Document Classification and Visualisation to Support the Investigation of Suspected Fraud

Johan Hagman[1], Domenico Perrotta[2], Ralf Steinberger[1], and Aristide Varfis[1]

[1] European Commission – Joint Research Centre (JRC)
Institute for Systems, Informatics and Safety (ISIS)
Risk Management and Decision Support (RMDS)
T.P. 361, 21020 Ispra (VA), Italy
{johan.hagman, ralf.steinberger, aristide.varfis}@jrc.it
[2] European Commission – Information Society DG
Information Society Technologies: Content, Multimedia Tools and Markets
Linguistic Applications, including the "Multilingualism" Programme
Domenico.Perrotta@cec.eu.int

Abstract This position paper reports on ongoing work where three clustering and visualisation techniques for large document collections – developed at the Joint Research Centre (JRC) – are applied to textual data to support the European Commission's investigation on suspected fraud cases. The techniques are (a) an implementation of the neural network application WEBSOM, (b) hierarchical cluster analysis, and (c) a method to present collections in two-dimensional space which is based on previous hierarchical clustering. In order to put these three techniques into their context, we describe the general design of a multilingual document retrieval, information extraction and visualisation system which is being developed at the JRC to support the Anti-Fraud Office (OLAF) of the European Commission in their fight against fraud. The description includes information on the individual components of the system, i.e. an agent to retrieve documents from the internet, a language recogniser, a tool to recognise geographical references in text, a keyword identification tool, as well as a word clustering component.

1 Introduction

Like any other organisation, the European Commission (EC) has a need to monitor activities and events in their fields of interest, including that of fraud detection. One way of keeping abreast of recent developments is by gathering relevant documents which are available from open sources such as the internet or which are stored on intranets and on the extranets of associated organisations. An added challenge for a multinational organisation such as the EC is the fact that textual data is written in a variety of languages. Information-related applications developed for the EC will thus have to pay particular attention to the multilinguality aspect.

In this paper we present a set of tools which are becoming part of a more complex information retrieval and visualisation system. After an introduction in which we describe our goals (1.1 and 1.2) and specify the status of our work (1.3), we briefly discuss the individual tools we use to prepare the textual data for visualisation (section 2). The third section discusses three different components which we created in-house to meet our specific needs for viewing document and keyword collections in a variety of ways, i.e. by displaying whole document collections hierarchically or in a two-dimensional space. Section 4 contains some remarks on the multilingual aspect of the planned system and on the evaluation of the visualisation techniques.

1.1 The role of the JRC and the goal of the ISIS-RMDS NLP activities

The mission of the *Joint Research Centre* (JRC) is to provide customer-driven scientific and technical support for the conception, development, implementation and monitoring of EU policies[1]. The JRC carries out research and development for other Commission services and evaluates commercial software solutions. Among other things, the JRC Unit *Risk Management and Decision Support* (RMDS) of the *Institute for Systems, Informatics and Safety* (ISIS) has the task of providing their customers with solutions for dealing with large amounts of information written in natural language.

In order to inform selected groups of Commission staff such as the members of the European Anti-Fraud Office OLAF about what is happening in their fields of interest, the JRC is developing an agent-based system which retrieves relevant documents on a regular basis, extracts information from them and automatically presents reports to the users which contain links to the most relevant documents plus the extracted information.

As the number of potentially relevant documents is rather large, it is crucial to present the retrieved data in an intuitive form, by visualising the document collections and their contents in an intelligible way. The JRC is working on several alternative visualisation methods to reflect the different kinds of data and results and to offer the users several alternatives, leaving it to them to choose the one which suits them best for each type of data and purpose.

Once the most relevant documents and their contents have been identified over a period of time, there will be enough data to calculate not only relations between individual parameters (specific pieces of information such as, for instance, geographic information and groups of products), but also changes of the parameters and their relations over time. In the long term, we want a tool which points out major developments by visualising the main parameters and their diachronic changes with respect to other parameters.

1.2 Where we are now

The JRC is not yet at the point of having realised the whole system described in 1.1, but we have produced and gathered a number of working tools which will be part of the whole system. These include an agent-based system which retrieves documents matching some keywords from the internet (2.1), a language recogniser (2.2), a tool to identify geographical references in texts (2.3), and an automatic keyword identification tool (2.4). Furthermore, we have developed a series of alternative tools to cluster similar words or documents and to visualise the document collection, producing both document hierarchies (3.2) and alternative two-dimensional document maps showing the relative vicinity and relatedness of the documents in the collection (3.1 and 3.3). We also used the clustering tool to cluster indexing words (2.6) which we extracted from the documents of our test collection.

The examples in this paper refer to a small text collection of 260 English questions made to the European Parliament, including the official answers (EP-Q&A) and our methods will be illustrated using that data.

2 Tools and Techniques

In the following, we provide a brief description of some of the tools we have chosen to develop ourselves and which we intend to use in the information gathering, extraction and presentation system.

[1] See: http://www.jrc.org/jrc/index.asp

2.1 Safe retrieval of documents from the WWW

We are currently using two alternative systems to retrieve documents from the internet and are in the process of developing a third. All three systems have the common goal of retrieving and downloading documents from internal and external sites which satisfy a number of conditions, the primary of which is that they match one or more search words.

The main feature of our in-house system is that it allows defining the search words by browsing through one or more thesauri and that, for security reasons, it automatically strips out active contents from the retrieved files. The second tool is the commercial agent software *Teleport Pro*[2] which searches a user-defined set of web sites for files and retrieves the ones satisfying a number of conditions. The third tool, which is under development, has similar functionality to *Teleport Pro*, but allows a more sophisticated formulation of the conditions.

2.2 Language recognition

Some of the texts the JRC was asked to analyse and visualise are truly multilingual in the sense that the language in which they are written changes inside the text, and occasionally even in the middle of a paragraph. We have therefore developed a purely bigram-based language recognition tool, working without any lexical information. It determines for each paragraph at a time which language it is likely to be written in. It does this by first deciding for each paragraph on a language A, and then it marks up islands of suspected non-A language words. The "suspicion sensitivity" and the size of such an island are parametrisable. Figure 1 shows how the predominant language in the paragraphs is successfully recognised in spite of allowing islands of words which *could* belong to another language. Each of the languages is represented by one colour/shade of grey.

Figure 1. Visualisation of language recognition results. The "suspicion sensitivity" was set to be very high.

For some languages we had to train the tool on both accentuated and non-accentuated texts in order to guarantee good performance because we have to deal with large amounts of texts which are written with diacritics omitted. The performance of the language recogniser is rather good even though it sometimes stumbles when being confronted with very short paragraphs or with paragraphs containing many abbreviations, foreign names and addresses.

[2] See http://www.tenmax.com/teleport/pro/home.htm

ignalina	221937.67	seize	3836.83	makeup	1353.95	emit	1290.11
material	913.23	lithuanian	823.75	nuclear	729.36	fuel	525.96
uranium	456.52	radioactive	306.87	plant	278.40	facility	115.44
arrest	112.44	component	93.13	origin	84.31		

Table 1. Keywords assigned automatically to the English text in Figure 1, and their relative relevance ("keyness").

2.3 Automatic Recognition of Geographical References

The European Anti-Fraud Office is interested in the automatic recognition of place names in texts. We have therefore developed a tool which scans each text for geographical references and which produces statistics on the relative importance of these in a given text and on the distribution of the place and area names according to countries.

The system is rather simple in that it only checks whether any word or word group in the text is in our extensive database of place names (received from the European statistical office *Eurostat*), without using any pattern matching techniques, but when developing this tool we had to solve several problems which caused difficulties: firstly, some place names are also common words in one or more European languages so that we had to reduce the number of wrong hits as much as possible; secondly, some place and area names have translations in different languages, and thirdly, some geographical references consist of more than one words and variations are common so that we had to recognise these multi-word terms and normalise the variations (e.g. *Frankfurt am Main* can also be written as *Frankfurt a. M., Frankfurt/Main, Frankfurt (Main)*, etc.).

2.4 Automatic Keyword Extraction

The next step in the chain is to lemmatise[3] the corpus and to identify its keywords automatically. We refer to this a *automatic indexing*. We basically index the texts for two reasons. Firstly, the indexing words will eventually be used by the users to quickly get an idea about the contents of the documents and, secondly, we use them in two of the three document clustering and visualisation methods (3.2 and 3.3).

The indexing software we currently use is a customised version of the keyword identification part of *WordSmith Tools*[4] [3]. It indexes individual words if they are more frequent than expected when comparing their text frequency with their frequency in a reference corpus, using the log-likelihood and the χ^2 algorithm for the calculation (see Table 1).

We are currently also working on an alternative method of indexing texts, which consists of assigning keywords from the controlled vocabulary of the multilingual thesaurus *Eurovoc*, which was developed by the European Parliament. This new method has the advantage of producing higher indexing consistency (different types of *bread* would all be indexed as *bakery products*) and, due to the parallel language versions available, of allowing cross-language keyword assignment.

2.5 Further information extraction planned

In the future, we also plan to use tools to extract named entities, dates, references to instances of product groups, subject domains, and other information facets. For the product

[3] For this purpose we use the lemmatiser of Lernout & Hauspie's commercially available software *IntelliScope Search Enhancer*.

[4] See http://www.liv.ac.uk/ ms2928/wordsmith/index.htm

groups, using the multilingual and hierarchical *Common Nomenclature*[5] seems particularly attractive as the hierarchical structure of the taxonomy allows the users to decide on the degree of granularity they are interested in and, as with *Eurovoc*, the results are multilingual.

2.6 Clustering of related words

We carried out experiments in which we clustered the indexing words extracted from the 260 European Parliament Questions and Answers (EP-Q&A) using the hierarchical clustering method described in 3.2 as well as the maps based on these hierarchies, outlined in 3.3. Even though the number of documents and the number of indexing words should have been too small to produce any significant results, several interesting term clusters were created. For technical details, again see sections 3.2 and 3.3.

We intend to use term clusters generated on the basis of larger corpora for two purposes. The first one is interactive query expansion in document retrieval as each word will have several other terms associated to it. Unlike query expansion using hierarchical taxonomies, the term associations reflect up-to-date co-occurrence relations between words.

The second purpose for the word clustering is that we are thinking of using the word clusters as an input for the construction of a thesaurus which will be custom-built for our Commission-internal customers.

3 Visualisation of document collections

When we started the development of the visualisation techniques, the larger document collections were not available yet. Therefore we tested the techniques on the basis of 260 EP-Q&As. The figures in sections 3.2 and 3.3 will, however, not show the *document* clustering results, but rather results of the clustering of *keywords* because these are easier to evaluate by the reader than anonymous document names.

3.1 WEBSOM

The first visualisation technique is a customised version of a neural network approach called WEBSOM. WEBSOM is a method which has been developed by the Neural Network Research Centre of Helsinki University of Technology [1] and which organises document collections and presents them in two-dimensional space. The method owes its name to its original use, which was a successful large-scale implementation of the system using internet (web) documents, on the one hand, and, on the other hand, to the fact that it is based on two successive runs of the connectionist Self Organising Map (SOM) model.

The SOM algorithm is considered one of the most flexible algorithms amongst many that map high-dimensional data such as document vectors onto a much more tractable two-dimensional representation. Loosely speaking, due to the SOM organisation properties, documents which will be attached to the same or to neighbouring cells in the 'document map', will be expected to be "similar". Figure 2 displays the bottom-right region of a document map with 18x18 cells, which was trained on the small set of 260

[5] The European Commission's *Common Nomenclature* (CN) is a classification of goods which was defined by the Council for Customs Cooperation and which is based on the international classification known as the Harmonised System. It includes more than 10 000 eight-digit codes and is in use since 1988 (Official Journal of the European Commission L 238, dated 19 September 1996).

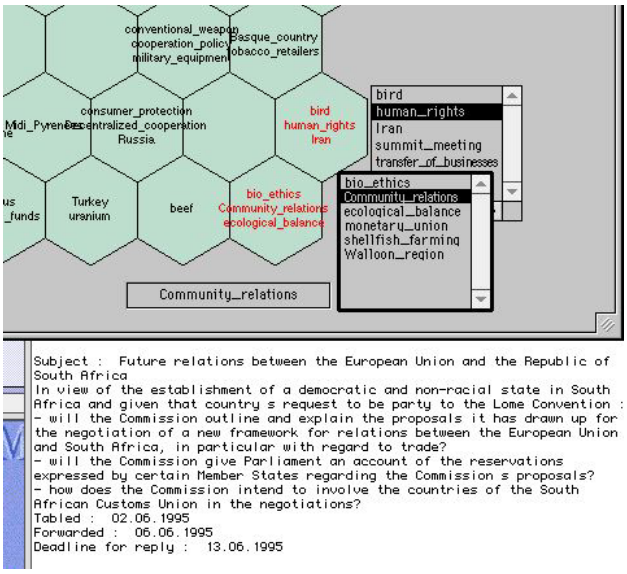

Figure 2. Snapshot of the lower right part of a document map, plus the full text of the *Community relations* document.

documents. Each cell shows a maximum of three document names. Clicking on one cell opens a scrolling list window, and selecting a document name in the list lets its full text be available.

Since WEBSOM uses twice the SOM neural network model, we start the description with an informal presentation of SOM.

Self-Organising Maps. We mentioned that a SOM may be viewed as a device that *projects numerical data from a high-dimensional space onto a two-dimensional one.* The keywords in this very coarse definition call for the following comments:

- *Numerical* input data are required. Input quantities that are not intrinsically associated to numerical feature vectors have to be encoded in some way. For instance, to deal with words or with documents, these will be given a numerical representation (to be discussed later).
- *Two-dimensional* data is easy to visualise. Indeed, the main purpose of most SOM applications is precisely the exploratory data analysis via visual inspection. The representation of SOM models consists of a two-dimensional grid (or lattice), where labels for the input quantities – e.g. words or document names – are displayed. Hexagonal lattices have been used in our implementation of WEBSOM.
- *Projections* are continuous. The SOM algorithm owes its name to the fact that it aims at preserving the topological organisation of the data from the input high-dimensional space onto the output lattice. Accordingly, data labels that are mapped (projected) in one same cell in the SOM grid indicate maximal similarity for their numerical vectors – the feature patterns – in the source space; in like manner, labels that appear in neighbouring grids have closer feature patterns than labels that fall several cells apart on the grid.

The 'Word Map'. At a global level, the input to WEBSOM is a document and its output is the document location over the 'document map' discussed in the next section. Visualising the document collection then amounts to inspect the latter SOM lattice, as for instance in the Figure 2. A SOM has to be fed with numerical input. There is no known method as how to devise meaningful or merely workable feature patterns for whole documents in an automatic way, without making use of domain knowledge or upstream classification procedures. This is one of the reasons why WEBSOM first feeds a SOM model with (numerically coded) words to produce a 'word map', which will eventually be used to generate the desired numerical coding for full documents. In view of the topology-preserving properties of the SOM, a successfully implemented 'word map' achieves the following: broadly speaking, words in the same or in neighbouring cells tend to have similar meanings or contexts or uses, whereas words falling far apart have semantically very little in common. Although we will keep avoiding technical matters, we deem that the following series of comments may shed some light on the 'word map' formation.

- Although it may seem odd that SOM training eventually works, the numerical code attached to each word is purely random. So the coding step is straightforward and does not require any domain knowledge.
- In training mode, triplets of 'consecutive' words are presented to the SOM (the quotes stand to recall that several pre-processing steps may have been carried out: stemming, removing of stop words or of rare words, etc.). The word being learned is the middle one. Yet in terms of numerical importance in the learning algorithm, most weight is given to the contextual terms. This particular approach enables the algorithm to organise the vocabulary in a semantically meaningful way. To foster intuitive understanding, consider the ideal instance where two synonyms would occur throughout the documents with the same ensembles of contextual pairs: since learning is based on the context much more than on the central word itself, the algorithm manages to output the synonyms to one same cell; otherwise it would be hard to imagine a way to associate synonyms without human intervention. Note that in retrieval mode only bare words are required.
- All (triplets of relevant) words from all documents available are normally mapped into one SOM during the first stage.
- The trained SOM accepts as input any code pattern of appropriate length. So if we wish to encode the 'word' ABCDEF and feed it into the model, it will indeed activate an output cell. However, since it is very unlikely that other occurrences of ABCDEF have been met previously during the training phase, the association of ABCDEF to other labels in its output cell does not make any sense. Reasons in the same vein explain why rare words in the training documents should be profitably discarded from the training phase.

The 'Document Map'. The numerical coding which is used for any given document is issued directly from the 'word map'. We present to the 'word map' only the words of the document at hand, which typically constitute a small strict subset of the ensemble of words that have been used to train the 'word map' in the first place. We record the frequency of activation of all 'word map' cells during the document scanning. This bi-dimensional histogram of sorts eventually constitutes (after some additional transformations) the coding or signature of the document at hand.

Again, due to the SOM organisation properties, documents with similar signatures will be attached to the same or to neighbouring cells in the 'document map', and will thus be deemed to be "similar" documents. Looking back at the document coding process described in the previous section, it may be seen that document similarity is grosso modo proportional to the degree of overlap of their signature over the 'word map'. Stated differently – yet still in loose terms – similarity grows with the number of concepts that are

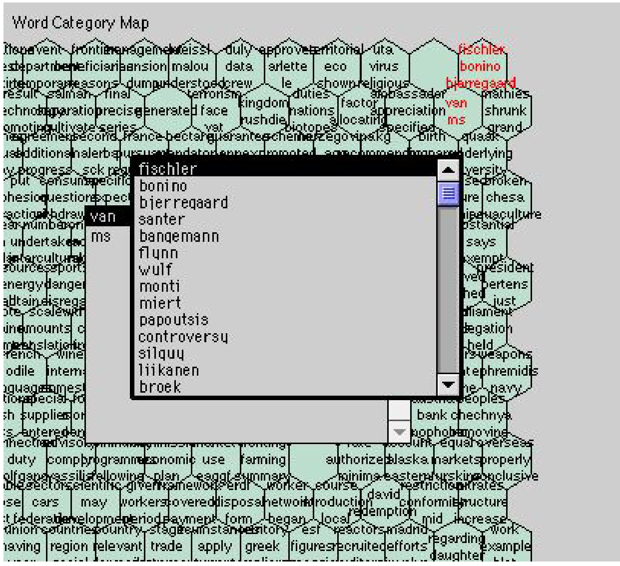

Figure 3. Snapshot of the upper right part of a word map.

common to both documents (as measured by the organisation of words via their contextual frame).

Contrary to the problem evoked in the last bullet in the 'word map' description, a trained 'document map' should cope properly with an unseen document, provided that a fair part of the concepts in the new document also belong to those of the 'word map'. Then the newcomer's signature is meaningful and the document will be attached to other documents sharing similar concepts, as measured by the signatures' overlap. A small amount of unseen words in the new document just act as a little random noise in the coding, which should normally not affect the matching of the signatures.

First WEBSOM tests. This subsection is made of the description of two computer screen snapshots. These snapshots are on Figures 3 and 2, which display a word map and a document map screen, respectively.

The upper-right portion of a trained word map is displayed in Figure 3. Each cell is represented by a hexagon. The vast majority of cells is activated by several words, the three most frequent of which are displayed within the corresponding hexagonal box. Although the map results to be a little bit crowded, this feature readily enables the user to have an overview of the word organisation and possibly to analyse the logic that steers it. For users who would like to perform more detailed investigations on the word map, there is also the possibility to browse the full content of one or more cells online, simply by clicking on them. The selected cells are marked by red characters, and a pop-up window appears which contains all words from the receptive field – ordered according to their frequency – in a scrollable list. The selected upper-right cell contains many European Commission staff names ('Commissioners'), which certainly is a most acceptable clumping. Thinking of it, this somehow impressive grouping could be due – we didn't check – to frequent occurrences of the word 'Commissioner' or the like as left word context and/or 'answers' or synonyms of it as the right word context. Our tentative explanation is insightful as for the kind of organisations that word maps may achieve, and the 'Commissioner' unit rather advocates the efficiency of the "word context" steering principle. The neighbouring cell with 'van' and 'ms' in it is much less satisfactory, inasmuch as these words probably come from fake left contexts like those in 'Van Miert' or 'Ms Bonino', thereby keeping this cell being attached to the 'Commissioner' one. As a matter of fact, words like 'van' and 'ms' are hardly informative and strongly suggest that our word maps are likely to benefit from more accurate data pre-processing procedures. To conclude this section, we recall that the word map merely constitutes an intermediate stage in the WEBSOM computation. Inspection of Figures like Figure 3 is not part of routine operational mode and rather pertains to the development phase.

The upper part of Figure 2 displays the bottom-left region of a trained document map together with the full text of a document. At the present development stage, output document maps are much less crowded than word maps, since we are dealing with only 260 documents which are organised onto a 18x18 map. Document maps will be less sparse when large document collections will be involved.

A quick and dirty labelling of our documents has been carried out by means of the first descriptor in the primary EUROVOC descriptor list. These labels do not purport to be appropriate and are introduced merely to illustrate the on-line use of WEBSOM maps on the computer screen. As for the word map, no more than three document names are displayed within each hexagonal box. Winning frequency is not pertinent here, since any document occurs only once in the training set. Waiting for better criteria, the documents are displayed in alphabetic order. As before, clicking on one cell marks it with red characters and lets a scrolling list window appear. Furthermore, selecting a document name in the list lets its full text be available, as is illustrated in Figure 2 for the 'Community-relations' document.

3.2 Cluster analysis

The other method we have been exploring in parallel is the traditional *hierarchical cluster analysis*. Once the textual data has passed through the modules of *language recognition*, *lemmatisation* and *keyword identification*, we have a table where each document is represented by a set of (possibly weighted) indexing words (see Table 1).

From a set of similar tables we calculate two *proximity matrices* (see e.g. [2], pag. 110–120): the *docXdoc matrix* and the *wordXword matrix*. For the latter matrix the similarity between each pair of indexing words is based on their co-occurrence as descriptors of documents throughout the corpus and also here weights are used, i.e. words occurring very often and/or being widely spread in the collection are treated as less "informative" descriptors and therefore a proportionally lower weight is assigned to them before calculating the proximity matrices.

From the two matrices we create a couple of *dendrograms*, one for the documents and one for the keywords. The algorithm is binary, hierarchical, agglomerative, and uses a treesize-weighted average linkage between the documents/keywords. Except for the proximity calculus, the procedure is virtually the same for both kinds of dendrograms.

Figure 4 shows how 29 of the 321 indexing words in our small sample are organised in the tree diagram. The number at each node is the percentage of the maximum similarity encountered between any two items in the data set; the three columns to the right show the node number, the number of words contained under each node, and the number of occurrences of each word, respectively. In the document dendrogram, a ranking list of the most representative indexing terms is given instead of this last number. Some "upper" (here: rightmost) parts of the tree in Figure 4 are dimmed. The reason for this will be given below.

```
==================================================
bovine·········\                          34  1  4
               83····\                   338  2
bse···········/     |                     36  1  6
                     54-\                415  5
encephalopathy·\     |  |                105  1  3
               87·\  |  |                335  2
spongiform·····/  |  |  |                271  1  4
                  58·/  |                400  3
consumer··········/     |                 68  1  2
                        28=\             547 10
infect·········\        |  |             153  1  3
               83·\     |  |             344  2
scrapie········/  |     |  |             260  1  2
                  64·\  |  |             383  4
scientific·····\  |  |  |  |             259  1  4
               68·/  |  |  |             375  2
veterinary·····/     |  |  |             315  1  5
                     34-/  |             526  5
committee············/     |              55  1  6
                           10=\          588 13
intend············\        |  |          159  1  4
                  33·······/  |          530  3
labelling······\  |           |          173  1  2
               42·/           |          493  2
transparency···/              |          299  1  3
                              |8=\       596 17
disease········\              |  |        89  1  3
               83·\           |  |       341  2
orphan·········/  |           |  |       212  1  2
                  62·\        |  |       389  3
medicine··········/  |        |  |       189  1  5
                     53·······/  |       418  4
product··············/           |       226  1 11
                                 |2====\ 627 29
antibiotics····\                 |     |  15  1  2
               50·\              |     | 428  2
human··········/  |              |     | 148  1  2
                  46····\        |     _ 466  4
additive·······\  |     |        |         3  1  2
               83·/     |        |       337  2
feed···········/        |        |       123  1  3
                        22-\     |       557  9
detect·········\        |  |     |        86  1  2
               75····\  |  |     |       356  2
report·········/     |  |  |     |       249  1  4
                     35-/  |     |       523  5
residue········\     |     |     |       253  1  2
               75·\  |     |     |       362  2
use············/  |  |     |     |       311  1  4
                  40·/     |     |       498  3
animal············/        |     |        12  1  5
                           |3====/       620 12
limburg········\           |             180  1  2
               50·\        |             451  2
protection·····/  |        |             232  1  2
                  38·······/             517  3
netherlands·······/                      200  1  4
==================================================
```

Figure 4. Part of the dendrogram showing how the automatically extracted indexing terms in our sample cluster. This part is a subtree containing 29 (i.e. 9%) of the 321 indexing terms used.

3.3 Document maps

In order to visualise in a more compact way how the documents or words relate to each other, we map them onto a two-dimensional grid. The procedure is to first cut it up into nine subtrees and then to distribute these within a 3x3 grid in such a way that more related subtrees come closer to each other. Cutting up the tree into N subtrees the way illustrated by Figure 4, i.e. starting from the root and dissolving the weakest nodes until there are N subtrees, guarantees that the most stable subtrees/clusters are kept intact the longest. The cut-away branches of the tree are dimmed in Figure 4.

Figure 5 shows how the nine subtrees of the word dendrogram in our sample are distributed to optimally reflect the similarities between them. The subtree of Figure 4 here falls in the cell in the lower left corner. In the next step we continue this subdivision and create nine subtrees out of each of these nine which we have already. In one case, however, (the cell/subtree in the upper right corner) there are not enough words to do that, so we leave it as it is[6]. Figure 6 shows the order in which this subdivision is carried out subtree-wise, and 7 shows that when calculating the best cell for each "subsubtree", the other eight subtree-internal and – if available – the sixteen closest surrounding subsubtree-external cells/subsubtrees are considered.The result of this is shown by Figure 8 where we also indicated the similarity between each neighbouring subtree pairs by means of thickness of the line separating the two cells: the thinner, the more similar. Again, the

[6] Currently, though, we are experimenting on slicing up those cells in ≤ 8 stripes

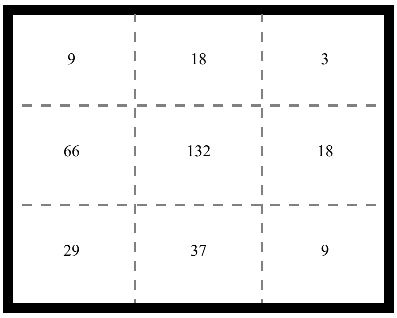

Figure 5. A 3x3 grid where each cell contains one of nine subtrees into which the dendrogram has been cut up. The distribution reflects optimally the inter-subtree relations and the numbers refer to the size of each subtree in terms of words.

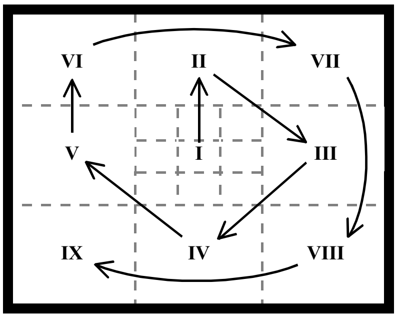

Figure 6. The cells of the 3x3 grid of Figure 5 are collapsed each into another 3x3 grid (where possible), starting with the central cell and proceeding as indicated by numbers and arrows.

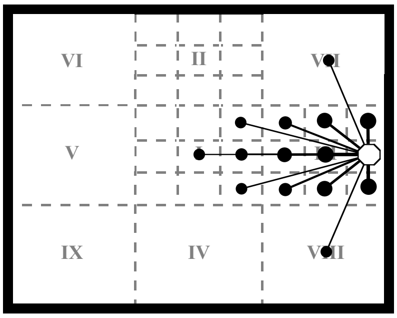

Figure 7. When deciding which cell best suites a given subtree, other internal and external cells and their already assigned subtrees are related to.

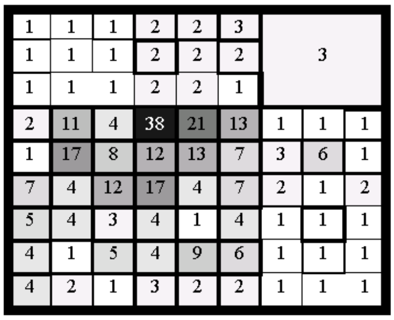

Figure 8. Each cell of the 3x3 grid of Figure 5 has here been itself subdivided into a 3x3 grid where this was possible. We note that most of these smaller subtrees fall into the left-central area.

number of words contained in the subtrees/cells is indicated, this time additionally by grey-shading.

Figure 9 is a variant of Figure 8 without the number indicating cell size. We notice that the strongest correlating word(group)s occur with a low frequency in this data set as the "walls" between their cells are thin or even non-existent. Looking at this map a scale drawing of a house comes to one's mind. Figure 9 is shaded except for the parts occupied by the subtree of Figure 4. Figure 10 zooms in on this non-shaded part and shows the words contained in its cells.

Of course, as soon as the document sample size grows beyond almost trivially small sizes, this 9x9 grid must be collapsed further (e.g. re-iteratively into an 81x81 grid) and/or the cells made clickable to have their underlying subtree pop up in a session of data exploration or information retrieval.

4 Final remarks

To conclude, we would like to make some remarks regarding our goal to develop a multilingual system and regarding the evaluation of the visualisation results.

4.1 Multilinguality of the system

One aspect of the system which clearly needs a lot of attention in the European context is multilinguality. In the system we are working on, the visualisation of document collections

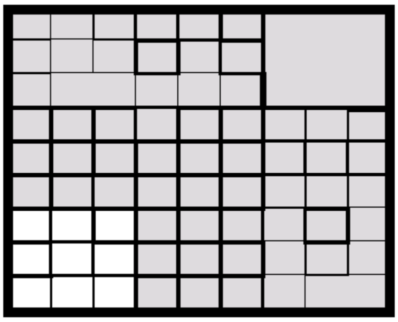

Figure 9. Same as Fig. 8, without density indicators and with all subtrees shaded except the one shown in detail by Figures 4 and 10.

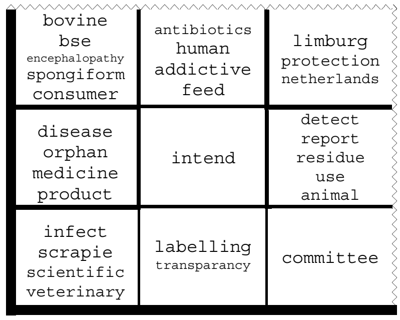

Figure 10. Zoom in of lower left corner of Figure 9. An optimized 2D re-arrangement of the subtrees shown in Figure 4.

depends on a uniform representation of the documents of different languages, e.g. by using the same keywords, codes, etc. for all covered languages. This is the main reason why we have to focus on techniques for multilingual indexing, subject domain recognition, and information extraction (see the related comments in 2.4 and 2.5).

According to our current plan, the document visualisation techniques presented in sections 3.2 and 3.3 will not only use keywords, but also subject domains, geographical references and names. While the representation of names and geographical references are easily standardised for all languages, we have to make a special effort to produce a language-neutral representation of subject domains, keywords, product groups, etc. Our approach for doing this is to link the texts to multilingual thesauri such as *Eurovoc* and the *Combined Nomenclature* (see sections 2.4 and 2.5). It is thus our goal to extract as much information as possible, reformulate it in a language-neutral way and to use this information for the clustering and visualisation.

4.2 A note on the evaluation of the visualisation of the document collections

So far, the JRC has developed a number of tools which have not yet been integrated. Some of these tools have not been evaluated fully and are still under development. Furthermore, the evaluation procedure is not always straightforward. Especially complex procedures such as the visualisation of document collections are a non-trivial task as there is no automatic way of evaluating the performance. There is not even an agreed standard with which to compare our results. It is rather the case that only the customers will eventually decide how useful each individual visualisation technique is.

References

1. T. Kohonen and S. Kaski and K. Lagus and T. Honkela. *Very large two-level SOM for the browsing of newsgroups.* Proc. of ICANN'98, 1996. Springer, Berlin.
2. Michael P. Oakes. *Statistics for Corpus Linguistics.* Edinburgh Textbooks in Empirical Linguistics. Edinburgh University Press, Edinburgh, 1998.
3. Mike Scott. *WordSmith Tools v. 3.0.* Oxford University Press, Oxford, UK, 1999.