TRIGRAM CODING IN THE FAMILY EXPENDITURE SURVEY OF THE CBS

Martje Roessingh and Jelke Bethlehem
Netherlands Central Bureau of Statistics, The Netherlands

1. Introduction

Coding of answers to open questions is an important part of processing survey data. Coding can vary from the entry of simple sequence numbers of answer categories to the assignment of multiple digit codes corresponding to complex hierarchical classifications. An example of simple coding is attaching an area code to a municipality. Complex codings are, for example, classifications of purchased goods, occupations, or industrial activities.

Coding has traditionally been a manual activity carried out by subject- matter experts. This is a time-consuming, expensive, and error-prone process. Some of the problems can be solved by computerizing the coding process. The computer can be incorporated in two ways. In automated coding the computer assigns codes to descriptions automatically, without user interference. Another approach is computer-assisted coding. In this approach the verbal description is entered and presented on the screen, and additional facilities are available to help the coder in an intelligent way to establish the correct code. The major difference between automated coding and computer assisted coding is that in the former the computer is in control of the coding operation whereas in the latter the human coder is in control.

Computer-assisted coding is available in the Blaise system. The coding module can be used in two different ways, called hierarchical coding and alphabetical coding.

Hierarchical coding starts by entering the first digit of the code by selecting the proper category from a menu. After entering a digit, the typist is pre-

sented a subsequent menu containing a refinement of the previously selected category. So the description becomes more and more detailed until the final digit is reached. In the case of alphabetical coding a verbal description is entered, and the computer tries to locate it in an alphabetically ordered list. If the description is not found, the list is displayed, starting at a point as close as possible to the entered description. For the sake of efficiency, the list should contain almost all possible descriptions, including synonyms and alternative spellings.

The Blaise team felt that the coding module could be made more useful and more effective. Research led to the development of trigram coding. This paper describes a test that was carried out with a special version of Blaise that contained a prototype of trigram coding. It was used in processing the Family Expenditure Survey. This special version recorded some extra information, allowing us to obtain more insight to the way trigram coding was used.

Section 2 describes the framework in which the test was carried out. It gives a short overview of trigram coding, and also presents some information about the Family Expenditure Survey. Section 3 contains an overview of the results of the analysis of the collected data. The subsequent sections go into more detail. The final section contains some conclusions.

2. Trigram coding in the Family Expenditure Survey

The CBS has been carrying out a Family Expenditure Survey since 1978. The survey collects data on income and expenditure of households. The sample consists of approximately 2000 households. They report on income and expenditure habits by means of questionnaires and diaries. The processing of the diaries with detailed daily expenditures is a particularly costly and time-consuming activity. Since 1988, the CBS has been using a Blaise CADI program to process the diaries. It uses the coding module to classify the expenditures. The coders first try the hierarchical approach to coding, and they only switch to alphabetical coding if they do not succeed.

Trigram coding is a new approach to coding. Trigrams are three-letter combinations. If trigram coding is applied, the entered description is split into all its succesive three-letter substrings. For example, the trigram set of the string 'bread' is {' br', 'bre', 'rea', 'ead', 'ad '}. Note that also the leading and trailing space are included. For trigram coding, Blaise splits all descriptions in the dictionary into trigram sets, and creates a special trigram index file for these sets of trigrams. When the coder enters a description, it is split into trigrams, and then the program locates those descriptions in the dictionary that have a high percentage of trigrams in common with the entered description. Only descriptions with a fit percentage above a certain threshold value will be shown on the screen. The coder can pick the proper description and attached code from the displayed list.

Trigram coding has a number of advantages. In the first place, it is able to cope with spelling errors. For example, if 'brown bread' is in the dictionary, and the entered description is 'bron bread', then there will still be a high trigram match. So, the item 'brown bread' will be in the list on the screen. In the second place, permutations in the wording of the description are no problem. The entered text 'bread, brown' would still be linked to the dictionary item 'brown bread'. In the third place, if the entered text is a substring of a dictionary description, then the program will also present the complete text as a possible candidate for classification. So entering 'bread' would lead to the suggestion 'brown bread', but maybe also to 'white bread'.

There is a lot more to trigram coding than is explained in this paper. For details we refer to Lina (1993).

Coding with trigrams was first implemented in a special test version of Blaise 2.4. From March 1992 up to March 1993, this version was used for processing the diaries of the Family Expenditure Survey. Each diary contains all expenditures in one week, expenditures of more than 50 guilders during holidays, the total amount spent per holiday, and expenditures of more then 25 guilders during a full year.

The classification consists of 2289 articles and the dictionary (with descriptions, alternate spellings, and synonyms) contains 11,221 entries. Every code

consists of three one-digit levels and one two-digit level. For example, the code for brown bread is 111.01.

The special test version of Blaise recorded data on the use of the different coding methods. Among the recorded information were: coding method used, keys pressed during coding, intermediate and final codes, and time needed to code one article. This information was collected on the coding process of 146,171 articles. The data is analyzed in the subsequent sections. There were 150 records not included in the analysis. For these records coding took more than 10 minutes. In these cases the coding process was probably interrupted by some activity like drinking coffee or taking a small break.

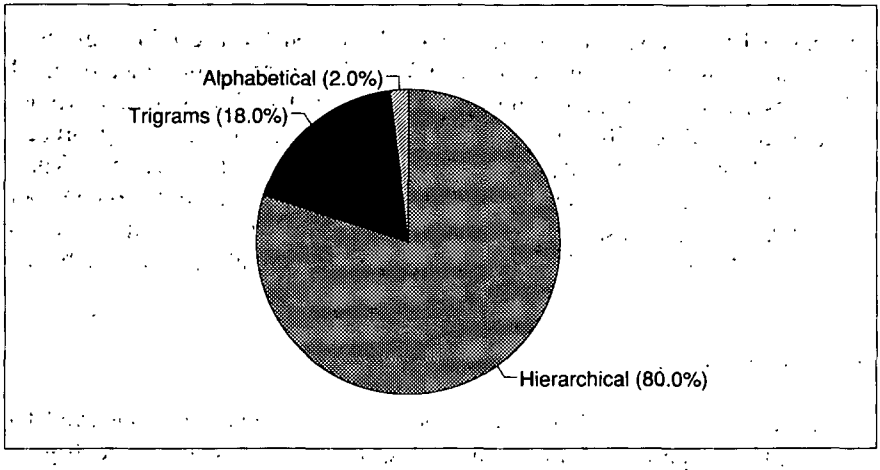
3. Overview of the results

The coders of the expenditure descriptions had three coding methods available during the test period: hierarchical coding, alphabetical coding and trigram coding. They were completely free in choosing the appropriate method for each description. Most coders had a lot of experience with hierarchical coding and alphabetical coding for the Family Expenditure Survey. Trigram coding was new to them, and they first needed some instruction on how to use this method. Their usual strategy is to start with hierarchical coding. They only change if they do not succeed in finding the complete code hierarchically to either alphabetical or trigram coding.

A coding attempt is classified by the final coding method used. So, if a coder starts with hierarchical coding and then changes to trigram coding, it is classified as trigram coding.

A coding attempt can lead to a succes or a failure. An attempt is classified as a success if a complete code is obtained, and it is classified as a failure if no code or only a partial code is obtained (some, but not all digits). Figure 3.1 contains a pie chart indicating which coding method is used. Hierarchical coding is the absolute favourite. This comes as no surprise. There are two reasons for this. In the first place, some descriptions occur

Figure 3.1. Frequency distribution of coding method used



very frequently. Many coders know the corresponding codes by heart. For example, the description 'brown bread' appeared 2,767 times during the test period. That is 3% of the cases. The corresponding code 111.01 is well known, and also easy to remember. In the second place, the coders do not need to type in the description for hierarchical coding, whereas alphabetical coding and trigram coding can only be carried out after the description has been entered. So hierarchical coding is much less work. In a very small percentage of cases both trigram coding and alphabetical coding are used as alternatives to hierarchical coding.

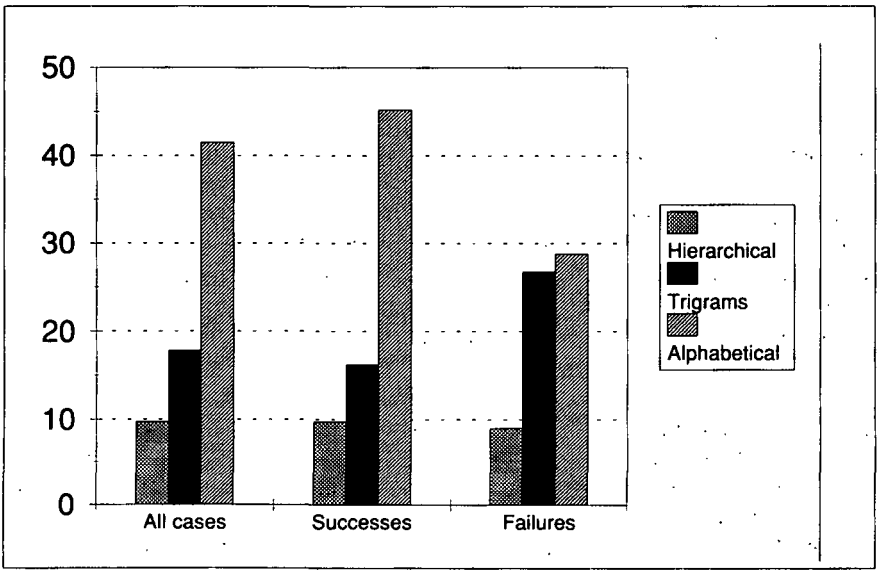
Table 3.1 compares the succes rates of the three coding methods. As could be expected, hierarchical coding is the most successful approach. Trigram coding is more successful than alphabetical coding. It is a much more powerful method than alphabetical search. This will come as no surprise as trigram coding has been designed to work in situations in which simple alphabetical coding fails.

Table 3.1. Success rates of the three coding methods

Coding method	Percentage of attempts	Percentage of successes
Hierarchical	81%	94%
Trigram	18%	85%
Alphabetical	2%	78%

Success rate is only one aspect of the usefulness of a coding method. Another aspect is time needed to find a code. Figure 3.2 contains a bar chart with the average time used for coding one article with one of the different methods.

Figure 3.2. Average time needed for coding one article



The first three bars, labeled 'All cases', relate to all cases (successes and failures). The second set of three bars, labeled 'Successes', relate to successful coding attempts only, and the last set of three bars, labeled 'Failures', denote the cases where no final code was determined.

Clearly, hierarchical coding is the fastest coding method. This can be explained by the fact that no text has to be entered, and moreover this method is used for the easy cases. Trigram coding takes more time, and alphabetical coding is very time-consuming. However, in case of failure the time spent is approximately the same for trigram coding and alphabetical coding.

From this general overview we can draw the conclusion that hierarchical coding is the prefered method for coding expenditures. In case hierarchical coding fails, one should turn to trigram coding, and not to alphabetical coding. Indeed, trigram coding is a valuable improvememt in the Blaise coding module.

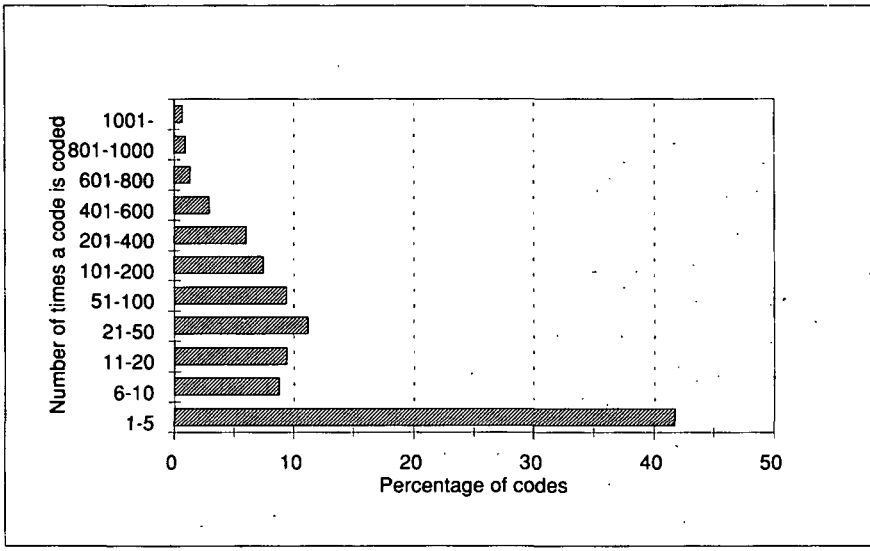
In the next sections the data about the three coding methods are analyzed in more detail.

4. Hierarchical coding

Coders prefer hierarchical coding, because they do not need to enter descriptions. In more than 65% of the cases (95,439 cases) they determine the final codes this way. These cases relate to 1126 different articles. When a coder does not know a code by heart and still wants to use only hierarchical coding, he has to make three choices from a list with at most 9 items and one choice from a list with at most 90 (but usually about 15) items.

Figure 4.1 contains a graph of the frequency distribution of the codes. The frequency distribution of the codes is divided into a number of classes, and for each class the length of the bar denotes the number of different codes in that category. On the average each article appears 85 times in the file, but the frequency distribution is very skew. For example, brown bread appears 2767 times (3% of the cases).

Figure 4.1. Frequency distribution of the codes

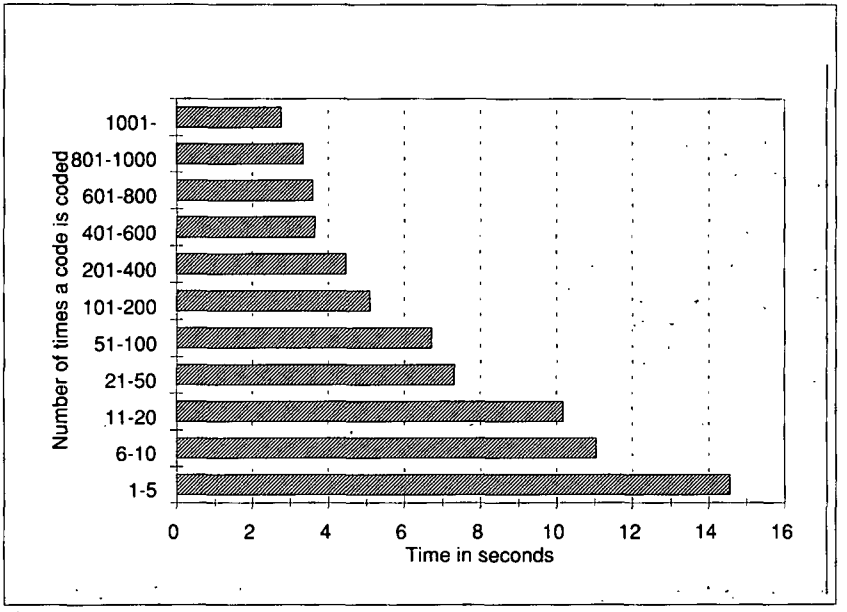


Coders learn from experience. The higher the frequency of appearance of an article, the faster they code it. This is illustrated in figure 4.2. Here the length of the bar denotes the average time needed to code items in that category.

Figure 4.2 shows a clear correlation: the higher the frequency of an item, the less time it takes to code it hierarchically. On average, it takes a little more than 10 seconds to code an item hierarchically. Items with a frequency under 5 require on the average approximately 15 seconds, whereas items with a frequency of more than 1000 need less than 3 seconds. Brown bread is coded in the shortest time: on the average in 2 seconds. This code is also rather easy to type: 111.01.

To reduce errors, most codes are checked by a second coder. This check is carried out much faster than assigment of the code by the first coder. The

Figure 4.2. The average time needed for hierarchical coding



reason is that the assigned code and the corresponding description are already displayed on the screen. In 98% of the cases the second coder agrees with the result, and simply presses the enter key to confirm this. This requires on average 3 seconds. In cases where only the final level is wrong, it takes 8 seconds to make the change. All other cases relate to more serious problems, and there the mean time was 15 seconds.

In 8% of the hierarchical coding cases a text is also entered. It is not clear why the coders do that. Maybe they think they will not find the code hierarchically and will eventually have to change to trigram or alphabetical coding. This theory is not very likely, because the average coding time (7 seconds) is shorter than in the cases without description text. Perhaps some coders always enter text.

In 4% of the cases the coders try to find a code by hierarchical coding, but do not succeed in obtaining a final code. In approximately 78% of these failures there is no code at all, and in 22% there is only a partial code.

The quality of hierarchical coding is quite high: 94% of the cases result in a final code in a mean time of 10 seconds. The cases in which the coder starts with hierarchical coding and switches to trigram or alphabetical coding are handled in the next sections.

5. Alphabetical coding

Alphabetical coding is the least used coding method. In table 3.1 it was already mentioned that this type of coding was used in only 2% of the cases. Moreover, the success rate of alphabetical coding is relatively low (78%).

Table 5.1 presents a further subdivision of the cases in which alphabetical coding was used in the final stage.

Table 5.1. Use of alphabetical coding

Direct alphabetical coding	55 %
First hierarchical, then alphabetical	10 %
First hierarchical, then alphabetical, and then hierarchical	12 %
First hierarchical, then alphabetical, no text entered	9 %
No code	12 %
Other	2 %
Total	100 %

In the majority of the alphabetically coded cases (55%), the direct approach is followed: first the coder enters a text and then he tries to locate it in the alphabetically ordered list. In 31% of the cases the coders try hierarchical coding first, and only change to alphabetical coding if they do not succeed.

In 9% of the cases the coder starts coding hierarchically and changes to alphabetical coding without entering text. This is not a very efficient method, because the coder has to page through a long list to find the right code. In the worst case this list contains 3,252 articles and on the average more than 1,000 articles (100 screens).

Table 5.2 contains the mean number of seconds required to code items in the various categories of alphabetical coding.

Table 5.2. Mean time required for alphabetical coding (in seconds)

Direct alphabetical coding	22
First hierarchical, then alphabetical	49
First hierarchical, then alphabetical, and then hierarchical	51
First hierarchical, then alphabetical, no text entered	70
No final code	12
Total	42

Direct alphabetical coding requires twice as much time as hierarchical coding. Starting with hierarchical coding and then switching to alphabetical coding at least doubles the time spent on coding an item. It also becomes clear that alphabetical coding without entering text is very inefficient.

6. Coding with trigrams

In 18% of the cases the coder uses trigram coding to obtain a final code. The success rate of trigram coding is 85%. Trigram coding can only be activated after the descriptive text of the article has been entered. In our test the lengths of the larger part of these texts varies between 3 and 20 characters. The modal length is 10 characters.

The time needed to code these descriptions varies between 9 and 15 seconds. The mean time is 11 seconds, making it slightly longer than hierarchical coding (10 seconds). Very short texts (3 characters) and very long

texts (15 characters or more) require most time (14 seconds or more). Texts of 6, 7 or 8 characters require the least amount of time (less than 10 seconds).

In 2% of the cases, the coder first attempts hierarchical coding and switches to trigram coding if he does not succeed in finding a code. Table 6.1 gives an overview of these cases.

Table 6.1. Trigram coding after a hierarchical coding attempt

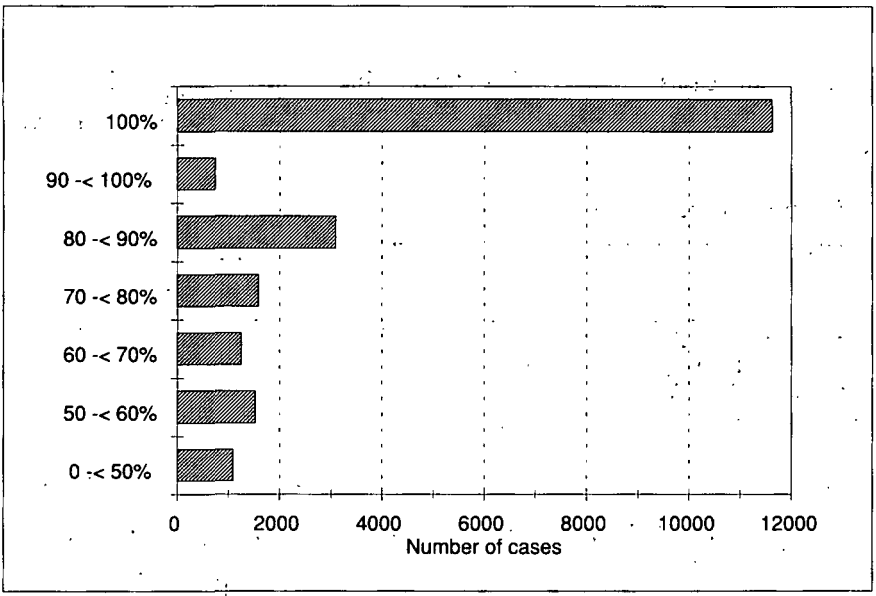
	Percentage of cases	Average time in seconds
After coding 1 level	12 %	36
After coding 2 levels	26 %	27
After coding 3 levels	57 %	26
After coding 4 levels	5 %	29
Total	100 %	29

The trigram search is also used after a partial hierarchical search (2,546 cases, 1.7% of the total). In total the coding takes then about half a minute. These must be some of the more difficult cases. Probably the coder thinks he can do it hierarchically, but at a certain point he cannot continue so he switches to trigrams. On the average this type of trigram coding takes about half a minute.

In 1% of the cases the coder uses trigrams, but does not succeed in finding the right code. In a small part of these cases the length of the descriptive text is less than three characters, in which case the trigram method does not work.

The trigram algorithm works in such a way that only candidate descriptions and codes are displayed on the screen which have a high matching percentage compared with the entered description. Furthermore the candidates are ordered in decreasing order of matching percentage. To see how well this works we recorded two quantities for items that were successfully coded

Figure 6.1. Frequency distribution of the matching percentages

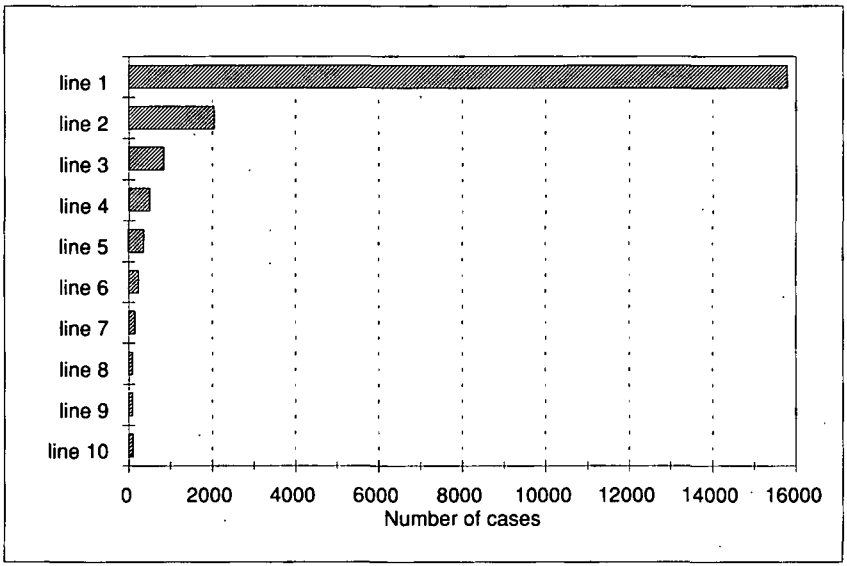


with trigrams: the matching percentage for the selected description and the line number of the proper code. Figure 6.1 contains the frequency distribution of the matching percentages.

It turns out that the matching percentages are rather high. For more than 70% of the cases the matching percentage lies between 80% and 100%. Figure 6.2 gives the distribution of the line numbers on which the correct item appeared.

In almost 97% of the cases the correct item was on one of the first ten lines, and in 75% of the cases it appeared on the first line. This is an indication that trigram coding works well, and is also very simple to use. In more than 75% of the cases the coder only has to read the first line and

Figure 6.2. Frequency distribution of the line numbers



press enter to select the right code. Still, it takes on the average 13 seconds to make this selection on the first line. Of course, this includes typing in the text.

does not

7. Conclusions

Blaise version 2.5 offers three ways of coding open questions: hierarchical coding, trigram coding and alphabetical coding. In our test with coding articles for the expenditure survey, hierarchical coding turned out to be the best method: it is easy to carry out and also requires relatively little time. However, one should observe that the coders are very experienced. They even know a lot of codes by heart. So this result is not very surprising. One

should also take into account that hierarchical coding can only be successful if a workable classification of items is available. It will often happen that a coder gets stuck in the middle of the process and therefore has to switch to a different coding method.

Alphabetical coding is not much used in this experiment. It has the disadvantage that the descriptive text has to be entered, making it more time consuming. Also there is no guarantee that this text will be encountered in the alphabetically ordered list. To make this method useful, the list has to contain alternative spellings, permutations of words and synonyms. This will make the list very long, and thus difficult to maintain. It is also possible to use alphabetical coding without entering the text. This faces the coder with the difficult task of paging through a very long list, which takes a lot of time, and moreover reduces the chance of locating the required item.

Trigram coding turns out to be a very attractive compromise. Although text also has to be entered, the method seems to lead to results in an amount of time that is not much longer than that of hierarchical coding. Of course, the succes of trigram coding also depends of the quality of the list with descriptions. Fortunately, this list need not be as long as the alphabetically ordered list, and therefore makes maintenance a lot easier. A final point in favour of trigram coding is that it clearly turned out be useful although the coders had no experience at all with it.

We may conclude that trigram coding can be a valuable tool for coding open questions. However, more research may be necessary to see how this method works in other situations.

References

Lina, M. (1993): *Blaise 2.5: Interactive Coding*. Netherlands Central Bureau of Statistics, Voorburg, The Netherlands.