How to handle multiple expertise from several experts: a general text clustering approach

Stéphane Lapalut
ACACIA project, INRIA Sophia Antipolis,
BP 93, 06 902 Sophia Antipolis cedex, France
Stéphane.Lapalut@sophia.inria.fr

Abstract

At the earlier stage of the knowledge acquisition process, interviews of experts produce a large amount of rich but ill-structured texts. Knowledge engineers need some tool to help them in the exploitation of all these texts, especially when dealing with expertise from different experts, in different but overlapping domains. We propose the use of a statistical method, the top-down hierarchical classification and a new interpretation of its results. The initial statistical analysis proposed by M. Reinert [19, 20] gives two kinds of results: first a segmentation of texts that reflects their "semantic contexts", which we use to raise structures of texts and inter-relations between them, and second, classes of significant terms belonging to these contexts, which can be related to the experts or to their specialities. In this paper, we briefly describe the method, then we explain the exploitation of its result on a multiple text corpus case study, relating interviews of road safety experts. We conclude with some research directions to deal with so-called "ontologies" on expert's domains.

1 Introduction

In case of a domain without established theory, such as complex accident analysis, the only way to obtain a significant and useful amount of data is to observe and interview experts working on various selected cases [5, 4]. This produces ill-structured text interviews. When reading all these interviews, the knowledge engineer lacks guidelines to model the domain, to distinguish and characterize the different approaches of the experts and to produce useful knowledge bases.

Given a huge corpus of experts' interviews, we propose the use of a statistical method called "top-down hierarchical classification", to handle both self contained texts and sets of chosen texts. It detects groups of terms of the corpus (a set of one or more texts) strongly distinguishable, according to the statistical occurrence of meaningful terms or pairs of them in small text units, such as sentences. These groups are called classes and have been identified as relevant semantic contexts [1, 19]. They lead to a partition of the corpus that reflects its structure. After interpreting each class with the help of the related terms, the knowledge engineer knows the subject of each part of the structure of the corpus. We experimentaly find out that in the case of a mutliple text corpus, parts of the raised structure are correlated to experts or specialities. With these results, the knowledge engineer is able to draw differences and similarities between experts' knowledges. He can determine sub-domains and select parts of experts' interviews relevant to his purpose, and focus his work.

We briefly recall in the next section the method fully described in [16], which we have used to raise structure of single text corpora. In the third section, we show the results of a multiple text from several experts corpus analyse. In the fourth section, we put back our work in the field of knowledge acquisition from several experts. Then we conclude in the last section with research directions to carry on the method one step further.

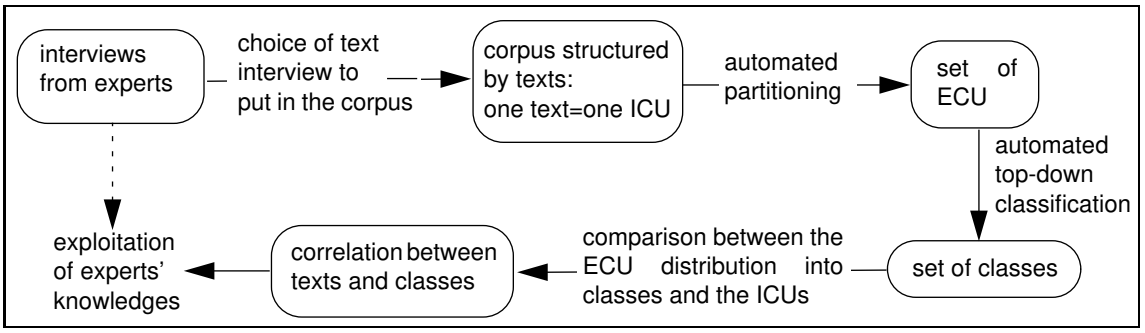


Figure 1: Our extension of the method towards knowledge acquisition process.

2 The top-down hierarchical classification method and its extension

The initial objective addressed was to discover semantic contexts characterized by groups of terms, from a given corpus, which we call classes. A principle derived from the Huyghens decomposition formula is used: "to find a partition of a set that minimizes the intra-class variance, it is sufficient to find dichotomies that maximize the inter-class variance".

This section give a taste of the method that has been developed by Max Reinert since 1984 and implemented in a tool called ALCESTE [20]. This tool presently processes only French texts, even if an English extension is coming up. The current version is a commercial one. The purpose of Reinert was to use semantics contexts to help psychologists in their analysis and research of models. Given the statistical analysis results, the semantic contexts and the way they are represented in the corpus, we have found text clustering as another application, with a useful interpretation for the knowledge acquisition purposes.

2.1 Our extension of the initial method

We refine the initial cycle to adapt it for our purpose of discovering the structure of a corpus. We have found a correlation between natural articulations of the corpus and the structure given by the classes to this corpus. We can sketch our method in figure 1. The classes give a structure that is refined by the user with the help of class interpretations. Formally, we can consider the corpus processing as the research of correspondences between sets from a triplet $\{U_1, S, TU_1\}$ where:

U_1 is the elementary segmentation of the corpus (the set of elementary context units (ECU)),

TU_1 the objective clustering of U_1 into classes (found by the top-down classification) and

S the implicit organization of the corpus according to the interrelated topics in experts' interview: this organization is to be discovered.

The whole analysis permits the clarification of S, which is the correspondence between the corpus viewed as a sequence of ECU, U_1, and semantics contexts defined by classes in TU_1.

2.2 Input data

In this section we highlight the data format and the way they are extracted from the initial corpus. ALCESTE takes two inputs: the corpus, composed by several texts, and a set of parameters, called the analysis plan, which the user gives to conduct the analysis. The corpus is organized by the user as a sequence of ICU, mainly one ICU per text in the case of texts from several experts. The whole analysis is driven by an analysis plan, seted up by the knowledge engineer.

Before the main algorithm processing, the statistical analysis called "top-down hierarchical classification", a morphological reduction of the terms of the corpus is performed. Two lists are

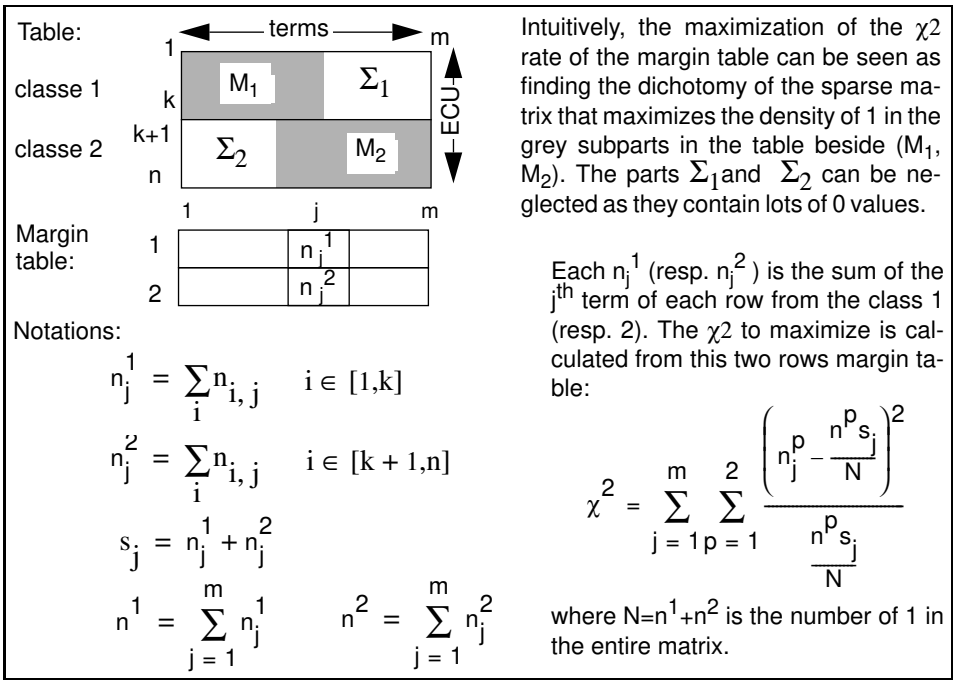


Figure 2: Principles of the sparse matrix dichotomy using χ^2 distance.

extracted from the corpus. One list contains all the words from the corpus in alphabetical order, and the other list contains the sequence of ECU that composed the corpus in the order they appear in the corpus. Some words, called toolwords (noisy words), such as prepositions or pronouns are recognized and typed according to dictionaries. Other words, such as nouns, verbs, adjectives, are considered as meaningful terms and called plainwords (non-noisy words) and reduced to a primitive form. An ECU is a word sequence that integrates a fixed number of plainwords, as specified by the user in the analysis plan. ECU useful length ranges from 10 to 20 plain-words. ECU can be sketched as sentences from the corpus. The ECU segmentation is performed according to the punctuation with a priority order of the signs (. > ? > ! > ; > : > , > space). The fixed length is needed to validate the statistical algorithm and it is not a strong constraint for the purposed context identification. The analysis plan we use allows a double classification with ECU of two different lengths. It permits an adjustment of ECU length to obtain a better classification. The cross between the two hierarchies of classes found determines a stable classification, according to a χ^2 criteria maximisation.

The program uses the two above lists, ECU and plainwords (terms), as rows and columns of a double entry boolean table. The presence of a term in ECU is noted with 1 and its absence with 0 (figure 2). The complete table, which is a sparse matrix, is used as the input of the main algorithm. We describe it in the next section.

2.3 The Algorithm

The applicability of the underlying statistical theory used by this algorithm was proven in [1, 15, 19]. Given the boolean table, which use terms as column entries and ECU as row entries ($m_{ij} = 1$ means that the i^{th} term belongs to the j^{th} ECU), we define classes as particular sets of rows; a given row belongs to a single class. The final set of classes is an incomplete partition of the initial set of ECU. To distinguish classes, the algorithm uses a χ^2 distance between the margins of two given sub-tables, determined by successive dichotomy as describes in figure 2. A margin is a row vector formed by the sum of all values in each column. At each step of the algorithm, the dichotomy that maximizes the χ^2 rate of the margin tables is found. In a simpler form, it can be

stated as:

1. *First step:* find the dichotomy that maximizes the χ^2 association rate between the margins of the two determined classes;

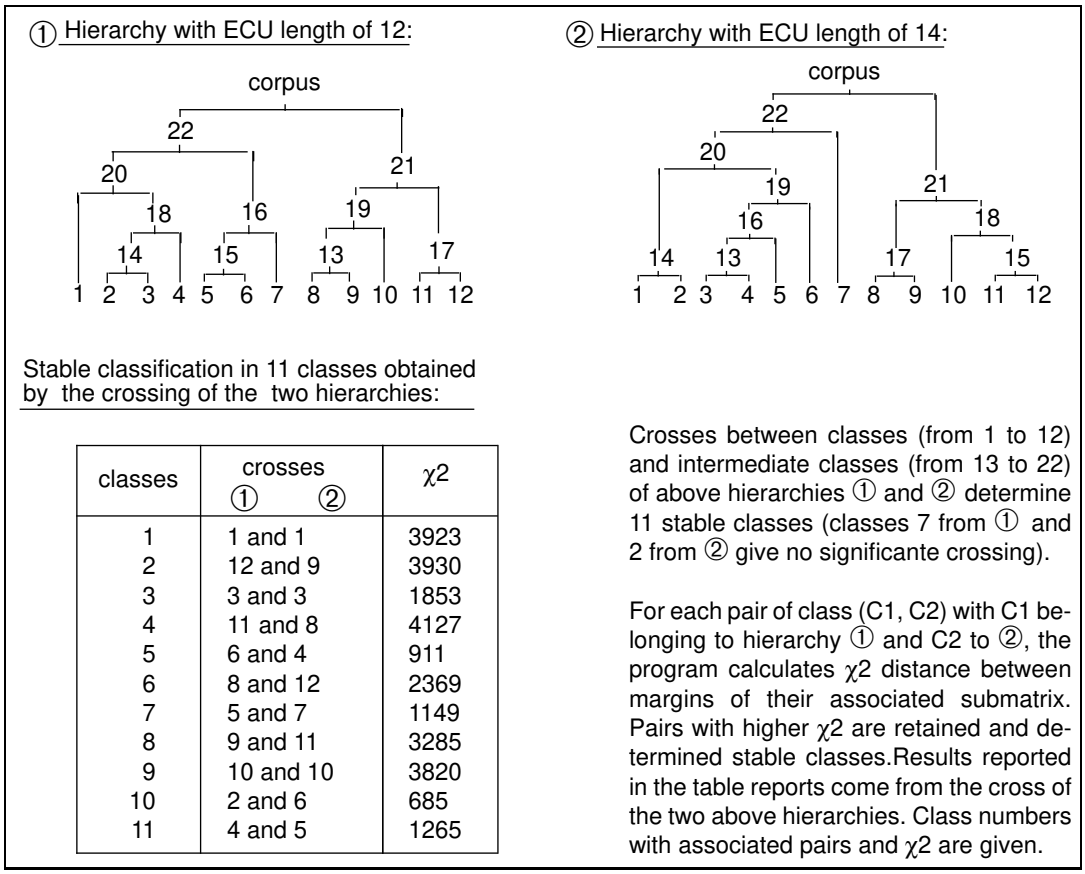
2. *Other steps:* until the maximum number of classes allowed is reached, do:

   1 pick the class with the greatest number of rows,

   2 among all row combinations, find the dichotomy that splits this class into two subtables and maximizes the χ^2 of their margins,

   3 replace the picked class with the two found classes.

The length of ECU in the corpus determines the quality of the found hierarchy of classes. The quality of a hierarchy refers to the partitioning of the corpus and to the related sets of terms that defined each class. The higher the ratio of terms that mostly belongs to the same class (occurrences of a term in the ECU of the same class against the total number of its occurrences in all ECU), the better the classification. To improve this ratio and obtain a good hierarchy, we have to choose the appropriate length for ECU. The tool gives a percentage of terms that mostly appear in a single class to estimate the classification correctness, the average ranges from 50% to 70%.



classes	crosses ① ②	χ^2
1	1 and 1	3923
2	12 and 9	3930
3	3 and 3	1853
4	11 and 8	4127
5	6 and 4	911
6	8 and 12	2369
7	5 and 7	1149
8	9 and 11	3285
9	10 and 10	3820
10	2 and 6	685
11	4 and 5	1265

Figure 3: Example of stable classification resulting from crossing between two hierarchies of classes, obtained from a ten text corpus, with two different ECU lengths (correctness of 54,08%).

2.4 Statistical analysis outputs

The first result we exploit is the double classification and the selected stable classes. A ten text corpus gives the hierarchies in figure 3. In this example, the analysis plan asks for 12 classes and

the comparison between the two classifications enlightens 11 stable classes. For each of them, the number of correlated ECU is given with χ^2 rates. The rate of correctness of 54% empiricaly validates the choice of the ECU lengths made (12 and 14 plain-words per ECU).

To be useful, a meaning must be attached to each class. This work is the burden of the knowledge engineer, as the classification is not supervised nor guided with a pre-determined lexicon. Each class is characterized by the list of its ECU and four files of correlated terms. These files permit the class labelling. For each class, they consist in two lists of terms and two lists of couples of terms. For both couples of lists, one list concerns the best correlated terms to the class (profile) and the other the less correlated terms (anti-profile). An excerpt from a term profile is shown in figure 4.

As few words are preponderant in profiles for each class, different readers are led to the same semantics context interpretations. These interpretations require some domain knowledge. For each class, the user cross-checks the correctness of his four interpretations, one per files of characterizing terms. In most cases, no contradiction occurs and one expression results to characterize each class. These class tags are used to help the interpretation of the correlation between initial texts from the corpus and classes. Typically, classes are revelvant to an expert speciality or a sub-domain. The outcome of our method arises in this last stage, the structure refinement. The set of preponderant ECU attached to the classes permits the clustering of the initial set of ECU, e.g. the corpus. With the help of a graphic distribution of ECU from the corpus for each class according to their χ^2 association rate, the codification enabling us to locate the small corpus parts (less than a page) that contain articulations of the implicit structure of the corpus. This stage is detailed in [16] with examples.

The algorithm is blind with respect to the semantics of the word correspondences it makes. It only gives us an abstract result of all word associations that a reader can outline from the text. Of course, these associations reflect a feature of the natural language, called semantic context. We describe here the way to interpret the analysis results and relevant informations given about the corpus.

Our method addresses two goals, firstly to retrieve the natural articulations and the structure of the corpus, which is the main goal when the corpus contains only one text, and secondly to have qualitative informations about differences and similarities between components of the corpus, especially in the case of corpus composed of texts that come from several experts. We detailled a multiple text corpus study in the next section.

3 Multiple texts and multiple experts

In this case, the main goal is not to discover the structure of the corpus, but to obtain some useful informations to compare several texts. As each text characterizes one or several experts working together, this sort of analysis helps to establish experts' knowledge and reasoning differences. Several parameters occur at the same time and many combinations between texts are relevant; we have not investigated all the possible ways. We have especially studied the case of a single text corpus and a multiple text corpus that includes this text. We present here the kind of results we obtained for a multiple text corpus analysis.

3.1 A case study

To test this kind of analysis, we put ten experts' interview texts (about 15000 lines) within the same corpus, which gives around 10000 ECU (ECU length of 12 and 14). The obtained classes presented in figure 5 lead to a corpus structure very close to the existing ICU partition, e.g. the initial texts. Precisely the program points out 11 classes. Each of them is directly relevant for one or two ICU:

(a) 5 ICU are relevant for a single class (ICU 2, 5, 7, 9 and 10),

(b) two ICU for two classes (ICU 1 and 4),

List of terms correlated to the class 7 in the previous classification, with χ2 superior to the average χ2 (62.96):

①	②	③	④	⑤		⑥
409	50.	75.	66.67	1057.00	X	mesur+
72	25.	30.	83.33	667.29		grip+
122	20.	25.	80.00	510.39		characteris+
88	17.	20.	85.00	462.62		equipment+
57	13.	13.	100.00	420.22	0	join+
134	24.	51.	47.06	342.56		road+
593	12.	14.	85.71	329.12		surface+
121	5.	7.	71.43	112.47		countr+
247	5.	8.	62.50	97.20		light+
368	7.	19.	36.84	74.82		itinerar+
475	6.	15.	40.00	70.56		tyre+
934 *	91.	448.	20.31	503.27	*	*fra-ent
945 *	91.	996.	9.14	158.69	*	*F
944 *	150.	2653.	5.65	130.33	*	*ENT

The tagging of this class has been stated as: "interview of Fra dealing with grip of tyre on roads".

Columns in the above table mean:

① the place number in the dictionary of terms built from the corpus

② the number of occurrences in the ECU of the class

③ the number of occurrences in the whole ECU

④ ratio of ECU in the corpus in which the term appears

⑤ χ2 association rate between the term and the class

⑥ the term in its reduced form with an optional mark

Figure 4: Example of a class tags, according to its term *profile*.

(c) one ICU for three classes (ICU 6),

(d) and the last two ICU (3 and 8) are not directly characterized by a class, but they are weakly related to two classes.

In the case (a), we have verified that the semantic contexts described by the classes are in accordance with the related texts. Case (b) leads to the same conclusion. The two classes related to ICU 1 and 4 correspond to the same domain with a slight point of view difference. Case (c) is a very interesting one and we investigate it in further details below. Anyway, classes related to ICU 6 are in accordance with an analysis of that text we have made before. At that time, we have no well-established explanation for the case (d). So, we already can conclude that this kind of analysis makes precise similitudes and differences between the texts entered as ICU in the corpus. For each of them we obtained a synthetic point of view of their main subject (cases (a) and (b)).

3.2 Class interpretation

The huge number of terms and ECU contained in this sort of multiple text corpus allows a very contrasted class characterization. χ^2 rates are higher than in the case of single text that makes the interpretation of classes easier. For each class, a small set of terms are preponderant. But each of them is very precise and specific, such as the class 8: "the accident factors for the case 003, central lane and indicator ambiguity". This class is relevant for the text 9 and its characterization is exactly the conclusion of a manual analysis we made before.

Many classes strongly characterize experts' specific vocabulary and skills. So, we obtain specific terms for these experts from these classes. The main problem is to obtain the shared terms and to identify the way each expert uses them. We suppose that a link exists between this shared vocabulary and the only unexplained phenomena about the weakly characterized texts 3 and 8 (case (d) above).

3.3 Conclusion

From the analysis of multiple text corpora we conducted, both inter-text and intra-text comparisons are possible. Above a given number of different kinds of text, e.g. ICU in the corpus, the overall text differences are given back by classes. Texts strongly characterized by specific subjects lead to peculiar classes. Side effect phenomena reveal some specific features of very precise parts of texts. So, according to the way the classification is carried out, different grains and structure levels can be reached. The main remaining problem is to put relevant texts together within a corpus to highlight a level of detail or some desired differences between texts and so, expert's knowledge particularities. We think that our approach complets the conceptual one, as proposed in [6, 4], especially to identify similarities and differences between experts' vocabularies. It can be used as a concurrent approach, as well as a preprocess of the knowledge modelisation phase.

	classes	*yve-int	*pie-int	*man-int	*jlo-int	*fra-int	*dan-jlo-24	*dan-man-002	*dom-fra-24	*man-pie-jlo-003	*man-solo-003
interviews, Yve's detailed accident studies (DAS)	1	X	~	~							
duo, impacts, traces and car deformations	2						**X**	X	~		~
*pie-int ($\chi 2$ of 1332 for an average of 32.15)	3		X								
crash, kinematics reconstitution	4				X		**X**				
road infrastructure and its users	5			~		~					
perception of infrastructure by the drivers	6	~									
the grip tyres on the roadways	7					X					
accident factors, central lane and indicator	8									X	X
infrastructure and approach before the crash	9						**X**	~	~		
the whole accident analysis, data and indices	10				X						
the accident situation	11	X									
class interpretations		1	2	3	4	5	6	7	8	9	10

*<label> names each experts' interviews.
X states a strong correlation between the class and the text (more than half of the ECU that characterized the class belong to these texts).
~ states a weak but existing correlation between the class and the text.

Figure 5: Correlation between texts and classes from a multiple text corpus.

4 Related Work

Among the knowledge acquisition community, few works deal with the practical exploitation of interviews from several experts. Most papers deal with the conceptual and methodological aspects, such as [5, 4, 10, 8, 22]. One key issue in these papers is the conflict detection and management between experts' knowledges. To our knowledge, there is no approach that tackle the problem at the earlier stage of the knowledge acquisition process as we do. One another originality of our work is the handling of the statistical analysis.

Statistical approach is far from being a new one in text processing. It has been widely used in text categorization [12, 18], as well as in language analysis [23, 3]. Some applications have been made to the knowledge acquisition field. Statistical methods are mostly used to strengthen a conceptual method [18, 7, 13]. Tools such as NLDB [13] and SKIS [18] combine both conceptual and statistical approaches within a hybrid system. Most applications tend to provide a set of tools such as NLDB, which is a "set of statistical method", or to offer an assistant to a human expert, like INLEN [14] that "performed a sophisticated data analysis", KITTEN [21] where "the knowledge acquired is being feedback to facilitate the intelligence of people", MOCA [7] that "helps analysts to cope with large quantities of intelligence data" and SADC [17]. A few of them pretend to perform automated analysis, such as NLDB and SKIS, but they require the user to build lexicons (NLDB) or well defined sets of criteria (keyword $\leftrightarrow$ category correspondences in SKIS) adapted to the knowledge engineer purpose and to the considered domain.

Texts in natural language format serve as entries in KITTEN, NLDB, SADC and SKIS. Corpora are used in various ways, from "a set of entities that are sentences whose features are words they contain" (IMS project, [9]) to sequences of keywords (SKIS). The ALCESTE segmentation method is close to the one used in KITTEN with the constraint of the ECU fixed length. Statistical treatments mostly use term weighing. Some algorithms, such as TEXAN in KITTEN or the "similarity measuring component" in SKIS implement a distance measurement too. The only algorithm with a pure distance measurement is TEXAN. But it is a "simple distance-in-text measure" that lacks to take into account the whole corpus. Crossed entries of non-noisy words and ECU in the matrix of the presented top-down hierarchical classification allow a measurement based on a χ^2 distance. It meets the requirements proposed in MOCA, as "it does not need to input the number of clusters desired", it allows "overlapping cluster" on non-noisy words and it analyses the corpus as a whole to produce clusters.

The algorithm we presented in this paper performs an overall analysis and a correspondence between text units. We think it is more suitable for ill-structured text, such as interviews, than Hearst's texttiling algorithm [11]. With our method, we obtain more than the overall corpus structure. For instance, detailed indications about term distribution and correlation enable us to tag the structure and can be used later for other purposes, such as ontology building.

To conclude, the presented method and algorithm meet previous work. They are candidate to be a "more sophisticated text analysis technique" [21], but till now with a human assistance. The corpus organization (ICU), its segmentation (ECU) and the clustering algorithm (cf. section 2.3) contribute to enhance and refine the set of statistical tools and methods commonly used; then the overall extension reaches a useful feature to help the knowledge acquisition process, especially in the global treament of multiple texts from several experts. We detail further possible extensions in the next section.

5 Conclusion and Further Research

In this paper we have detailed the basis and the results of a method to enlighten both the structures of corpora and the subjects of their parts. We have shown on a ten texts corpus how it can be use to deal with multiple interviews from several experts. It help the knowledge engineer to identify specific vocabulary, terminologies of domains and subdomains. The tool already allows the research of terms related to a given one, without class consideration. Combined with the multiple text corpus analysis, we hope to find a way to help knowledge engineers to establish

ontologies from a given domain or expert. The use of decision tree algorithms as in CART [7] can be a way to help exploitation of the big amount of produced data. Anyway, the knowledge engineer must have to read analysed texts and to understand them. Our method results in a spare of time and facilitates management of a huge amount of expertise texts.

So, the use of term correspondence analysis by the means of the top-down hierarchical classification provides useful results to help the management of roughly structured texts. Implicit structures of single texts and differences inside a set of texts can be discovered. It guides the reader in his work, which becomes quicker and safer than reading them without help. The method to exploit multiple text corpus gives lot of way to the knowledge engineer to deal with multi-expertise. One interesting way to investigate concerns the characterization of domain terminologies, according to a given expert or not. A work in this direction has been made by D. Bourigault with the help of the tool LEXTER [2], a terminology extraction software. It mainly produces an hypertext version of the discovered domain ontology. As it uses a morpho-syntactical analysis to build a grammatical network of term relevant to the domain, it is a conceptual approach, opposed to the statistic one. Each approach can be helpfull according to the kind of corpus the knowledge engineer have to deals with.

Acknowledgments

All examples come from a research supported by the "Ministère de la Recherche et de l'Espace" under contract n° 92 C 0757 and the "Ministère de l'Equipement, des Transports et du Tourisme" under contract n° 93.0003. Special thanks to Max Reinert and Krystel Amerge who lead me to the text statistical analysis, especially with the help of the ALCESTE program. Thanks to Rose Dieng and Olivier Corby who have encouraged me in this work. Last but not least, thanks to the experts of INRETS, on the expertise of whom our analysis relied.

References

[1] J. Benzecri. *L'analyse des Données.* Dunod, Paris, 1973.

[2] D. Bourigault. Lexter, a terminology extraction software for knowledge acquisition from texts. In *Proceedings of the 9th Knowledge Acquisition for Knowledge Based Systems Workshop*, Banff, Canada, 1995.

[3] E. Charniak. *Statistical language learning.* Bradford books. The MIT Press, Cambridge, Mass., 1993.

[4] R. Dieng. Conflict management in knowledge acquisition. *International Journal of Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 9(4):337–351, 1995.

[5] R. Dieng, O. Corby, and S. Labidi. Expertise conflicts in knowledge acquisition. In B. Gaines and M. Musen, editors, *Proceedings of the 8th Knowledge Acquisition for Knowledge-Based Systems Workshop*, volume 2, pages 23.1–23.19, Banff, Canada, Feb. 1994.

[6] R. Dieng, S. Labidi, S. Lapalut, and P. Martin. Comparaison de graphes conceptuels dans le cadre de l'acquisition des connaissances à partir de multiples experts. In *Journées Graphes Conceptuels*, LIRMM, Montpellier, 1992.

[7] S. Fall, T. Crawford, S. Souders, and M. Rabin. Automated knowledge acquisition technics for intelligence analysts. *Applications of Artificial Intelligence*, 1095 of SPIE:66–77, 1989.

[8] B. Gaines and M. Shaw. Comparing the conceptual systems experts. In *Proceedings of the 9th International Join Conference on Artificial Intelligence*, pages 633–638, Detroit, 1989.

[9] B. Gaines and M. Shaw. Using knowledge acquisition and representation tools to support scientific communities. In *Proceedings of the Twelve National Conference on Artificial Intelligence*, volume 1, pages 707–712. AAAI, AAAI Press/The MIT Press, 1994.

[10] N. Glaser, M.-C. Haton, and J.-P. Haton. Models and knowledge acquisition cycles for multi-agent systems. In B. Gaines and M. Mussen, editors, *Proceedings of the 9th Knowledge Acquisition for Knowledge-Based Systems Workshop*, volume 2, pages 40.1–40.20, Banff, Canada, Feb. 1995.

[11] M. Hearst. Multi-paragraph segmentation of expository text. In *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistic*, Las Cruces, NM, June 1994.

[12] P. Jacobs. Joining statistics with NLP for text categorization. In *Third Conference on Applied Natural Language Processing*, pages 178–85, Morristown, NJ, USA, Apr. 1992. Assoc. Comput. Linguistics.

[13] P. Jacobs. Using statistical methods to improve knowledge-based news categorization. *IEEE Expert*, 8(2):13–23, Apr. 1993.

[14] K. Kaufman, R. Michalsky, and L. Kershberg. Knowledge extraction from database: design principles of the INLEN system. In *Proceedings of the 6th International Symposium on Methodology for Intelligent Systems*, number 542 in LNCS, pages 152–161, Berlin, 1991. Springer-Verlag.

[15] M. Kendall and A. Stuart. *Inference and Relationship*, volume 2 of *The advanced Theory of Statistics*. Charles Griffin and Co Ltd, 1979.

[16] S. Lapalut. Text clustering to support knowledge acquisition from documents. Technical Report RR-2639, INRIA U.R. de Sophia Antipolis, BP 93, 06902 Sophia Antipolis Cedex, 1995. ftp://ftp.inria.fr/INRIA/publication/RR/RR-2639.ps.gz.

[17] B. Moulin and D. Rousseau. Automated knowledge acquisition from regulatory texts. *IEEE Expert*, 7(5):27–35, Oct. 1992.

[18] M. Register and N. Kannan. A hybrid architecture for text classification. In *Fourth International conference on Tools with Artificial Intelligence, TAI'92*, pages 286–92, Arlington, VA, USA, 1992. IEEE Compu. Soc. Press.

[19] M. Reinert. *Classification descendante hiérarchique pour l'analyse de contenu et traitement statistique de corpus*. PhD thesis, Université Pierre et Marie Curie, Paris 6, Paris, 1979.

[20] M. Reinert. *Notice du logiciel ALCESTE, version 2.0*, 1992.

[21] M. Shaw and B. Gaines. Kitten: Knowledge initiation and transfert tools for experts and novices. *Int. J. Man-Machine Studies*, 27:251–280, 1987.

[22] M. Shaw and B. Gaines. A methodology for recognizing conflict, correspondence, consensus and contrast in knowledge acquisition systems. *Knowledge Acquisition*, 1(4):341–363, Dec. 1989.

[23] Z. Wu, L. Hsu, and C. Tan. A survey on statistical approaches to natural language processing. technical report TRA4/92, Departement of Information Systems and Computer Science, National University of Singapore, Kent Ridge, Singapore 0511, Apr. 1992.