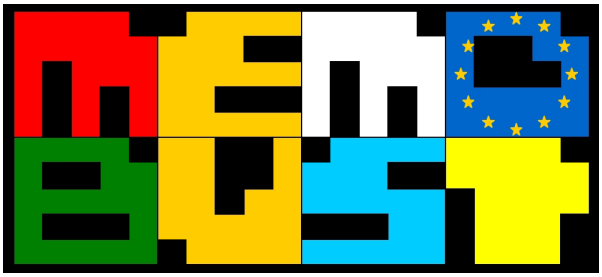

This module is part of the

Memobust Handbook

on Methodology of Modern Business Statistics

26 March 2014

Method: Computer-Assisted Coding

Contents

General section

1. Summary

First we should define what we mean by computer-assisted coding: it is a situation where a person codes an answer using the computer to search for possible classifications based on some search text. Compared to automatic coding the demands for such a program are less strict: the program may return multiple results, ordered by relevance.

Computer-assisted coding can be used:

1. During the interview if a question arises that requires coding: the coding can be done either by the respondent (e.g., CAWI) or by the interviewer (e.g., CAPI or CATI).
2. After the interview has taken place and some of the variables need to be coded at the statistical office by a coding expert.

Obviously, these situations require different approaches depending on the knowledge of the person that codes the question: if a respondent fills in a coding question, one must assume he has little or no knowledge of the targeted classification. On the other hand, a coding expert trying to code an open answer from the interview just has the information supplied in the open text answer as a basis for coding. Both situations require a different interaction with the computer: an unknowledgeable respondent needs to be taken by the hand to arrive at the classification, whereas the expert needs to be able to formulate a detailed search.

In the following sections we will describe two situations with a different degree of interaction. It will depend on the underlying search system how much interaction is possible.

All the pre-processing steps, such as stop word removal are applicable here as well, but will not be described; for more detail on these steps see Hacking and Willenborg (2012) and Sebastiani (2001).

2. General description of the method

Note that the computer-assisted coding task is less strict compared to automatic coding, where the computer not only has to search for possible codes, but also has to make a decision. That is not necessary in the method described in the present section: it is sufficient if the computer gives the N most probable classifications. The user makes the final choice.

2.1 Simple interaction (for the knowledgeable coder)

Performing assisted coding using an informative base constituted only by the classification manual would not be efficient, above all if coding is made directly by the respondent who has not the knowledge of the classification to be used. It would be better to build an informative base, integrated by pre-coded descriptions and/or other materials like synonymous, hypernyms, hyponyms (Macchia and Murgia, 2002)

When used interactively the search program only needs to supply a number of codes plus scores given a text from the user. The descriptions corresponding to these codes are shown in a list, sorted by score in descending order. As a search program, any of the automatic coding programs mentioned in the module “Coding – Automatic Coding Based on Pre-coded Datasets” can be used.

Interaction with the user

In this situation, the automatic coding program only supplies a list of possible codes; especially when this list is large (e.g., due to a vague description), a respondent may be overwhelmed. A possible solution might be to show only the list if the number of items is less than N items. If larger, the computer may ask for a rephrase of the search text. Alternatively, one may cut off the list at N items; but this may be dangerous, because one might throw away the correct code this way.

2.2 More interaction (for the less knowledgeable coder)

By its nature, this method is suitable to be used during the interview in both cases: if the person who codes is the respondent (self-interviewing) and if the person who codes is the interviewer (CATI/CAPI).

If more interaction is needed, the search program must be able to pose additional questions in case of a vague or ambiguous text. The automatic coding programs mentioned in the module "Coding – Automatic Coding Based on Pre-coded Datasets" could be altered to accommodate this. However, to our knowledge, this has not been done yet, except for the automatic coding program using "spreading activation". We will describe this in more detail later in this section. Blaise[1] (Blaise) also offers more interaction than just a list: it can also show the classification tree or part of it corresponding to the result of the search. We will now describe an extension of the "spreading activation" method, allowing the program to pose further question in case of vague or ambiguous answers.

Detailed description

The semantic networks described in "Coding – Automatic Coding Based on Semantic Networks" can serve as the basis for an interactive search technique as well. By assigning a dimension with the associated question text to every word in the network, we can use the network for interactive questioning. To illustrate: for the coding of 'education', we can add the dimensions 'level', 'is a teacher training', 'subject', etc.

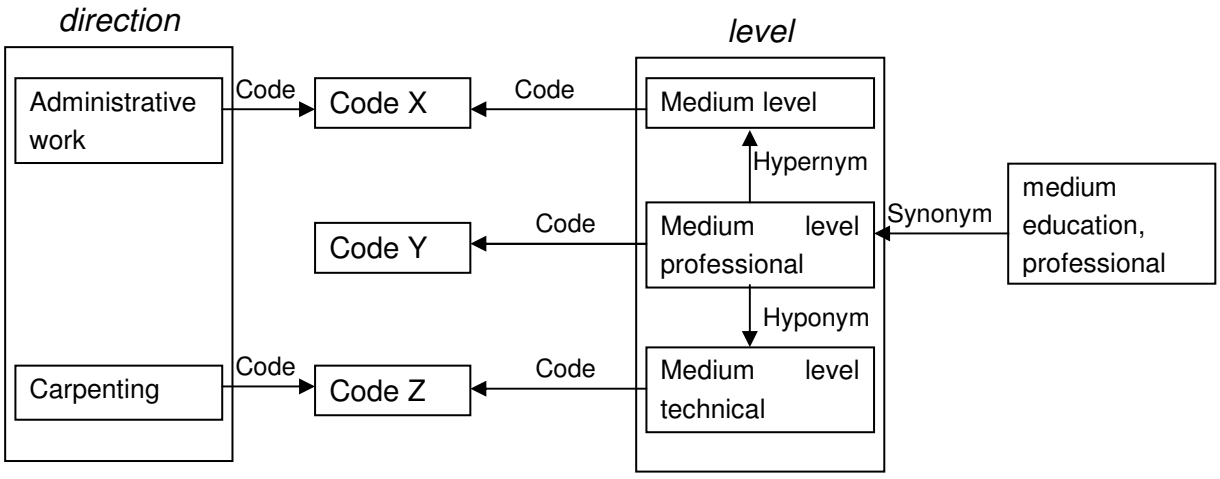

Figure 1: A fragment of the semantic network for the coding of 'education' to illustrate the use of a semantic network with dimensions (direction and level).

[1] Blaise is a general package for designing and doing electronic interviews (see www.blaise.com).

The interactive coding process starts with an open question about, for example, education. Based on the answer, which is used as a search string, a number, say N, of codes are selected in the semantic network (see Figure 1).

1. Open question $\rightarrow$ Codes $C = \{C_1,...C_N\}$ with the associated scores $S = \{S_1,...S_N\}$, sorted based on score $(S_{i-1} \geq S_i)$; N is the total number of hits. Call the number of codes with the same highest score M.
2. If $M = 0$ (in other words, no suitable codes have been found): either stop or ask for another description.
3. If $1 \leq M \leq_{MAX}$: show the codes found and let the user choose one.
4. If $M \geq M_{MAX}$: select the (next[2]) dimension D_i and make a list of all words W_k, which are both linked with the dimension D_i and with the codes in C; now also add the synonyms. Show a question or additional question associated with D_i and let the user make a choice from the list W_k. Each word in this list leads to a sub-selection $S_k \subseteq S$.[3]
5. The user makes a selection and reduces the set of possible codes: $S := S_k$; now continue with step 2.

Interaction with the user

As described above, semantic networks allow for much more user interaction: this makes it possible to guide a respondent towards a description that increasingly fits any of the classifications:

1. The respondent starts with an open text answer (starting with a closed answer would influence the answer too much, in general)
2. Then, either
    a. Very little codes apply: let the user make a selection from a list
    b. Too many codes apply: ask the next closed question (as described above) to try and reduce the number of codes.

Especially when a respondent uses such a system, user-friendliness is very important. There is no general method for this, but many little details contribute to the user-friendliness when constructing a program for computer-assisted coding. Note that this method is much more intended for the untrained user, in contrast with the previous method. To enhance the user-friendliness one may use so called fuzzy string-matching techniques (such as trigrams, Levenshtein, etc.; see Hall and Dowling (1980) and Navarro (2001)); these techniques allow the coding system to recognise incorrectly spelled words, e.g., ‘aple’ instead of ‘apple’.

[2] This order is predefined.

[3] This is therefore a ‘hard’ sub selection. If this is not desired, we can, for example, also reduce the set of records by repeatedly expanding the search string with the selected string from list W_k.

3. Preparatory phase

The preparation for both approaches of computer-assisted coding are almost identical to the preparations for both automatic coding methods. The main difference with automatic coding is the user interaction part, which needs to be added and thought about. Especially when dealing with a system that is intended for use by respondents, user-friendliness is very important to obtain good responses. In the case of computer-assisted coding based on a semantic network, there is also an additional piece of information that needs to be added to the semantic network: how are the words grouped and linked to a dimension and what is the question text associated with that dimension.

4. Examples – not tool specific

5. Examples – tool specific

5.1 Example 1: Interactive coding during CAPI/CATI at Statistics Netherlands

We will now look at an example of the interactive coding of occupations (based on the method described above) as realised at Statistics Netherlands[4]. This coding is performed during the electronic interview process for CAPI and CATI where one of the answers must be coded. To this end, in the Blaise interview, use is made of the option of adding a so called external plugin, which makes it possible to integrate external programs during the interview process. This plugin reads information from the Blaise interview and, on this basis, starts a coding session in which one or more questions are asked to arrive at a classification code. After the coding session, the selected classification code is written back to the Blaise form, and the interviewer or the respondent continues with the interview.

The method as described in section 2.1 is used to offer the respondent several options: sometimes one option in the case of a specific search string, and sometimes several options (e.g., via a drop-down list) in the case of a vague search string.

The interested user is referred to Michiels and Hacking (2004) for more information on the implementation details.

5.2 Example 2: More interaction

As described in section 2.2, coding based on a semantic network offers more possibilities to construct a computer program that interacts with the user.

To illustrate this, Table 1 shows an extract from the semantic network for education, in the form of a search table, to emphasise the link between the words and their associated dimensions (columns) and codes (rows). The extract is based on the initial answer ‘English’.

[4] In addition, a comparable module was developed for the coding of business activities (see Hacking et al., 2009).

Table 1. A small extraction from the table for the coding of education

C	***Level***	***Subject***	***IsTeacher-Training***	***University***	***TeacherType***
1	Senior secondary vocational education (MBO)	English	No		
2	Higher professional education (HBO)	English	No		
3	University	English literature	No	Master's	
4	Higher professional education (HBO)	Interpreter English	No		
5	Higher professional education (HBO)	Translator English	No		
6	University	English	Yes	Master's	First level teaching qualification
7	Higher professional education (HBO)	English	Yes		Second level teaching qualification

To further reduce the number of possible codes, we select the dimension 'IsTeacherTraining' with the associated question 'Is this a teacher training?' and answer set $W_k = \{yes, no\}$ with $S_{yes} = \{6,7\}$ and $S_{no} = \{1,2,3,4,5\}$. If, for example, we had chosen 'level', then the question would have been 'What is the level of the education?', $W_k = \{HBO, university, academic\}$ (*academic* is a synonym of *university* in the network) and $S_{hbo} = \{2,4,5,7\}$, $S_{university} = \{3,6\} = S_{academic}$.

This continued questioning technique is currently used for both the coding of education and the coding of the economic activity[5].

6. Glossary

For definitions of terms used in this module, please refer to the separate "Glossary" provided as part of the handbook.

7. References

Blaise, www.blaise.com.

Hacking, W. J. G. and Janssen-Jansen, S. (2009), The coding of economic activity based on spreading activation. Report, Statistics Netherlands, Heerlen.

[5] For the coding of economic activity, the subselection is slightly more subtle: instead of a hard subselection, there is a repeated search action based on an increasingly expanding search string.

Hacking, W. J. G., Michiels, J., and Janssen-Jansen, S. (2006), Computer assisted coding by Interviewers. IBUC2006.

Hacking, W. and Willenborg, L. (2012), *Coding – interpreting short descriptions using a classification.* Contribution to the CBS Methods Series, Statistics Netherlands, The Hague and Heerlen.

Hall, P. V. and Dowling, G. R. (1980), Approximate string matching. *Computing Surveys* **12**, 381–402.

Joachims, T. (2002), *Learning to classify text using support vector machines*. Kluwer.

Macchia, S. and Murgia, M. (2002), Coding of textual responses: various issues on automated coding and computer assisted coding. Journée d'Analyse des Données Textuelles JADT, Saint Malo.

Michiels, J. and Hacking, W. (2004), Computer assisted coding by interviewers. European Conference on Quality and Methodology in Official Statistics, Mainz, Germany.

Navarro, G. (2001), A guided tour to approximate string matching. *ACM Computing Surveys* **33**, 31–88.

Sebastiani, F. (2001), Machine learning in automated text categorization. *ACM Computing Surveys* **34**, 1–47.

Specific section

8. Purpose of the method

The purpose of computer-assisted coding is to guide a person when trying to classify an open text answer.

9. Recommended use of the method

1. Recommendations on the use of the different methods for coding (automatic or assisted) have been given in the module "Coding – Different Coding Strategies": the decision about which is the most suitable coding approach to be adopted in a survey depends on different correlated factors.

   There are two possible situations when CAC (computer-assisted coding) is useful:

   a. During an electronic interview: the method/program will facilitate the interviewer (CAPI,CATI) or the respondent (CASI) to arrive at a code corresponding to the description of the initial text.

   b. After all data have been collected, at the statistical office: coding experts can use the method/program to code the texts more quickly.

10. Possible disadvantages of the method

1. In some cases the computer-assisted coding method may show codes not suitable to the context, leading to incorrect codes. This is particularly so, when coding takes place by someone who does not know enough about the classification, e.g., a respondent. Another source of problems with selecting a description from a list, is that respondents or interviewers often select the first answer, without looking at or scrolling through all possible descriptions.

11. Variants of the method

1.

12. Input data

1. During the coding phase, the input is quite simple: a textual description, in most cases no more than 10 words.
2. During the construction of the "coding machine" the inputs can be:

   - pre-coded datasets and lists of different kinds of synonymous (hypernyms, hyponyms) that are used to train the coding algorithm;
   - the knowledge from experts to construct the semantic network.

13. Logical preconditions

1. Missing values

   1.

2. Erroneous values

    1.

3. Other quality related preconditions

    1.

4. Other types of preconditions

    1. The input text to be coded should not be too large; in general, this will result in many classifications by the method. This can be understood, since most methods do take word order into consideration and many different subsets from a large description may fit many classifications.

14. Tuning parameters

1. Often each result returned by the method has a score and the descriptions of the resulting codes are shown by score in descending order. Sometimes there are many low score codes at the end of this list that are probably not relevant. For that purpose, there is a score threshold value T: only codes with score > T are shown.

15. Recommended use of the individual variants of the method

1.

16. Output data

1. For each input description, the method returns a set of codes, each with a score.

17. Properties of the output data

1.

18. Unit of input data suitable for the method

Incremental processing.

19. User interaction - not tool specific

1. This has been described in more detail above.

20. Logging indicators

1.

21. Quality indicators of the output data

1. The quality indicators have been described in the module "Coding – Measuring Coding Quality":

    - Coding rate (efficacy) → percentage of coded texts on the total of texts to be coded.
    - Precision rate (accuracy) → percentage correct coded texts on the total of coded texts.

The verification of coding can be performed by a (different) team of coders on a sample of texts. If the original code and the verification code differ, the 'correct' code can be decided by expert coders by a reconciliation process.

22. Actual use of the method

1. The methods described above are used at Statistics Netherlands.

Interconnections with other modules

23. Themes that refer explicitly to this module

1. Coding – Main Module
2. Coding – Different Coding Strategies
3. Coding – Measuring Coding Quality

24. Related methods described in other modules

1. Coding – Manual Coding
2. Coding – Automatic Coding Based on Pre-coded Datasets
3. Coding – Automatic Coding Based on Semantic Networks

25. Mathematical techniques used by the method described in this module

1.

26. GSBPM phases where the method described in this module is used

1. 5.2 Classify and code

27. Tools that implement the method described in this module

1.

28. Process step performed by the method

Coding

Administrative section

29. Module code

Coding-M-Computer-Assisted Coding

30. Version history

Version	Date	Description of changes	Author	Institute
0.1	02-04-2013	first version	Wim Hacking	CBS
0.2	20-01-2014	following review by Stefania Macchia	Wim Hacking	CBS
0.3	30-01-2014	following review from EB	Wim Hacking	CBS
0.3.1	30-01-2014	preliminary release		
1.0	26-03-2014	final version within the Memobust project		

31. Template version and print date

Template version used	1.0 p 4 d.d. 22-11-2012
Print date	21-3-2014 18:07