11 | 5.82 |
| 30049 | 6 | 1 | 3221 | v3652 | 29 | 0.717 | 6 | 23 | 0 | 1 | 0.2069 | 0.0002 | 1 | 6394 | 3197 | 6 | 7.67 |
| 30050 | 105 | 1 | 4121 | v4209 | 161 | 0.735 | 30 | 131 | 75 | 0.2857 | 0.1863 | 0.0156 | 1 | 7510 | 3755 | 30 | 32.47 |
| 30051 | 8 | 1 | 3862 | v4560 | 440 | 0.746 | 7 | 433 | 1 | 0.875 | 0.0159 | 0.0067 | 1 | 7384 | 3692 | 7 | 57.29 |
| 30053 | 34 | 1 | 3950 | v3528 | 212 | 0.713 | 25 | 187 | 9 | 0.7353 | 0.1179 | 0.0071 | 1 | 7582 | 3791 | 25 | 69.84 |
| 30055 | 11 | 1 | 3846 | v3169 | 12 | 0.702 | 6 | 6 | 5 | 0.5455 | 0.5 | 0.0091 | 1 | 7450 | 3725 | 6 | 3.67 |
| 30057 | 1 | 1 | 1034 | d9727 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 2066 | 1033 | 1 | 0 |
| 31001 | 28 | 1 | 3934 | v7802 | 23 | 0.839 | 9 | 14 | 19 | 0.3214 | 0.3913 | 0.0137 | 1 | 3876 | 1938 | 9 | 5.89 |
| 31002 | 25 | 1 | 3861 | v2329 | 6 | 0.672 | 4 | 2 | 21 | 0.16 | 0.6667 | 0.0168 | 1 | 5948 | 2974 | 4 | 3 |
| 31004 | 12 | 1 | 3469 | v4064 | 8 | 0.73 | 3 | 5 | 9 | 0.25 | 0.375 | 0.015 | 1 | 4860 | 2430 | 3 | 3 |
| 31006 | 20 | 1 | 3871 | v4960 | 2 | 0.758 | 2 | 0 | 18 | 0.1 | 1 | 0.018 | 1 | 4292 | 2146 | 2 | 1 |
| 31009 | 41 | 1 | 3988 | v4259 | 382 | 0.736 | 26 | 356 | 15 | 0.6341 | 0.0681 | 0.0108 | 1 | 7506 | 3753 | 26 | 21.04 |
| 31012 | 15 | 1 | 3123 | v347 | 2 | 0.379 | 2 | 0 | 13 | 0.1333 | 1 | 0.0173 | 1 | 4376 | 2188 | 2 | 1 |
| 31013 | 32 | 1 | 3786 | v5467 | 7 | 0.774 | 7 | 0 | 25 | 0.2188 | 1 | 0.0156 | 1 | 6032 | 3016 | 7 | 3.86 |
| 31014 | 25 | 1 | 3725 | v4242 | 30 | 0.735 | 9 | 21 | 16 | 0.36 | 0.3 | 0.013 | 1 | 5734 | 2867 | 9 | 11.11 |
| 31016 | 23 | 1 | 3252 | v7838 | 51 | 0.84 | 20 | 31 | 3 | 0.8696 | 0.3922 | 0.0029 | 1 | 3630 | 1815 | 20 | 21.35 |
| 31017 | 14 | 1 | 3630 | v1932 | 3 | 0.656 | 2 | 1 | 12 | 0.1429 | 0.6667 | 0.0172 | 1 | 6742 | 3371 | 2 | 1.5 |
| 31019 | 27 | 1 | 3905 | v6806 | 12 | 0.814 | 6 | 6 | 21 | 0.2222 | 0.5 | 0.0156 | 1 | 5234 | 2617 | 6 | 6.33 |
| 31020 | 10 | 1 | 3803 | v746 | 2 | 0.538 | 2 | 0 | 8 | 0.2 | 1 | 0.016 | 1 | 5736 | 2868 | 2 | 1 |
| 31022 | 21 | 1 | 2587 | v6924 | 22 | 0.816 | 6 | 16 | 15 | 0.2857 | 0.2727 | 0.0144 | 1 | 4356 | 2178 | 6 | 8 |
| 31024 | 27 | 1 | 3967 | v2973 | 26 | 0.696 | 5 | 21 | 22 | 0.1852 | 0.1923 | 0.0165 | 1 | 7538 | 3769 | 5 | 10.8 |
| 31026 | 117 | 1 | 3560 | v6038 | 153 | 0.791 | 43 | 110 | 74 | 0.3675 | 0.281 | 0.0137 | 1 | 6146 | 3073 | 43 | 33.81 |
| 31029 | 18 | 1 | 3819 | v1099 | 6 | 0.6 | 4 | 2 | 14 | 0.2222 | 0.6667 | 0.0156 | 1 | 7402 | 3701 | 4 | 3.25 |
| 31031 | 93 | 1 | 4228 | v4209 | 161 | 0.735 | 35 | 126 | 58 | 0.3763 | 0.2174 | 0.0137 | 1 | 7510 | 3755 | 35 | 50.89 |
| 31033 | 388 | 1 | 5072 | v4701 | 48 | 0.75 | 27 | 21 | 361 | 0.0696 | 0.5625 | 0.0188 | 1 | 6682 | 3341 | 27 | 19 |
| 31034 | 16 | 1 | 3857 | v4972 | 12 | 0.758 | 3 | 9 | 13 | 0.1875 | 0.25 | 0.0163 | 1 | 5778 | 2889 | 3 | 7.33 |
| 31035 | 37 | 1 | 3997 | v4913 | 24 | 0.757 | 7 | 17 | 30 | 0.1892 | 0.2917 | 0.0164 | 1 | 7128 | 3564 | 7 | 9.29 |
| 31036 | 40 | 1 | 3212 | v6038 | 153 | 0.791 | 21 | 132 | 19 | 0.525 | 0.1373 | 0.0108 | 1 | 6146 | 3073 | 21 | 37.19 |
| 31040 | 12 | 1 | 3505 | v4116 | 52 | 0.732 | 7 | 45 | 5 | 0.5833 | 0.1346 | 0.0088 | 1 | 6878 | 3439 | 7 | 18.86 |
| 31042 | 20 | 1 | 3859 | v3970 | 9 | 0.727 | 3 | 6 | 17 | 0.15 | 0.3333 | 0.0171 | 1 | 5228 | 2614 | 3 | 3 |
| 31043 | 31 | 1 | 3860 | v4066 | 321 | 0.73 | 21 | 300 | 10 | 0.6774 | 0.0654 | 0.0094 | 1 | 7528 | 3764 | 21 | 14.86 |
| 31044 | 41 | 1 | 3983 | v5258 | 16 | 0.767 | 6 | 10 | 35 | 0.1463 | 0.375 | 0.0172 | 1 | 6712 | 3356 | 6 | 6.5 |
| 31045 | 16 | 1 | 3861 | v4996 | 8 | 0.759 | 3 | 5 | 13 | 0.1875 | 0.375 | 0.0163 | 1 | 3884 | 1942 | 3 | 3.33 |
| 31048 | 24 | 1 | 3277 | v5230 | 69 | 0.766 | 15 | 54 | 9 | 0.625 | 0.2174 | 0.008 | 1 | 6370 | 3185 | 15 | 14.87 |
| 31051 | 23 | 1 | 3256 | v6039 | 12 | 0.792 | 11 | 1 | 12 | 0.4783 | 0.9167 | 0.0104 | 1 | 5216 | 2608 | 11 | 6 |
| 31057 | 24 | 1 | 3252 | v5704 | 79 | 0.781 | 7 | 72 | 17 | 0.2917 | 0.0886 | 0.0149 | 1 | 6358 | 3179 | 7 | 10 |
| 31059 | 35 | 1 | 4126 | v4242 | 30 | 0.735 | 5 | 25 | 30 | 0.1429 | 0.1667 | 0.0174 | 1 | 5734 | 2867 | 5 | 7.2 |
| Averages | 30.30 | 1 | 3458.74 | | 62.81 | 0.6841 | 10.70 | 52.11 | 19.60 | 0.5419 | 0.4415 | 0.0097 | 1 | 5994.32 | 2997.16 | 10.70 | 13.09 |

## D.2 TDT3 sample cluster structure complete linkage

| *Topic* | | | | *Cluster* | | | *Effectiveness* | | | | | | *Hierarchy above* | | | *Hierarchy below* | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *Id* | *Size* | *Fuzz.* | *#Relevant* | *Id* | *Size* | *Diss.* | *#Union* | *#FA* | *#Miss* | *Recall* | *Precision* | *Det.cost* | *#Root paths* | *Avg enc. branches from root* | *Avg path length from root* | *#Topicdoc paths* | *Avg path length to topicdocs* |
| 30001 | 5 | 1 | 466 | v2520 | 11 | 0.72 | 3 | 8 | 2 | 0.6 | 0.2727 | 0.0081 | 1 | 814 | 407 | 3 | 3.67 |
| 30002 | 33 | 1 | 722 | v7811 | 18 | 0.921 | 13 | 5 | 20 | 0.3939 | 0.7222 | 0.0122 | 1 | 576 | 288 | 13 | 4.92 |
| 30003 | 60 | 1 | 757 | v8426 | 66 | 0.943 | 53 | 13 | 7 | 0.8833 | 0.803 | 0.0025 | 1 | 498 | 249 | 53 | 7.51 |
| 30006 | 30 | 1 | 605 | v8514 | 29 | 0.947 | 25 | 4 | 5 | 0.8333 | 0.8621 | 0.0034 | 1 | 98 | 49 | 25 | 6.84 |
| 30014 | 1 | 1 | 465 | d5750 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 928 | 464 | 1 | 0 |
| 30016 | 4 | 1 | 860 | v8305 | 24 | 0.938 | 4 | 20 | 0 | 1 | 0.1667 | 0.0002 | 1 | 1690 | 845 | 4 | 5.25 |
| 30017 | 9 | 1 | 526 | v8738 | 30 | 0.96 | 4 | 26 | 5 | 0.4444 | 0.1333 | 0.0114 | 1 | 494 | 247 | 4 | 7 |
| 30018 | 1 | 1 | 552 | d7084 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1102 | 551 | 1 | 0 |
| 30022 | 10 | 1 | 249 | v7636 | 36 | 0.916 | 10 | 26 | 0 | 1 | 0.2778 | 0.0003 | 1 | 440 | 220 | 10 | 5.9 |
| 30030 | 3 | 1 | 282 | v7775 | 6 | 0.92 | 3 | 3 | 0 | 1 | 0.5 | 0 | 1 | 548 | 274 | 3 | 3 |
| 30031 | 8 | 1 | 77 | v7778 | 17 | 0.92 | 8 | 9 | 0 | 1 | 0.4706 | 0.0001 | 1 | 110 | 55 | 8 | 4.25 |
| 30032 | 1 | 1 | 61 | d8296 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 120 | 60 | 1 | 0 |
| 30033 | 41 | 1 | 949 | v7089 | 14 | 0.899 | 11 | 3 | 30 | 0.2683 | 0.7857 | 0.0147 | 1 | 318 | 159 | 11 | 5 |
| 30034 | 5 | 1 | 588 | v4226 | 18 | 0.798 | 4 | 14 | 1 | 0.8 | 0.2222 | 0.0041 | 1 | 1136 | 568 | 4 | 3.75 |
| 30038 | 9 | 1 | 332 | v5371 | 11 | 0.842 | 9 | 2 | 0 | 1 | 0.8182 | 0 | 1 | 626 | 313 | 9 | 4.11 |
| 30040 | 17 | 1 | 655 | v7504 | 42 | 0.911 | 14 | 28 | 3 | 0.8235 | 0.3333 | 0.0038 | 1 | 1190 | 595 | 14 | 6.36 |
| 30042 | 12 | 1 | 682 | v6508 | 24 | 0.881 | 10 | 14 | 2 | 0.8333 | 0.4167 | 0.0035 | 1 | 130 | 65 | 10 | 4.8 |
| 30043 | 3 | 1 | 440 | v2770 | 2 | 0.734 | 2 | 0 | 1 | 0.6667 | 1 | 0.0067 | 1 | 854 | 427 | 2 | 1 |
| 30046 | 17 | 1 | 859 | v5963 | 9 | 0.863 | 8 | 1 | 9 | 0.4706 | 0.8889 | 0.0106 | 1 | 260 | 130 | 8 | 3.5 |
| 30047 | 10 | 1 | 79 | v7778 | 17 | 0.92 | 10 | 7 | 0 | 1 | 0.5882 | 0.0001 | 1 | 110 | 55 | 10 | 4.5 |
| 30048 | 28 | 1 | 960 | v8864 | 22 | 0.97 | 14 | 8 | 14 | 0.5 | 0.6364 | 0.0101 | 1 | 478 | 239 | 14 | 4.64 |
| 30049 | 6 | 1 | 291 | v4080 | 11 | 0.792 | 4 | 7 | 2 | 0.6667 | 0.3636 | 0.0067 | 1 | 530 | 265 | 4 | 4 |
| 30050 | 105 | 1 | 1157 | v9025 | 47 | 0.979 | 21 | 26 | 84 | 0.2 | 0.4468 | 0.0163 | 1 | 38 | 19 | 21 | 7.62 |
| 30051 | 8 | 1 | 773 | v8288 | 9 | 0.937 | 4 | 5 | 4 | 0.5 | 0.4444 | 0.01 | 1 | 1476 | 738 | 4 | 3.5 |
| 30053 | 34 | 1 | 706 | v8679 | 45 | 0.956 | 20 | 25 | 14 | 0.5882 | 0.4444 | 0.0085 | 1 | 1188 | 594 | 20 | 7.95 |
| 30055 | 11 | 1 | 477 | v8927 | 17 | 0.973 | 8 | 9 | 3 | 0.7273 | 0.4706 | 0.0055 | 1 | 866 | 433 | 8 | 5.12 |
| 30057 | 1 | 1 | 653 | d9727 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1304 | 652 | 1 | 0 |
| 31001 | 28 | 1 | 904 | v8146 | 14 | 0.932 | 9 | 5 | 19 | 0.3214 | 0.6429 | 0.0136 | 1 | 1042 | 521 | 9 | 4.22 |
| 31002 | 25 | 1 | 936 | v8079 | 15 | 0.93 | 9 | 6 | 16 | 0.36 | 0.6 | 0.0129 | 1 | 1154 | 577 | 9 | 4.22 |
| 31004 | 12 | 1 | 448 | v4628 | 8 | 0.814 | 3 | 5 | 9 | 0.25 | 0.375 | 0.015 | 1 | 80 | 40 | 3 | 3.33 |
| 31006 | 20 | 1 | 733 | v8680 | 6 | 0.956 | 3 | 3 | 17 | 0.15 | 0.5 | 0.017 | 1 | 212 | 106 | 3 | 2.67 |
| 31009 | 41 | 1 | 965 | v2666 | 12 | 0.728 | 10 | 2 | 31 | 0.2439 | 0.8333 | 0.0151 | 1 | 180 | 90 | 10 | 4.5 |
| 31012 | 15 | 1 | 644 | v8454 | 24 | 0.944 | 4 | 20 | 11 | 0.2667 | 0.1667 | 0.0149 | 1 | 1122 | 561 | 4 | 4.5 |
| 31013 | 32 | 1 | 788 | v8974 | 16 | 0.976 | 12 | 4 | 20 | 0.375 | 0.75 | 0.0125 | 1 | 610 | 305 | 12 | 4.33 |
| 31014 | 25 | 1 | 737 | v7636 | 36 | 0.916 | 9 | 27 | 16 | 0.36 | 0.25 | 0.0131 | 1 | 440 | 220 | 9 | 6.11 |
| 31016 | 23 | 1 | 747 | v9066 | 47 | 0.982 | 17 | 30 | 6 | 0.7391 | 0.3617 | 0.0055 | 1 | 1338 | 669 | 17 | 7.06 |
| 31017 | 14 | 1 | 914 | v8742 | 19 | 0.96 | 3 | 16 | 11 | 0.2143 | 0.1579 | 0.0159 | 1 | 1094 | 547 | 3 | 3.67 |
| 31019 | 27 | 1 | 974 | v7730 | 10 | 0.919 | 6 | 4 | 21 | 0.2222 | 0.6 | 0.0156 | 1 | 760 | 380 | 6 | 3.67 |
| 31020 | 10 | 1 | 783 | v700 | 2 | 0.538 | 2 | 0 | 8 | 0.2 | 1 | 0.016 | 1 | 102 | 51 | 2 | 1 |
| 31022 | 21 | 1 | 955 | v8949 | 8 | 0.975 | 3 | 5 | 18 | 0.1429 | 0.375 | 0.0172 | 1 | 1354 | 677 | 3 | 3.33 |
| 31024 | 27 | 1 | 955 | v7937 | 44 | 0.926 | 5 | 39 | 22 | 0.1852 | 0.1136 | 0.0167 | 1 | 118 | 59 | 5 | 7 |
| 31026 | 117 | 1 | 1182 | v9039 | 82 | 0.98 | 20 | 62 | 97 | 0.1709 | 0.2439 | 0.0172 | 1 | 22 | 11 | 20 | 7.1 |
| 31029 | 18 | 1 | 894 | v7991 | 17 | 0.928 | 9 | 8 | 9 | 0.5 | 0.5294 | 0.0101 | 1 | 280 | 140 | 9 | 4.56 |
| 31031 | 93 | 1 | 1235 | v5381 | 26 | 0.842 | 16 | 10 | 77 | 0.172 | 0.6154 | 0.0167 | 1 | 18 | 9 | 16 | 6.06 |
| 31033 | 388 | 1 | 2321 | v8096 | 28 | 0.931 | 16 | 12 | 372 | 0.0412 | 0.5714 | 0.0193 | 1 | 252 | 126 | 16 | 6.06 |
| 31034 | 16 | 1 | 912 | v4990 | 10 | 0.828 | 3 | 7 | 13 | 0.1875 | 0.3 | 0.0163 | 1 | 484 | 242 | 3 | 5 |
| 31035 | 37 | 1 | 1033 | v7638 | 13 | 0.916 | 6 | 7 | 31 | 0.1622 | 0.4615 | 0.0168 | 1 | 270 | 135 | 6 | 5.17 |
| 31036 | 40 | 1 | 987 | v7675 | 19 | 0.917 | 9 | 10 | 31 | 0.225 | 0.4737 | 0.0156 | 1 | 220 | 110 | 9 | 4.89 |
| 31040 | 12 | 1 | 723 | v7610 | 44 | 0.915 | 6 | 38 | 6 | 0.5 | 0.1364 | 0.0104 | 1 | 330 | 165 | 6 | 6.5 |
| 31042 | 20 | 1 | 968 | v5543 | 10 | 0.849 | 3 | 7 | 17 | 0.15 | 0.3 | 0.0171 | 1 | 750 | 375 | 3 | 3.67 |
| 31043 | 31 | 1 | 815 | v8842 | 34 | 0.967 | 12 | 22 | 19 | 0.3871 | 0.3529 | 0.0125 | 1 | 742 | 371 | 12 | 6.92 |
| 31044 | 41 | 1 | 1002 | v8965 | 46 | 0.976 | 9 | 37 | 32 | 0.2195 | 0.1957 | 0.016 | 1 | 40 | 20 | 9 | 7.33 |
| 31045 | 16 | 1 | 796 | v5495 | 10 | 0.847 | 3 | 7 | 13 | 0.1875 | 0.3 | 0.0163 | 1 | 444 | 222 | 3 | 3.67 |
| 31048 | 24 | 1 | 925 | v8878 | 36 | 0.971 | 8 | 28 | 16 | 0.3333 | 0.2222 | 0.0136 | 1 | 492 | 246 | 8 | 7.5 |
| 31051 | 23 | 1 | 621 | v8637 | 22 | 0.953 | 15 | 7 | 8 | 0.6522 | 0.6818 | 0.007 | 1 | 622 | 311 | 15 | 5.6 |
| 31057 | 24 | 1 | 806 | v8156 | 22 | 0.933 | 7 | 15 | 17 | 0.2917 | 0.3182 | 0.0143 | 1 | 546 | 273 | 7 | 6.29 |
| 31059 | 35 | 1 | 1005 | v7636 | 36 | 0.916 | 5 | 31 | 30 | 0.1429 | 0.1389 | 0.0174 | 1 | 440 | 220 | 5 | 5.2 |
| Averages | 30.30 | 1 | 753.18 | | 21.84 | 0.8387 | 8.91 | 12.93 | 21.39 | 0.5009 | 0.5024 | 0.0101 | 1 | 587.37 | 293.68 | 8.91 | 4.65 |

## D.3 TDT3 sample cluster structure average linkage

| *Topic* | | | | *Cluster* | | | *Effectiveness* | | | | | | *Hierarchy above* | | | *Hierarchy below* | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *Id* | *Size* | *Fuzz.* | *#Relevant* | *Id* | *Size* | *Diss.* | *#Union* | *#FA* | *#Miss* | *Recall* | *Precision* | *Det.cost* | *#Root paths* | *Avg enc. branches from root* | *Avg path length from root* | *#Topicdoc paths* | *Avg path length to topicdocs* |
| 30001 | 5 | 1 | 53 | v7104 | 74 | 0.875 | 5 | 69 | 0 | 1 | 0.0676 | 0.0007 | 1 | 28 | 14 | 5 | 13.2 |
| 30002 | 33 | 1 | 158 | v9847 | 263 | 0.976 | 28 | 235 | 5 | 0.8485 | 0.1065 | 0.0053 | 1 | 20 | 10 | 28 | 13.68 |
| 30003 | 60 | 1 | 210 | v9469 | 79 | 0.95 | 56 | 23 | 4 | 0.9333 | 0.7089 | 0.0016 | 1 | 30 | 15 | 56 | 21.77 |
| 30006 | 30 | 1 | 80 | v8495 | 37 | 0.914 | 30 | 7 | 0 | 1 | 0.8108 | 0.0001 | 1 | 28 | 14 | 30 | 11.4 |
| 30014 | 1 | 1 | 15 | d5750 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 28 | 14 | 1 | 0 |
| 30016 | 4 | 1 | 33 | v5946 | 22 | 0.838 | 4 | 18 | 0 | 1 | 0.1818 | 0.0002 | 1 | 40 | 20 | 4 | 4.75 |
| 30017 | 9 | 1 | 95 | v5855 | 39 | 0.836 | 5 | 34 | 4 | 0.5556 | 0.1282 | 0.0092 | 1 | 40 | 20 | 5 | 6.2 |
| 30018 | 1 | 1 | 19 | d7084 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 36 | 18 | 1 | 0 |
| 30022 | 10 | 1 | 42 | v4620 | 15 | 0.792 | 10 | 5 | 0 | 1 | 0.6667 | 0 | 1 | 36 | 18 | 10 | 6.1 |
| 30030 | 3 | 1 | 22 | v4474 | 5 | 0.788 | 3 | 2 | 0 | 1 | 0.6 | 0 | 1 | 30 | 15 | 3 | 2.67 |
| 30031 | 8 | 1 | 37 | v6549 | 22 | 0.858 | 8 | 14 | 0 | 1 | 0.3636 | 0.0001 | 1 | 24 | 12 | 8 | 5.5 |
| 30032 | 1 | 1 | 27 | d8296 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 52 | 26 | 1 | 0 |
| 30033 | 41 | 1 | 274 | v9187 | 70 | 0.937 | 23 | 47 | 18 | 0.561 | 0.3286 | 0.0092 | 1 | 24 | 12 | 23 | 9.13 |
| 30034 | 5 | 1 | 43 | v8412 | 96 | 0.911 | 5 | 91 | 0 | 1 | 0.0521 | 0.0009 | 1 | 28 | 14 | 5 | 9.2 |
| 30038 | 9 | 1 | 32 | v3869 | 10 | 0.765 | 9 | 1 | 0 | 1 | 0.9 | 0 | 1 | 28 | 14 | 9 | 4 |
| 30040 | 17 | 1 | 104 | v7192 | 329 | 0.877 | 17 | 312 | 0 | 1 | 0.0517 | 0.0031 | 1 | 58 | 29 | 17 | 19 |
| 30042 | 12 | 1 | 49 | v8989 | 40 | 0.93 | 12 | 28 | 0 | 1 | 0.3 | 0.0003 | 1 | 26 | 13 | 12 | 9.83 |
| 30043 | 3 | 1 | 36 | v4371 | 11 | 0.784 | 2 | 9 | 1 | 0.6667 | 0.1818 | 0.0068 | 1 | 32 | 16 | 2 | 3 |
| 30046 | 17 | 1 | 134 | v9126 | 328 | 0.934 | 17 | 311 | 0 | 1 | 0.0518 | 0.0031 | 1 | 42 | 21 | 17 | 25.12 |
| 30047 | 10 | 1 | 39 | v6549 | 22 | 0.858 | 10 | 12 | 0 | 1 | 0.4545 | 0.0001 | 1 | 24 | 12 | 10 | 5.6 |
| 30048 | 28 | 1 | 154 | v9414 | 100 | 0.947 | 20 | 80 | 8 | 0.7143 | 0.2 | 0.0065 | 1 | 32 | 16 | 20 | 9.75 |
| 30049 | 6 | 1 | 43 | v5855 | 39 | 0.836 | 6 | 33 | 0 | 1 | 0.1538 | 0.0003 | 1 | 40 | 20 | 6 | 5.83 |
| 30050 | 105 | 1 | 510 | v9341 | 477 | 0.943 | 100 | 377 | 5 | 0.9524 | 0.2096 | 0.0047 | 1 | 38 | 19 | 100 | 22.85 |
| 30051 | 8 | 1 | 96 | v9550 | 456 | 0.954 | 7 | 449 | 1 | 0.875 | 0.0154 | 0.0069 | 1 | 22 | 11 | 7 | 22 |
| 30053 | 34 | 1 | 170 | v8103 | 364 | 0.902 | 33 | 331 | 1 | 0.9706 | 0.0907 | 0.0038 | 1 | 46 | 23 | 33 | 25.91 |
| 30055 | 11 | 1 | 112 | v7767 | 19 | 0.893 | 8 | 11 | 3 | 0.7273 | 0.4211 | 0.0056 | 1 | 32 | 16 | 8 | 6.38 |
| 30057 | 1 | 1 | 17 | d9727 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 32 | 16 | 1 | 0 |
| 31001 | 28 | 1 | 236 | v9596 | 59 | 0.956 | 15 | 44 | 13 | 0.5357 | 0.2542 | 0.0097 | 1 | 20 | 10 | 15 | 7 |
| 31002 | 25 | 1 | 170 | v8838 | 35 | 0.924 | 10 | 25 | 15 | 0.4 | 0.2857 | 0.0122 | 1 | 20 | 10 | 10 | 8 |
| 31004 | 12 | 1 | 101 | v9902 | 148 | 0.982 | 5 | 143 | 7 | 0.4167 | 0.0338 | 0.0131 | 1 | 16 | 8 | 5 | 10 |
| 31006 | 20 | 1 | 163 | v9571 | 29 | 0.955 | 9 | 20 | 11 | 0.45 | 0.3103 | 0.0112 | 1 | 22 | 11 | 9 | 4.89 |
| 31009 | 41 | 1 | 236 | v9688 | 108 | 0.963 | 30 | 78 | 11 | 0.7317 | 0.2778 | 0.0061 | 1 | 24 | 12 | 30 | 13.9 |
| 31012 | 15 | 1 | 122 | v9799 | 126 | 0.972 | 5 | 121 | 10 | 0.3333 | 0.0397 | 0.0145 | 1 | 14 | 7 | 5 | 8 |
| 31013 | 32 | 1 | 273 | v7951 | 18 | 0.898 | 13 | 5 | 19 | 0.4062 | 0.7222 | 0.0119 | 1 | 24 | 12 | 13 | 5.38 |
| 31014 | 25 | 1 | 178 | v9653 | 104 | 0.96 | 17 | 87 | 8 | 0.68 | 0.1635 | 0.0073 | 1 | 18 | 9 | 17 | 12.41 |
| 31016 | 23 | 1 | 121 | v9250 | 55 | 0.94 | 20 | 35 | 3 | 0.8696 | 0.3636 | 0.003 | 1 | 20 | 10 | 20 | 11.3 |
| 31017 | 14 | 1 | 149 | v9151 | 87 | 0.935 | 7 | 80 | 7 | 0.5 | 0.0805 | 0.0108 | 1 | 22 | 11 | 7 | 9.29 |
| 31019 | 27 | 1 | 248 | v9638 | 106 | 0.959 | 16 | 90 | 11 | 0.5926 | 0.1509 | 0.009 | 1 | 26 | 13 | 16 | 9.25 |
| 31020 | 10 | 1 | 114 | v6308 | 29 | 0.851 | 3 | 26 | 7 | 0.3 | 0.1034 | 0.0143 | 1 | 40 | 20 | 3 | 6.33 |
| 31022 | 21 | 1 | 169 | v9457 | 62 | 0.949 | 10 | 52 | 11 | 0.4762 | 0.1613 | 0.011 | 1 | 48 | 24 | 10 | 10.9 |
| 31024 | 27 | 1 | 250 | v3911 | 21 | 0.766 | 5 | 16 | 22 | 0.1852 | 0.2381 | 0.0165 | 1 | 60 | 30 | 5 | 6.4 |
| 31026 | 117 | 1 | 494 | v9495 | 382 | 0.951 | 96 | 286 | 21 | 0.8205 | 0.2513 | 0.0064 | 1 | 20 | 10 | 96 | 14.97 |
| 31029 | 18 | 1 | 160 | v9743 | 237 | 0.967 | 9 | 228 | 9 | 0.5 | 0.038 | 0.0122 | 1 | 26 | 13 | 9 | 14.67 |
| 31031 | 93 | 1 | 565 | v7338 | 150 | 0.881 | 48 | 102 | 45 | 0.5161 | 0.32 | 0.0107 | 1 | 62 | 31 | 48 | 19.12 |
| 31033 | 388 | 1 | 2126 | v9903 | 351 | 0.982 | 109 | 242 | 279 | 0.2809 | 0.3105 | 0.0168 | 1 | 18 | 9 | 109 | 19.44 |
| 31034 | 16 | 1 | 211 | v9803 | 51 | 0.972 | 5 | 46 | 11 | 0.3125 | 0.098 | 0.0142 | 1 | 24 | 12 | 5 | 11.8 |
| 31035 | 37 | 1 | 263 | v9555 | 378 | 0.954 | 23 | 355 | 14 | 0.6216 | 0.0608 | 0.0111 | 1 | 30 | 15 | 23 | 13.48 |
| 31036 | 40 | 1 | 208 | v8627 | 151 | 0.918 | 24 | 127 | 16 | 0.6 | 0.1589 | 0.0092 | 1 | 26 | 13 | 24 | 10.75 |
| 31040 | 12 | 1 | 92 | v5166 | 40 | 0.812 | 7 | 33 | 5 | 0.5833 | 0.175 | 0.0087 | 1 | 44 | 22 | 7 | 8.71 |
| 31042 | 20 | 1 | 224 | v4431 | 10 | 0.786 | 3 | 7 | 17 | 0.15 | 0.3 | 0.0171 | 1 | 38 | 19 | 3 | 3.67 |
| 31043 | 31 | 1 | 153 | v3355 | 17 | 0.744 | 11 | 6 | 20 | 0.3548 | 0.6471 | 0.013 | 1 | 42 | 21 | 11 | 5.82 |
| 31044 | 41 | 1 | 305 | v9514 | 484 | 0.952 | 27 | 457 | 14 | 0.6585 | 0.0558 | 0.0113 | 1 | 34 | 17 | 27 | 17.11 |
| 31045 | 16 | 1 | 152 | v9909 | 328 | 0.983 | 8 | 320 | 8 | 0.5 | 0.0244 | 0.0131 | 1 | 16 | 8 | 8 | 13.75 |
| 31048 | 24 | 1 | 168 | v9867 | 82 | 0.978 | 14 | 68 | 10 | 0.5833 | 0.1707 | 0.009 | 1 | 22 | 11 | 14 | 13.57 |
| 31051 | 23 | 1 | 106 | v9662 | 57 | 0.961 | 16 | 41 | 7 | 0.6957 | 0.2807 | 0.0065 | 1 | 14 | 7 | 16 | 9.69 |
| 31057 | 24 | 1 | 181 | v9653 | 104 | 0.96 | 17 | 87 | 7 | 0.7083 | 0.1635 | 0.0067 | 1 | 18 | 9 | 17 | 11.65 |
| 31059 | 35 | 1 | 347 | v9909 | 328 | 0.983 | 13 | 315 | 22 | 0.3714 | 0.0396 | 0.0157 | 1 | 16 | 8 | 13 | 13.08 |
| Averages | 30.2982 | 1 | 191.0351 | | 123.82 | 0.8437 | 17.84 | 105.98 | 12.46 | 0.7007 | 0.30 | 0.01 | 1 | 30.18 | 15.09 | 17.84 | 10.3 |

## D.4 TDT3 complete cluster structure average linkage

Obtained by adding the remaining documents to the clusters of the first 10 best matching sample documents.

| *Topic* | | | | *Cluster* | | | *Effectiveness* | | | | | | *Hierarchy above* | | | *Hierarchy below* | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *Id* | *Size* | *Fuzz.* | *#Relevant* | *Id* | *Size* | *Diss.* | *#Union* | *#FA* | *#Miss* | *Recall* | *Precision* | *Det.cost* | *#Root paths* | *Avg enc. branches from root* | *Avg path length from root* | *#Topicdoc paths* | *Avg path length to topicdocs* |
| 30001 | 26 | 1.2692 | 97 | v7104 | 288 | 0.875 | 24 | 264 | 2 | 0.9231 | 0.0833 | 0.0023 | 1 | 15 | 6 | 31 | 12.74 |
| 30002 | 108 | 1.0463 | 236 | v8761 | 363 | 0.922 | 81 | 282 | 27 | 0.75 | 0.2231 | 0.0058 | 1 | 17 | 7 | 85 | 8.39 |
| 30003 | 168 | 1.4583 | 145 | v9469 | 373 | 0.95 | 159 | 214 | 9 | 0.9464 | 0.4263 | 0.0017 | 1 | 13 | 5 | 236 | 22.44 |
| 30006 | 114 | 1.0088 | 191 | v9331 | 189 | 0.943 | 102 | 87 | 12 | 0.8947 | 0.5397 | 0.0023 | 1 | 12 | 5 | 103 | 13.26 |
| 30009 | 10 | 1 | 23 | v4770 | 26 | 0.798 | 9 | 17 | 1 | 0.9 | 0.3462 | 0.002 | 1 | 26 | 11 | 9 | 1.22 |
| 30014 | 10 | 1 | 11 | v3521 | 21 | 0.751 | 10 | 11 | 0 | 1 | 0.4762 | 0 | 1 | 24 | 9 | 10 | 0.8 |
| 30016 | 12 | 1 | 31 | v6529 | 137 | 0.857 | 12 | 125 | 0 | 1 | 0.0876 | 0.0003 | 1 | 29 | 13 | 12 | 5.58 |
| 30017 | 22 | 1 | 70 | v5855 | 126 | 0.836 | 14 | 112 | 8 | 0.6364 | 0.1111 | 0.0076 | 1 | 30 | 13 | 14 | 4.71 |
| 30018 | 10 | 1 | 96 | v8610 | 220 | 0.917 | 6 | 214 | 4 | 0.6 | 0.0273 | 0.0086 | 1 | 18 | 7 | 6 | 5 |
| 30021 | 3 | 1 | 31 | v9019 | 256 | 0.93 | 2 | 254 | 1 | 0.6667 | 0.0078 | 0.0074 | 1 | 22 | 9 | 2 | 5 |
| 30022 | 36 | 1.0278 | 60 | v9151 | 332 | 0.935 | 35 | 297 | 1 | 0.9722 | 0.1054 | 0.0014 | 1 | 16 | 6 | 36 | 12.03 |
| 30023 | 7 | 1 | 62 | v5171 | 6 | 0.812 | 2 | 4 | 5 | 0.2857 | 0.3333 | 0.0143 | 1 | 16 | 7 | 2 | 0 |
| 30030 | 14 | 1 | 37 | v6546 | 103 | 0.858 | 13 | 90 | 1 | 0.9286 | 0.1262 | 0.0017 | 1 | 21 | 9 | 13 | 4.46 |
| 30031 | 33 | 1.4848 | 27 | v6549 | 134 | 0.858 | 33 | 101 | 0 | 1 | 0.2463 | 0.0003 | 1 | 21 | 9 | 49 | 4.31 |
| 30032 | 10 | 1 | 51 | v6232 | 14 | 0.848 | 5 | 9 | 5 | 0.5 | 0.3571 | 0.01 | 1 | 28 | 13 | 5 | 0 |
| 30033 | 118 | 1.0254 | 282 | v9187 | 273 | 0.937 | 67 | 206 | 51 | 0.5678 | 0.2454 | 0.0092 | 1 | 16 | 6 | 70 | 8.41 |
| 30034 | 22 | 1.1818 | 48 | v8412 | 395 | 0.911 | 22 | 373 | 0 | 1 | 0.0557 | 0.001 | 1 | 16 | 7 | 26 | 9.15 |
| 30038 | 28 | 1.25 | 50 | v3869 | 41 | 0.765 | 27 | 14 | 1 | 0.9643 | 0.6585 | 0.0008 | 1 | 27 | 12 | 34 | 3.35 |
| 30040 | 45 | 1 | 138 | v7192 | 1337 | 0.877 | 43 | 1294 | 2 | 0.9556 | 0.0322 | 0.0045 | 1 | 37 | 19 | 43 | 20.84 |
| 30042 | 47 | 1.617 | 75 | v8989 | 267 | 0.93 | 46 | 221 | 1 | 0.9787 | 0.1723 | 0.001 | 1 | 14 | 5 | 74 | 10.08 |
| 30043 | 13 | 1 | 33 | v4371 | 55 | 0.784 | 11 | 44 | 2 | 0.8462 | 0.2 | 0.0032 | 1 | 21 | 9 | 11 | 2.18 |
| 30046 | 60 | 1 | 126 | v7256 | 515 | 0.879 | 54 | 461 | 6 | 0.9 | 0.1049 | 0.0033 | 1 | 38 | 18 | 54 | 18.41 |
| 30047 | 38 | 1.4211 | 30 | v7745 | 146 | 0.893 | 38 | 108 | 0 | 1 | 0.2603 | 0.0003 | 1 | 19 | 8 | 54 | 5.33 |
| 30048 | 92 | 1 | 192 | v9555 | 1416 | 0.954 | 72 | 1344 | 20 | 0.7826 | 0.0508 | 0.0081 | 1 | 11 | 4 | 72 | 11.38 |
| 30049 | 19 | 1 | 70 | v5855 | 126 | 0.836 | 14 | 112 | 5 | 0.7368 | 0.1111 | 0.0056 | 1 | 30 | 13 | 14 | 5.14 |
| 30050 | 377 | 1.0053 | 615 | v9341 | 1810 | 0.943 | 337 | 1473 | 40 | 0.8939 | 0.1862 | 0.0063 | 1 | 12 | 5 | 339 | 23 |
| 30051 | 42 | 1.0476 | 153 | v9550 | 1778 | 0.954 | 40 | 1738 | 2 | 0.9524 | 0.0225 | 0.0058 | 1 | 8 | 3 | 42 | 21.95 |
| 30053 | 128 | 1.0469 | 284 | v8103 | 1433 | 0.902 | 113 | 1320 | 15 | 0.8828 | 0.0789 | 0.006 | 1 | 26 | 13 | 119 | 27.53 |
| 30055 | 41 | 1.2683 | 139 | v7767 | 80 | 0.893 | 28 | 52 | 13 | 0.6829 | 0.35 | 0.0065 | 1 | 19 | 7 | 39 | 6.44 |
| 30057 | 3 | 1 | 11 | v5253 | 9 | 0.815 | 3 | 6 | 0 | 1 | 0.3333 | 0 | 1 | 25 | 10 | 3 | 0 |
| 31001 | 102 | 1.0098 | 440 | v9596 | 204 | 0.956 | 52 | 152 | 50 | 0.5098 | 0.2549 | 0.0102 | 1 | 14 | 5 | 53 | 6.43 |

Continued on next page.

Continued from previous page:

| *Topic* | | | | *Cluster* | | | *Effectiveness* | | | | | | *Hierarchy above* | | | *Hierarchy below* | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *Id* | *Size* | *Fuzz.* | *#Relevant* | *Id* | *Size* | *Diss.* | *#Union* | *#FA* | *#Miss* | *Recall* | *Precision* | *Det.cost* | *#Root paths* | *Avg enc. branches from root* | *Avg path length from root* | *#Topicdoc paths* | *Avg path length to topicdocs* |
| 31002 | 112 | 1.0893 | 376 | v9529 | 253 | 0.953 | 48 | 205 | 64 | 0.4286 | 0.1897 | 0.012 | 1 | 14 | 5 | 55 | 7.29 |
| 31004 | 47 | 1 | 204 | v6130 | 52 | 0.845 | 10 | 42 | 37 | 0.2128 | 0.1923 | 0.0159 | 1 | 16 | 7 | 10 | 5.5 |
| 31006 | 65 | 1.0308 | 380 | v9571 | 117 | 0.955 | 25 | 92 | 40 | 0.3846 | 0.2137 | 0.0126 | 1 | 17 | 6 | 25 | 4.36 |
| 31009 | 133 | 1.2707 | 409 | v8987 | 584 | 0.93 | 89 | 495 | 44 | 0.6692 | 0.1524 | 0.008 | 1 | 11 | 4 | 123 | 12.63 |
| 31012 | 53 | 1 | 284 | n140 | 826 | 1 | 18 | 808 | 35 | 0.3396 | 0.0218 | 0.0155 | 1 | 11 | 4 | 18 | 11.94 |
| 31013 | 113 | 1 | 488 | v7951 | 62 | 0.898 | 34 | 28 | 79 | 0.3009 | 0.5484 | 0.0141 | 1 | 19 | 7 | 34 | 4.62 |
| 31014 | 83 | 1.012 | 274 | v9382 | 366 | 0.945 | 55 | 311 | 28 | 0.6627 | 0.1503 | 0.0076 | 1 | 14 | 5 | 56 | 11.34 |
| 31016 | 87 | 1.2644 | 208 | v9250 | 262 | 0.94 | 73 | 189 | 14 | 0.8391 | 0.2786 | 0.0037 | 1 | 16 | 6 | 96 | 11.51 |
| 31017 | 49 | 1 | 260 | v9151 | 332 | 0.935 | 18 | 314 | 31 | 0.3673 | 0.0542 | 0.0135 | 1 | 16 | 6 | 18 | 8.78 |
| 31019 | 86 | 1 | 398 | v9638 | 442 | 0.959 | 50 | 392 | 36 | 0.5814 | 0.1131 | 0.0095 | 1 | 14 | 5 | 50 | 9.26 |
| 31020 | 55 | 1.0545 | 273 | v9122 | 90 | 0.934 | 11 | 79 | 44 | 0.2 | 0.1222 | 0.0162 | 1 | 17 | 6 | 11 | 3.73 |
| 31022 | 86 | 1 | 299 | v8753 | 115 | 0.921 | 28 | 87 | 58 | 0.3256 | 0.2435 | 0.0137 | 1 | 22 | 9 | 28 | 4.75 |
| 31024 | 92 | 1.0435 | 508 | n130 | 621 | 1 | 28 | 593 | 64 | 0.3043 | 0.0451 | 0.0156 | 1 | 11 | 4 | 28 | 9.29 |
| 31026 | 477 | 1.0105 | 576 | v9495 | 1518 | 0.951 | 390 | 1128 | 87 | 0.8176 | 0.2569 | 0.0068 | 1 | 8 | 3 | 394 | 15.77 |
| 31029 | 79 | 1.0886 | 316 | v9258 | 287 | 0.94 | 17 | 270 | 62 | 0.2152 | 0.0592 | 0.0164 | 1 | 17 | 7 | 17 | 9.65 |
| 31031 | 307 | 1.0098 | 808 | v7338 | 617 | 0.881 | 154 | 463 | 153 | 0.5016 | 0.2496 | 0.0113 | 1 | 36 | 17 | 154 | 18.55 |
| 31033 | 1346 | 1.0171 | 2780 | v9513 | 716 | 0.952 | 268 | 448 | 1078 | 0.1991 | 0.3743 | 0.0173 | 1 | 13 | 5 | 276 | 14.26 |
| 31034 | 63 | 1.0635 | 511 | v5338 | 15 | 0.818 | 9 | 6 | 54 | 0.1429 | 0.6 | 0.0172 | 1 | 23 | 9 | 9 | 1.22 |
| 31035 | 159 | 1 | 544 | v9555 | 1416 | 0.954 | 84 | 1332 | 75 | 0.5283 | 0.0593 | 0.0132 | 1 | 11 | 4 | 84 | 13.26 |
| 31036 | 152 | 1.0197 | 278 | v8627 | 665 | 0.918 | 101 | 564 | 51 | 0.6645 | 0.1519 | 0.0083 | 1 | 13 | 6 | 104 | 12.12 |
| 31040 | 40 | 1.45 | 183 | v9197 | 275 | 0.937 | 16 | 259 | 24 | 0.4 | 0.0582 | 0.0127 | 1 | 16 | 6 | 18 | 6.5 |
| 31042 | 85 | 1 | 499 | v1944 | 16 | 0.679 | 7 | 9 | 78 | 0.0824 | 0.4375 | 0.0184 | 1 | 28 | 13 | 7 | 0 |
| 31043 | 93 | 1.2688 | 205 | v6817 | 48 | 0.866 | 32 | 16 | 61 | 0.3441 | 0.6667 | 0.0132 | 1 | 19 | 7 | 36 | 3.31 |
| 31044 | 119 | 1.0084 | 396 | v9514 | 1969 | 0.952 | 70 | 1899 | 49 | 0.5882 | 0.0356 | 0.0135 | 1 | 12 | 5 | 70 | 16.43 |
| 31045 | 48 | 1.0625 | 314 | v8610 | 220 | 0.917 | 15 | 205 | 33 | 0.3125 | 0.0682 | 0.0143 | 1 | 18 | 7 | 15 | 6.87 |
| 31048 | 68 | 1.0441 | 197 | n131 | 639 | 1 | 43 | 596 | 25 | 0.6324 | 0.0673 | 0.009 | 1 | 11 | 4 | 45 | 12.51 |
| 31051 | 88 | 1.0795 | 315 | v9516 | 149 | 0.952 | 40 | 109 | 48 | 0.4545 | 0.2685 | 0.0112 | 1 | 17 | 6 | 47 | 8.49 |
| 31057 | 99 | 1.0404 | 322 | v9382 | 366 | 0.945 | 63 | 303 | 36 | 0.6364 | 0.1721 | 0.0081 | 1 | 14 | 5 | 64 | 10.97 |
| 31059 | 133 | 1.1053 | 608 | v9151 | 332 | 0.935 | 23 | 309 | 110 | 0.1729 | 0.0693 | 0.0174 | 1 | 16 | 6 | 24 | 10.08 |
| Averages | 102.92 | 1.09 | 284.45 | | 430.72 | 0.9022 | 54.88 | 375.83 | 48.0333 | 0.6328 | 0.2089 | 0.0084 | 1 | 18.52 | 7.62 | 59.6 | 9.08 |

Appendix E

# Demonstration

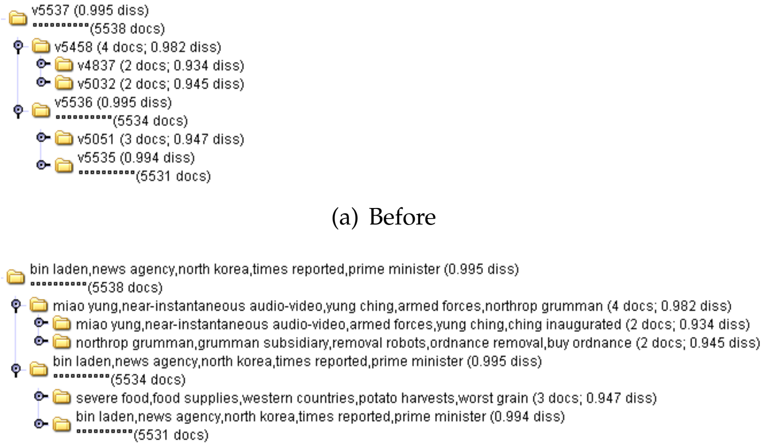
In this appendix a few demonstrations are given to actually use the created cluster structure. The goal of this appendix is to show that despite its lack of precision, the cluster structure still is capable of providing an interface for exploring and navigating a collection of documents. The appendix will not give an in depth analysis of these visualizations, but hopes to serve as an inspiration for further research and experiments to utilize the proposed cluster structure.

## E.1 Visualization in a tree

A simple tree widget available as a basic user interface component in most operating systems can be used to display the directed acyclic graph. Figure E.1 shows two examples of such a visualization.

Until now, no metadata was added to the created clusters. The clusters were nameless and only identifiable by its relationships with other clusters and documents. A simple labelling algorithm is used to describe the clusters: the most informative bigrams (sequences of two terms) in the documents contained by the cluster. The result is also shown in figure E.1. Notice that the parent label is the same as the label of the child cluster containing the most documents. As a result the child cluster cannot be easily located as it is overgrown by a large “brother”. Furthermore the binary branching factor requires much effort to acquire more information: many clicks are needed to reveal more information.

To overcome this problem a slider is added, which can cut the cluster structure at a certain dissimilarity threshold. Clusters higher in the tree have a higher dissimilarity value than their children. The clusters which have a



```
v5537 (0.995 diss)
▪▪▪▪▪▪▪▪▪▪(5538 docs)
  v5458 (4 docs; 0.982 diss)
    v4837 (2 docs; 0.934 diss)
    v5032 (2 docs; 0.945 diss)
  v5536 (0.995 diss)
  ▪▪▪▪▪▪▪▪▪▪(5534 docs)
    v5051 (3 docs; 0.947 diss)
    v5535 (0.994 diss)
    ▪▪▪▪▪▪▪▪▪▪(5531 docs)
```

(a) Before

```
bin laden,news agency,north korea,times reported,prime minister (0.995 diss)
▪▪▪▪▪▪▪▪▪▪(5538 docs)
  miao yung,near-instantaneous audio-video,yung ching,armed forces,northrop grumman (4 docs; 0.982 diss)
    miao yung,near-instantaneous audio-video,armed forces,yung ching,ching inaugurated (2 docs; 0.934 diss)
    northrop grumman,grumman subsidiary,removal robots,ordnance removal,buy ordnance (2 docs; 0.945 diss)
  bin laden,news agency,north korea,times reported,prime minister (0.995 diss)
  ▪▪▪▪▪▪▪▪▪▪(5534 docs)
    severe food,food supplies,western countries,potato harvests,worst grain (3 docs; 0.947 diss)
    bin laden,news agency,north korea,times reported,prime minister (0.994 diss)
    ▪▪▪▪▪▪▪▪▪▪(5531 docs)
```

(b) After

Figure E.1: The result of labelling

dissimilarity just below the threshold and have a parent with a dissimilarity above the threshold are used as the top level clusters in the treeview.

Figure E.2 shows the result of cutting the cluster structure at a high level. Note that the number of clusters displayed as root has increased, but also more information has become available about the document collection.

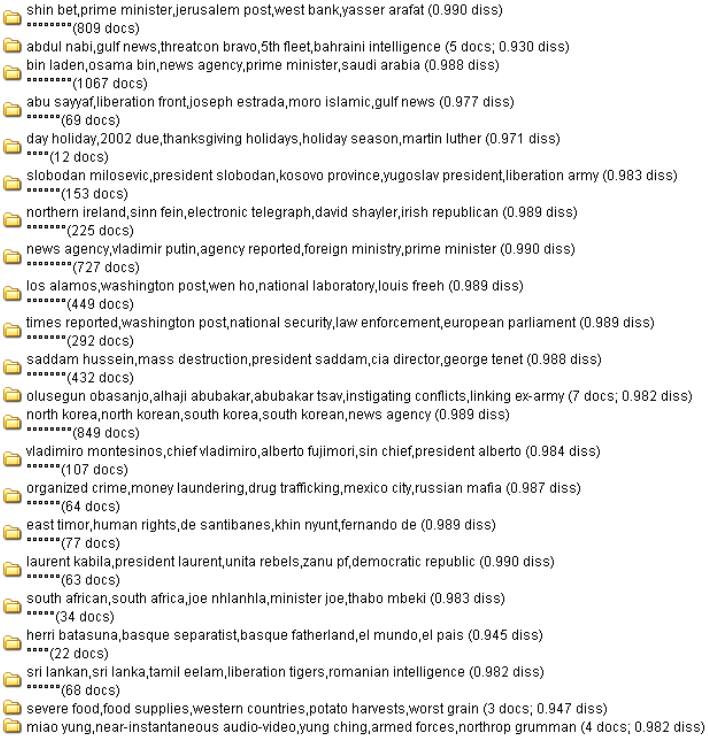
Cutting at a lower level presents the user with a long list of clusters, sometimes having overlapping bigrams. Figure E.3 shows a screenshot of such a cut.

Obviously this is only a start to visualize the structure using a treeview. Possible improvements could be another slider to filter on the size of clusters, or an option to gather intersting clusters (found during some browsing session) in a personalized basket.

## E.2 “Reuterizing” the cluster structure

In an attempt to make the top of the cluster structure more usable, a small experiment was carried out using the Reuters Corpus. The Reuters Corpus is a large collection of news items, each item is annotated with the geographic location and a general news category (e.g. sports, education, crime etc).

By creating an index of the Reuters documents the documents from a cluster structure can be “annotated”: the documents are used as queries and the annotation of the best Reuters documents is used as annotation. The cluster structure is cut at some level (as described before), and the root clusters

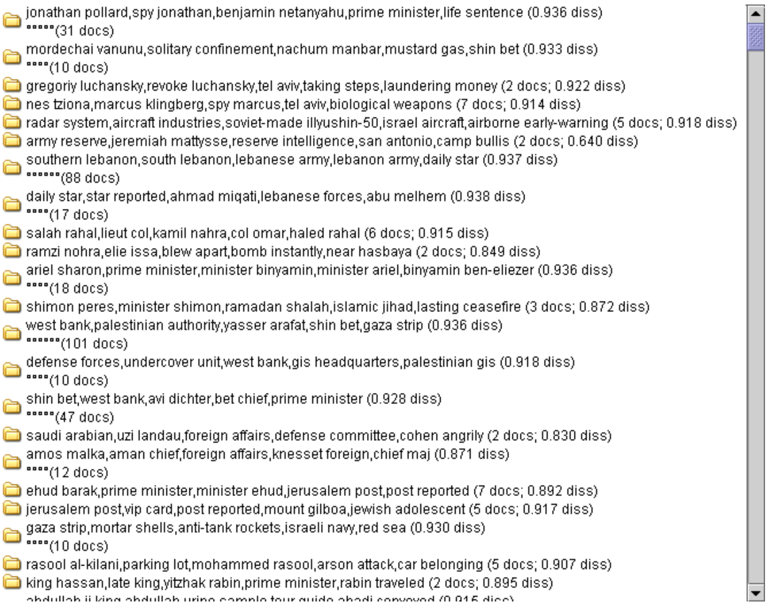


```
shin bet,prime minister,jerusalem post,west bank,yasser arafat (0.990 diss)
▪▪▪▪▪▪▪▪(809 docs)
abdul nabi,gulf news,threatcon bravo,5th fleet,bahraini intelligence (5 docs; 0.930 diss)
bin laden,osama bin,news agency,prime minister,saudi arabia (0.988 diss)
▪▪▪▪▪▪▪▪(1067 docs)
abu sayyaf,liberation front,joseph estrada,moro islamic,gulf news (0.977 diss)
▪▪▪▪▪▪(69 docs)
day holiday,2002 due,thanksgiving holidays,holiday season,martin luther (0.971 diss)
▪▪▪▪(12 docs)
slobodan milosevic,president slobodan,kosovo province,yugoslav president,liberation army (0.983 diss)
▪▪▪▪▪▪(153 docs)
northern ireland,sinn fein,electronic telegraph,david shayler,irish republican (0.989 diss)
▪▪▪▪▪▪▪(225 docs)
news agency,vladimir putin,agency reported,foreign ministry,prime minister (0.990 diss)
▪▪▪▪▪▪▪▪(727 docs)
los alamos,washington post,wen ho,national laboratory,louis freeh (0.989 diss)
▪▪▪▪▪▪▪(449 docs)
times reported,washington post,national security,law enforcement,european parliament (0.989 diss)
▪▪▪▪▪▪▪(292 docs)
saddam hussein,mass destruction,president saddam,cia director,george tenet (0.988 diss)
▪▪▪▪▪▪▪(432 docs)
olusegun obasanjo,alhaji abubakar,abubakar tsav,instigating conflicts,linking ex-army (7 docs; 0.982 diss)
north korea,north korean,south korea,south korean,news agency (0.989 diss)
▪▪▪▪▪▪▪▪(849 docs)
vladimiro montesinos,chief vladimiro,alberto fujimori,sin chief,president alberto (0.984 diss)
▪▪▪▪▪▪(107 docs)
organized crime,money laundering,drug trafficking,mexico city,russian mafia (0.987 diss)
▪▪▪▪▪▪(64 docs)
east timor,human rights,de santibanes,khin nyunt,fernando de (0.989 diss)
▪▪▪▪▪▪(77 docs)
laurent kabila,president laurent,unita rebels,zanu pf,democratic republic (0.990 diss)
▪▪▪▪▪▪(63 docs)
south african,south africa,joe nhlanhla,minister joe,thabo mbeki (0.983 diss)
▪▪▪▪▪(34 docs)
herri batasuna,basque separatist,basque fatherland,el mundo,el pais (0.945 diss)
▪▪▪▪(22 docs)
sri lankan,sri lanka,tamil eelam,liberation tigers,romanian intelligence (0.982 diss)
▪▪▪▪▪▪(68 docs)
severe food,food supplies,western countries,potato harvests,worst grain (3 docs; 0.947 diss)
miao yung,near-instantaneous audio-video,yung ching,armed forces,northrop grumman (4 docs; 0.982 diss)
```

Figure E.2: Cutting at a certain dissimilarity level

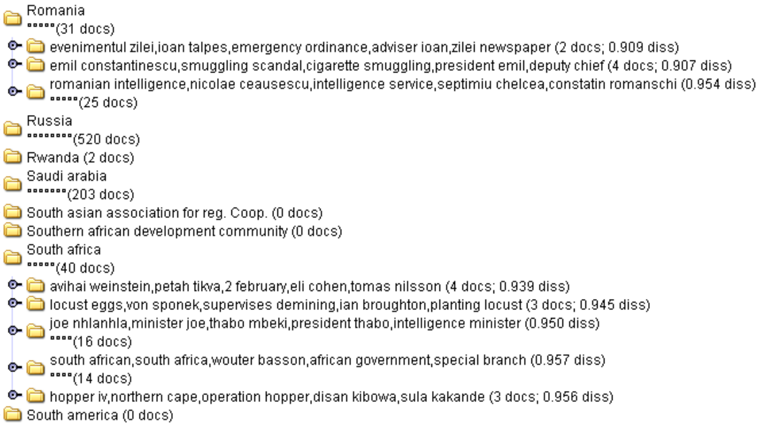
are annotated using the annotation of the documents contained in those clusters. If for example a cluster contains 30 documents and 15 documents have been annotated with the category “crime”, the cluster is annotated with this category.

The resulting ”merged” cluster structure offers potential; some clusters are assigned correctly to a particular country or category. Figures E.4 and E.5 give a small impression of the results. The approach gives the impression that using these strict categorization does rise user expectations of a particular cluster: A label ”Romania” does give the impression the child clusters are only about events in Romania; an incorrectly annotated child cluster seems to disappoint more when the label is less ambiguous.



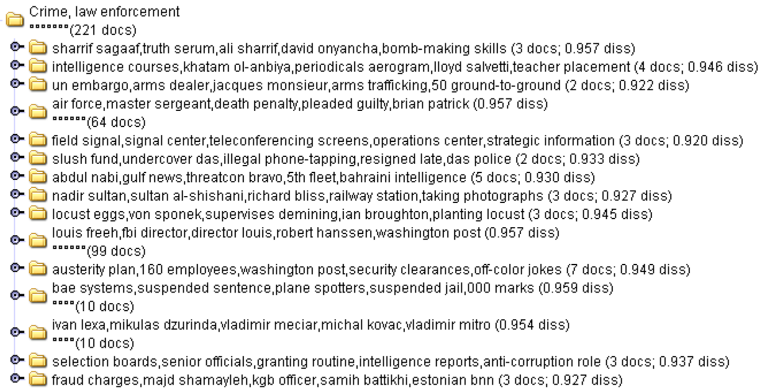
```
jonathan pollard,spy jonathan,benjamin netanyahu,prime minister,life sentence (0.936 diss)
▪▪▪▪▪(31 docs)
mordechai vanunu,solitary confinement,nachum manbar,mustard gas,shin bet (0.933 diss)
▪▪▪▪(10 docs)
gregoriy luchansky,revoke luchansky,tel aviv,taking steps,laundering money (2 docs; 0.922 diss)
nes tziona,marcus klingberg,spy marcus,tel aviv,biological weapons (7 docs; 0.914 diss)
radar system,aircraft industries,soviet-made illyushin-50,israel aircraft,airborne early-warning (5 docs; 0.918 diss)
army reserve,jeremiah mattysse,reserve intelligence,san antonio,camp bullis (2 docs; 0.640 diss)
southern lebanon,south lebanon,lebanese army,lebanon army,daily star (0.937 diss)
▪▪▪▪▪▪(88 docs)
daily star,star reported,ahmad miqati,lebanese forces,abu melhem (0.938 diss)
▪▪▪▪(17 docs)
salah rahal,lieut col,kamil nahra,col omar,haled rahal (6 docs; 0.915 diss)
ramzi nohra,elie issa,blew apart,bomb instantly,near hasbaya (2 docs; 0.849 diss)
ariel sharon,prime minister,minister binyamin,minister ariel,binyamin ben-eliezer (0.936 diss)
▪▪▪▪(18 docs)
shimon peres,minister shimon,ramadan shalah,islamic jihad,lasting ceasefire (3 docs; 0.872 diss)
west bank,palestinian authority,yasser arafat,shin bet,gaza strip (0.936 diss)
▪▪▪▪▪▪(101 docs)
defense forces,undercover unit,west bank,gis headquarters,palestinian gis (0.918 diss)
▪▪▪▪(10 docs)
shin bet,west bank,avi dichter,bet chief,prime minister (0.928 diss)
▪▪▪▪▪(47 docs)
saudi arabian,uzi landau,foreign affairs,defense committee,cohen angrily (2 docs; 0.830 diss)
amos malka,aman chief,foreign affairs,knesset foreign,chief maj (0.871 diss)
▪▪▪▪(12 docs)
ehud barak,prime minister,minister ehud,jerusalem post,post reported (7 docs; 0.892 diss)
jerusalem post,vip card,post reported,mount gilboa,jewish adolescent (5 docs; 0.917 diss)
gaza strip,mortar shells,anti-tank rockets,israeli navy,red sea (0.930 diss)
▪▪▪▪(10 docs)
rasool al-kilani,parking lot,mohammed rasool,arson attack,car belonging (5 docs; 0.907 diss)
king hassan,late king,yitzhak rabin,prime minister,rabin traveled (2 docs; 0.895 diss)
abdullah ii,king abdullah,urine sample,tour guide,abadi conveyed (0.915 diss)
```

Figure E.3: Cutting at a lower dissimilarity level yields more clusters



```
Romania
▪▪▪▪▪(31 docs)
    evenimentul zilei,ioan talpes,emergency ordinance,adviser ioan,zilei newspaper (2 docs; 0.909 diss)
    emil constantinescu,smuggling scandal,cigarette smuggling,president emil,deputy chief (4 docs; 0.907 diss)
    romanian intelligence,nicolae ceausescu,intelligence service,septimiu chelcea,constatin romanschi (0.954 diss)
    ▪▪▪▪▪(25 docs)
Russia
▪▪▪▪▪▪▪▪(520 docs)
Rwanda (2 docs)
Saudi arabia
▪▪▪▪▪▪▪(203 docs)
South asian association for reg. Coop. (0 docs)
Southern african development community (0 docs)
South africa
▪▪▪▪▪(40 docs)
    avihai weinstein,petah tikva,2 february,eli cohen,tomas nilsson (4 docs; 0.939 diss)
    locust eggs,von sponek,supervises demining,ian broughton,planting locust (3 docs; 0.945 diss)
    joe nhlanhla,minister joe,thabo mbeki,president thabo,intelligence minister (0.950 diss)
    ▪▪▪▪(16 docs)
    south african,south africa,wouter basson,african government,special branch (0.957 diss)
    ▪▪▪▪(14 docs)
    hopper iv,northern cape,operation hopper,disan kibowa,sula kakande (3 docs; 0.956 diss)
South america (0 docs)
```

Figure E.4: Using Reuters countries



```
Crime, law enforcement
▫▫▫▫▫▫▫(221 docs)
  sharrif sagaaf,truth serum,ali sharrif,david onyancha,bomb-making skills (3 docs; 0.957 diss)
  intelligence courses,khatam ol-anbiya,periodicals aerogram,lloyd salvetti,teacher placement (4 docs; 0.946 diss)
  un embargo,arms dealer,jacques monsieur,arms trafficking,50 ground-to-ground (2 docs; 0.922 diss)
  air force,master sergeant,death penalty,pleaded guilty,brian patrick (0.957 diss)
  ▫▫▫▫▫▫(64 docs)
  field signal,signal center,teleconferencing screens,operations center,strategic information (3 docs; 0.920 diss)
  slush fund,undercover das,illegal phone-tapping,resigned late,das police (2 docs; 0.933 diss)
  abdul nabi,gulf news,threatcon bravo,5th fleet,bahraini intelligence (5 docs; 0.930 diss)
  nadir sultan,sultan al-shishani,richard bliss,railway station,taking photographs (3 docs; 0.927 diss)
  locust eggs,von sponek,supervises demining,ian broughton,planting locust (3 docs; 0.945 diss)
  louis freeh,fbi director,director louis,robert hanssen,washington post (0.957 diss)
  ▫▫▫▫▫▫(99 docs)
  austerity plan,160 employees,washington post,security clearances,off-color jokes (7 docs; 0.949 diss)
  bae systems,suspended sentence,plane spotters,suspended jail,000 marks (0.959 diss)
  ▫▫▫▫(10 docs)
  ivan lexa,mikulas dzurinda,vladimir meciar,michal kovac,vladimir mitro (0.954 diss)
  ▫▫▫▫(10 docs)
  selection boards,senior officials,granting routine,intelligence reports,anti-corruption role (3 docs; 0.937 diss)
  fraud charges,majd shamayleh,kgb officer,samih battikhi,estonian bnn (3 docs; 0.927 diss)
```

Figure E.5: Using Reuters categories