

In the presentation of extracts of learner talk in the text of the thesis, however, Norwegian speech is translated into English. Where this is done it is marked with '<L1 *translated text* L1>?'.

Extract 5.8: Transcription of translated text

Line	Speaker	Text
688	Veronica:	<L1 what are you writing there L1>?/
689	Karen:	I played football,/
690		(.) outside the school._
		(9)
696	Karen:	you know that it's not --
		(6)
700	Karen:	<L1 hold on L1>.\

In extract 5.8, which illustrates this convention, the speech in lines 688 and 700 has been marked as translated. By contrast, the speech in lines 689, 690 and 696 was spoken in English. Note also that the more traditional pause convention is used in extract 5.8. That is, in extracts presented in the text of the thesis, pauses are not represented as two-second 'blocks'. The 'gaps' in the line numbers in extract 5.8 (e.g., between line numbers 690 and 696) reflects the fact that in the original transcription, which was used in the coding and visualisation of the data (cf. sub-sections 5.2.3 and 5.2.4), pauses were represented as two-second blocks (cf. sub-section 5.1.1).

5.2 Facilitating Coding and Visualisation

Moving from transcription of the data to the eventual visualisation of the dynamics of learner interaction involved identifying activity strands and threads, consistent with the dynamical perspective developed in chapter three. In addition, because of the lack of any precedent for the type of visualisation developed by the thesis, the constraints and conventions of visualisation needed careful consideration. Next, the transcripts needed to be exported to spreadsheet software where they could be coded. Finally, the coded transcripts needed to be visualised. These different elements in the development of the visualisation are discussed in the following four sub-sections.

5.2.1 Identifying Activity Strands and Threads

An implication of the dynamical perspective developed in chapter three was that activity strands and threads needed to be identified in the data. In chapter three, the two activity threads that Fischer and Granott (1995) identified, in the interaction of two adults trying to work out the operation of a Lego robot, were used as concrete examples (cf. sub-section 3.1.2). These two threads were 1) the participants' understanding of the Lego robot and 2) the participants' communication about the task. In addition, the theoretical discussion in chapter three suggested that, due to the emergent nature of the dynamics of learner interaction, each activity strand or thread would be non-reducible in terms of any other strand or thread, and that the identification of each activity strand and thread might require different analytical means (cf. sub-section 3.2.1).

The final identification of activity strands and threads in the transcribed learner interaction data was a highly recursive process. This process was in part driven by the example provided by Fischer and Granott, as well as the theoretical considerations outlined in chapter three. However, it also relied on a combination of 1) categories and findings of existing research on learner interaction, as reviewed in chapter two, 2) more general theoretical contributions in the literature on spoken interaction, 3) an understanding of the dialogue-writing activity the participants were engaged in (cf. sub-section 4.4.2), and 4) what emerged from the transcribed data itself. The in-depth discussion of each activity strand and thread, in section 5.3, elaborates on this identification process.

The following activity strands and threads emerged as the most promising for representing the dynamics of learner interaction in the particular activity the pupils were engaged in.

- A **writing strand**, defined as those points in time when each pupil completed writing individual turns of the role-play dialogue;
- An **attention strand**, defined as what the learners focused on at any point in the activity;
- A **language code thread**, defined by whether the pupils were speaking L1 (Norwegian) or L2 (English);
- Two **regulative threads**, both defined by spoken language that, in relation to the activity the pupils were engaged in, had a regulative function.

Consistent with the dynamical perspective, each activity strand and thread represent a distinct perspectives on the learner interaction. Moreover, the decision of naming the 'writing' and 'attention' perspectives as *strands*, and the language code and regulative perspectives as *threads*, was an outcome of the later visual analysis. Put briefly, the later visual analysis revealed that features in the language code and regulative threads were sometimes

‘embedded’ in features in the writing and attention strands (cf. in-depth discussion in section 6.1). Nevertheless, the full labels have been included in this chapter to facilitate the discussion of the identification, coding and visualisation of each strand and thread (cf. section 5.3), and to allow consistency in the naming of the strands and threads identified in the learner interaction data across the thesis.

5.2.2 Constraints and Conventions in Visualisation

There is little precedent for the visualisation of social or psychological processes in the time-dimension. Fischer and Granott’s (1995) research, reviewed in detail in chapter 3, is one exception. In addition, the microdevelopmental perspective more generally (cf. Granott & Parziale, 2002b) uses similar forms of visualisation. However, this approach relies on the same visual conventions used by traditional behaviour analysis (cf. Leslie & O’Reilly, 1999). That is, simple greyscale symbols (dots, squares, circles, triangles, and so on) signifying events, with connecting lines (solid or variously broken) representing some unspecified temporal connection or continuity between the behavioural events that are visualised. Other attempts at visualising interaction, widely interpreted, in psychological and applied linguistics research, have also followed these same general principles (e.g., Nelson, 1989; Spelman Miller, 2000). This traditional approach is not sufficient for the present study which has the objective to visualise a large number of activity strands and threads, each of which may be identified by a different analytical method, and each of which needs to be clearly differentiated from other activity strands and threads.

There is not only a lack of precedent for the type of visualisation that needs to be developed, but also a constraint in terms of computer software. That is, in the wake of new forms of visualisation there is usually a software development effort to make available the tools needed to undertake the relevant visualisation (e.g., Bernsen, Dybkjær & Kolodnytsky, 2002). For the type of visualisation envisioned in the present research no software exists as of yet, and the research has to make use of general-purpose visualisation resources available in existing software packages.

A final constraint is that visualisation is limited by the two-dimensional page, or screen, it will appear on. In general, this makes it difficult to visualise more than two, or possibly three, variables at the same time (Wainer & Velleman, 2001, p. 316). However, since the objective is to visualise learner interaction in the time-dimension, rather than visually determining quantitative relationships between two or more variables, this constraint is different in the present research. That is, several activity strands or threads may be visualised concurrently along the same timeline. The constraint, therefore, is more in terms of how many activity strands and threads can be interpreted, at any one time, by a reader/viewer of the visualisation.

This again underscored the need for each activity strand and thread to be visualised differently, so as to be clearly differentiable from other activity strands and threads.

Despite the lack of precedent, as well as numerous constraints, some potentially useful conventions were adopted from existing visualisation techniques. One such convention is the use of ‘time lines’ to illustrate contingency between stimulus and response events in behaviour analysis (cf. Leslie & O’Reilly, 1999, p. 27). These are horizontal lines, which spike upwards (much like a cardiogram) to signify a behavioural event. Another useful convention from behaviour analysis is the insertion of vertical lines, running from top to bottom in the visualisation of variables measured across treatment regimes (cf. Leslie & O’Reilly, 1999, p. 186). A final useful convention is hinted at by the distinction between visualising ‘events’ and ‘states’ in the software package *Observer* (Noldus Information Technology), designed for entering and analysing observational data in behavioural research. In a time-event plot feature of this software package, events are plotted as short vertical bars, evoking the association that what is plotted happens at a *point in time*, and states are plotted as longer horizontal bars, commensurate with the longer duration of states.

A useful decision in countering the software constraints was to use a modular approach to visualising the different activity strands and threads. That is, with existing software packages it is difficult to visualise different activity strands and threads concurrently, and at the same time give each strand or thread a distinct look. A modular approach, however, made it possible to visualise each activity strand and thread in a distinct manner, and then to display the separate visualisations alongside each other.

The development of the visualisation technique also adopted the standards for good *figures* as outlined by the American Psychological Association (APA, 1994, p. 142). These standards state that a good figure:

- augments rather than duplicates the text;
- conveys only essential facts;
- omits visually distracting detail;
- is easy to read - its elements (type, lines, labels, symbols, etc.) are large enough to be read with ease in the printed form;
- is consistent with and is prepared in the same style as similar figures in the same article; that is, the lettering is of the same size and typeface, lines are of the same weight, and so forth;
- is carefully planned and prepared.

Beyond this, the development of the visualisation technique relied on general design principles that take into account human perceptual abilities, such as well-planned layouts,

straight lines, starkly different colours and areas of simple shapes (Wainer & Velleman, 2001).

Where appropriate, the description of the visualisation of the activity strands and threads in section 5.3 makes reference to the above discussion of constraints and conventions in visualisation.

5.2.3 Preparing the Transcripts for Coding

While the transcription of the learner interaction data was done in Microsoft Word (MS Word), the coding and visualisation of the data was done in Microsoft Excel (MS Excel). Hence, to facilitate the coding of the data the transcripts had to be exported from a word-processing format to a spreadsheet format.

Figure 5.2 is a screen shot of what a ‘finished’ transcript would look like in MS Word. Note that the data was transcribed directly into a table in Word. This made it possible to export the finished transcript using the copy and paste commands.

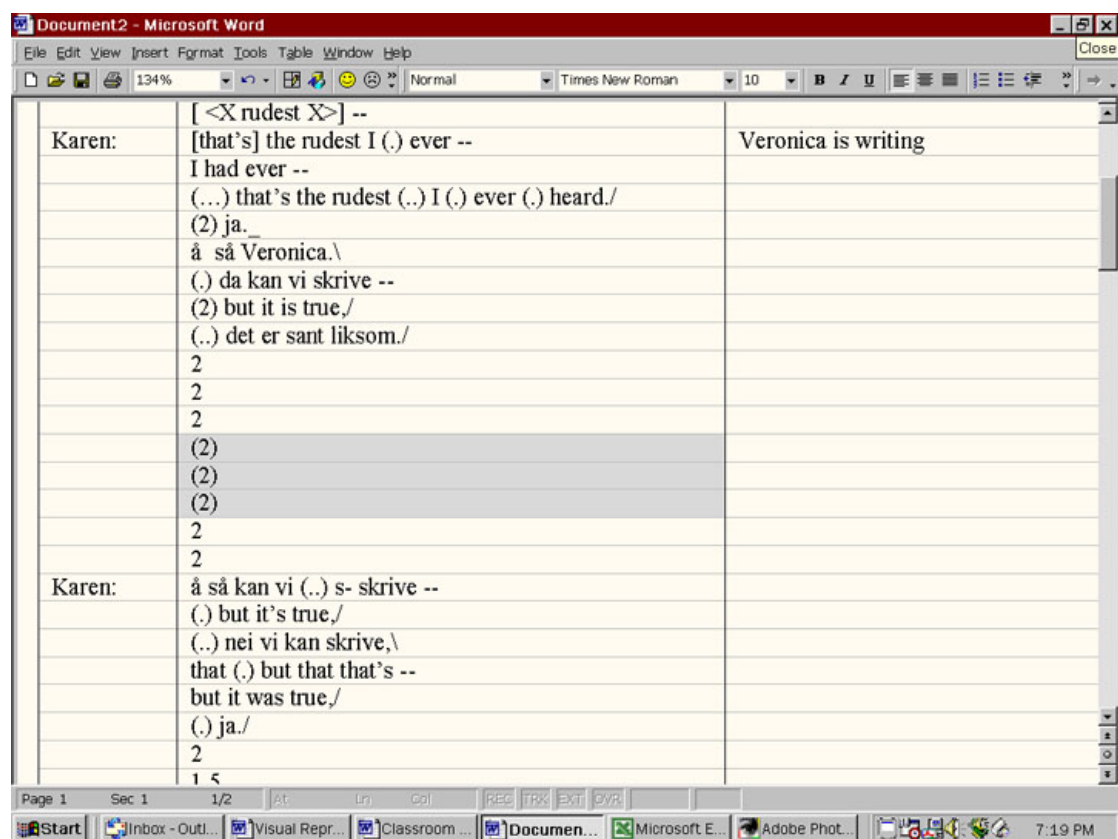


Figure 5.2: Transcript in MS Word (before export)

In order for the transcript in figure 5.2 to be successfully copied and pasted into MS Excel, the transcription would have to be what Du Bois (1991) calls a *robust transcription*. In the

words of Du Bois, “once the goals of maximizing data access and transportability across platforms are adopted, this immediately invokes a need for data representations which are robust enough to survive the moves with all their integrity intact” (1991, p. 88).

According to Du Bois, the necessary elements of a robust transcription include: 1) using characters that are available on all platforms; 2) avoiding contrasts which might be clearly visible in one format, but not in another format; 3) avoiding fragile contrasts, i.e., characters and formatting which will change when transcribed data is moved across software platforms. These elements are included in the design of the Du Bois et al. (1993) framework for transcription, which was used by the study. This ensured that, with only a few exceptions, the transcripts proved sufficiently robust to be pasted into the spreadsheet environment of MS Excel.

Figure 5.3 is a screen shot of the same transcript as in figure 5.2, but after it had been pasted into MS Excel.

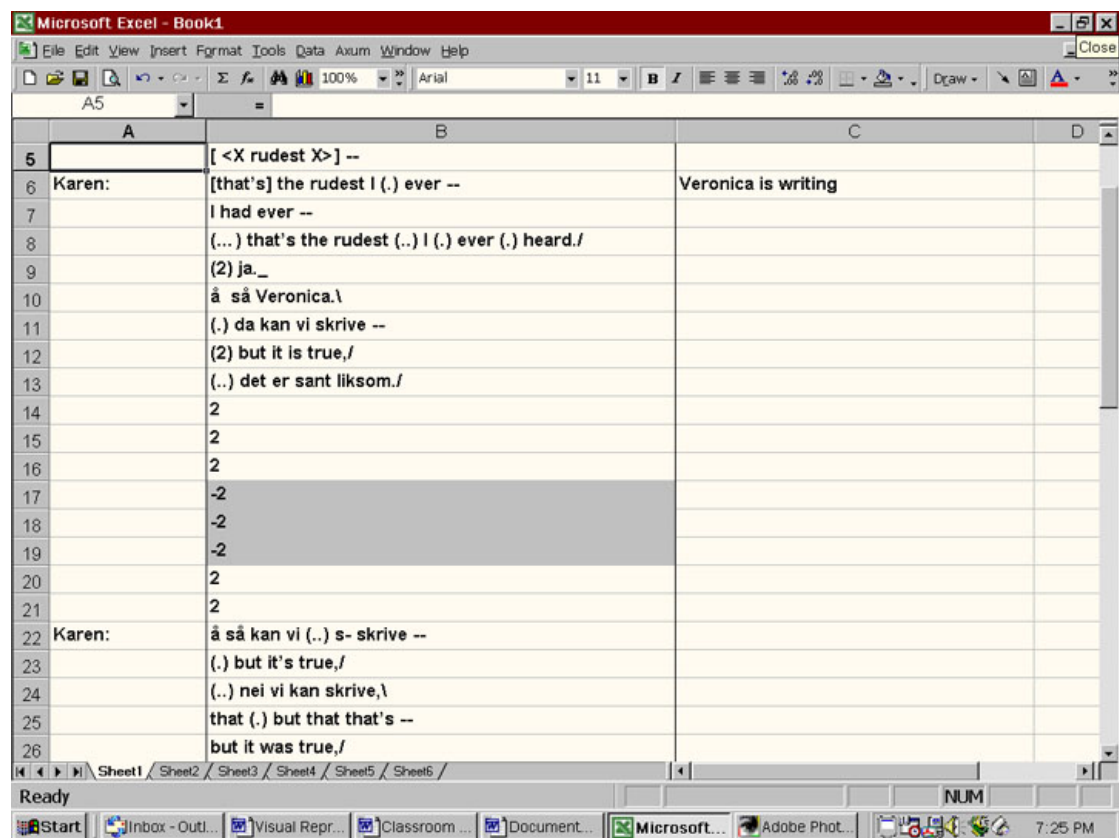


Figure 5.3: Transcript in MS Excel (after export)

Figures 5.2 and 5.3 contain an illustration of what may happen if transcription conventions are not sufficiently robust. In both figures, three pauses, which appear in two-second ‘blocks’ (cf. sub-section 5.1.1), are highlighted. In figure 5.2, which is the MS Word screen shot, these three pauses have parentheses around them. This is not the case for the pauses on the

preceding and following lines of transcription. Once the transcript was copied and pasted to MS Excel, this software, as is the convention for spreadsheets, interpreted the numbers in parentheses as negative numbers. To avoid this, all pauses appearing by themselves on a line were transcribed without parentheses.

Once the transcripts were successfully exported to MS Excel, extra columns could be added for the coding of the transcripts. For each activity strand and thread a separate spreadsheet, complete with the transcription of the learner interaction, as well the number of columns needed for the coding, was created. Figure 5.4 is a screen shot illustrating this for the coding of the language code thread. Appendices F through J, which will be referred to in later sections as well, provide similar illustrations of the spreadsheets used to code the other activity strands and threads.

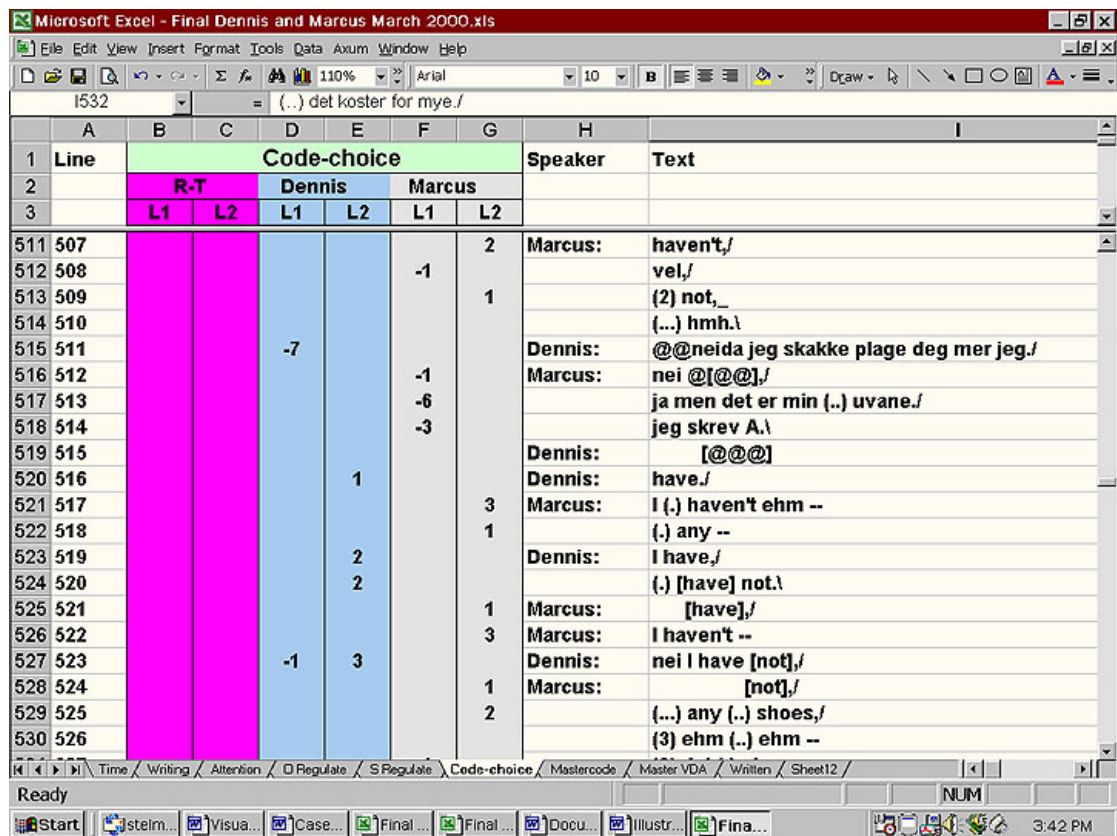


Figure 5.4: Coding of transcript in MS Excel

The use of numbers in the coding of the different activity strands and threads was sometimes deliberate, as in the coding of the language code thread in figure 5.4 (cf. sub-section 5.3.5), and sometimes more arbitrary. However, *the use of numbers was always deliberate in the sense that the numbers used would determine the 'shape', or layout, of the subsequent visualisation.* This should be clearer from the discussion of the visualisation of the data in MS Excel in the next sub-section, which picks up where this sub-section has left off. In addition,

section 5.3 provides a detailed discussion of the coding of each of the activity strands and threads.

5.2.4 Implementing the Visualisation

This section concludes the description of the process leading to the visualisation of the data represented in figure 5.1 in the beginning of this chapter. More specifically, this last section discusses any remaining technical aspects of the visualisation in MS Excel.

The practical problem of implementing the visualisation of the different activity strands and threads exploited the numbers used to code the transcripts in Microsoft Excel. Consequently, the numbers used in the coding of each activity strand and thread needed careful consideration. Appendices F through J, which contain illustrations of the spreadsheets used to code each activity strand and thread, show how the present study dealt with this element of the coding.

Line	Code-choice						Speaker	Text
	R-T		Dennis		Marcus			
	L1	L2	L1	L2	L1	L2		
4	000							
5	001							(2)
6	002							(2)
7	003	-1					Researcher:	her._
8	004	-3						(1) <X den andre boka X>./
9	005	-7						(1) så ta med den når dere flytter.\
10	006					-1	Marcus:	ok.\
11	007							(2)
12	008	-4					Pupil1:	skal du ha den./
13	009							{..}
14	010					-1	Marcus:	ja./
15	011	-2					Pupil1:	hvorfor det?_
16	012					-6	Marcus:	jeg er med på prosjektet./
17	013	-1					Pupil1:	pro[sj]-
18	014					-1	Marcus:	[pr]osjekt Andrew./
19	015							hæbedæ hædæbedæde hæbededa [blæbehæbe
20	016						Dennis:	[bæbediblæ]
21	017					-3	Marcus:	Julenissen [kommer ikveld]./
22	018		-2				Daniel:	[kommer ikveld]./
23	019							(1)
24	020	-5					Pupil2:	du er med i forskning...

Figure 5.5: Selecting data for visualisation in MS Excel

Once an activity strand or thread had been coded, with careful consideration of the numbers used, the first step in the actual visualisation was to select the data to be visualised. Both the line numbers of the transcript, as well as the coded data would have to be selected, as

illustrated in figure 5.5. In addition, (using MS Excel) unless transcription line zero (000 in figure 5.5) was included in the data selected for visualisation it was difficult to format the resulting visuals as required. The row numbers of the spreadsheet were not used in the visualisation.

After selecting the data to be visualised the MS Excel chart wizard was activated, the scatter plot option was chosen (for *all* the activity strands and threads), and the unformatted visual was inserted into a blank worksheet (it was not treated as a chart). Figure 5.6, which in fact is the language code thread of one of the pairs of participants, illustrates how such an unformatted visualisation would appear.

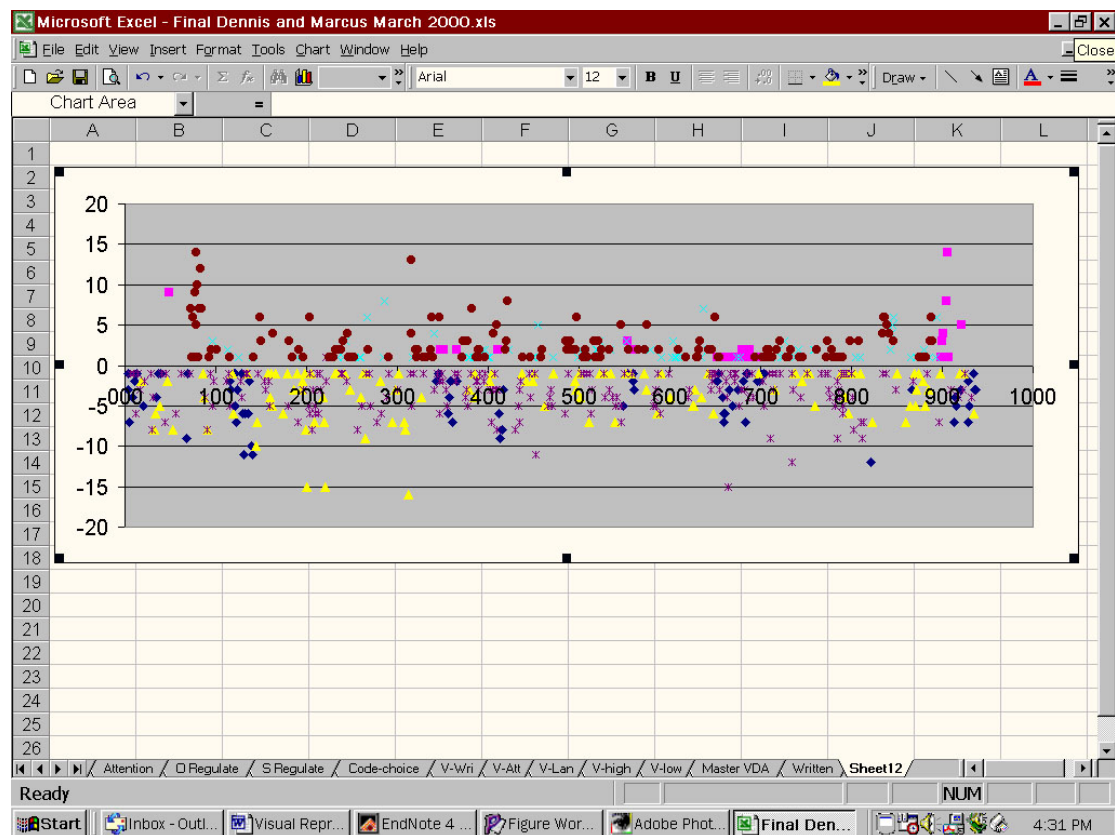


Figure 5.6: Unformatted visualisation in MS Excel

At this point the visual would be formatted as required. The first step here was to select individual data series, one by one, and change them to the desired chart type (e.g., for the language code thread each data series would be changed from scatter to columns). This was followed by any other required formatting, such as changing the colours of the data series, changing the size of the chart itself, formatting axes, inserting captions, and so on. Figure 5.7 shows a fully formatted visualisation of the language code thread (cf. also sub-section 5.3.6).

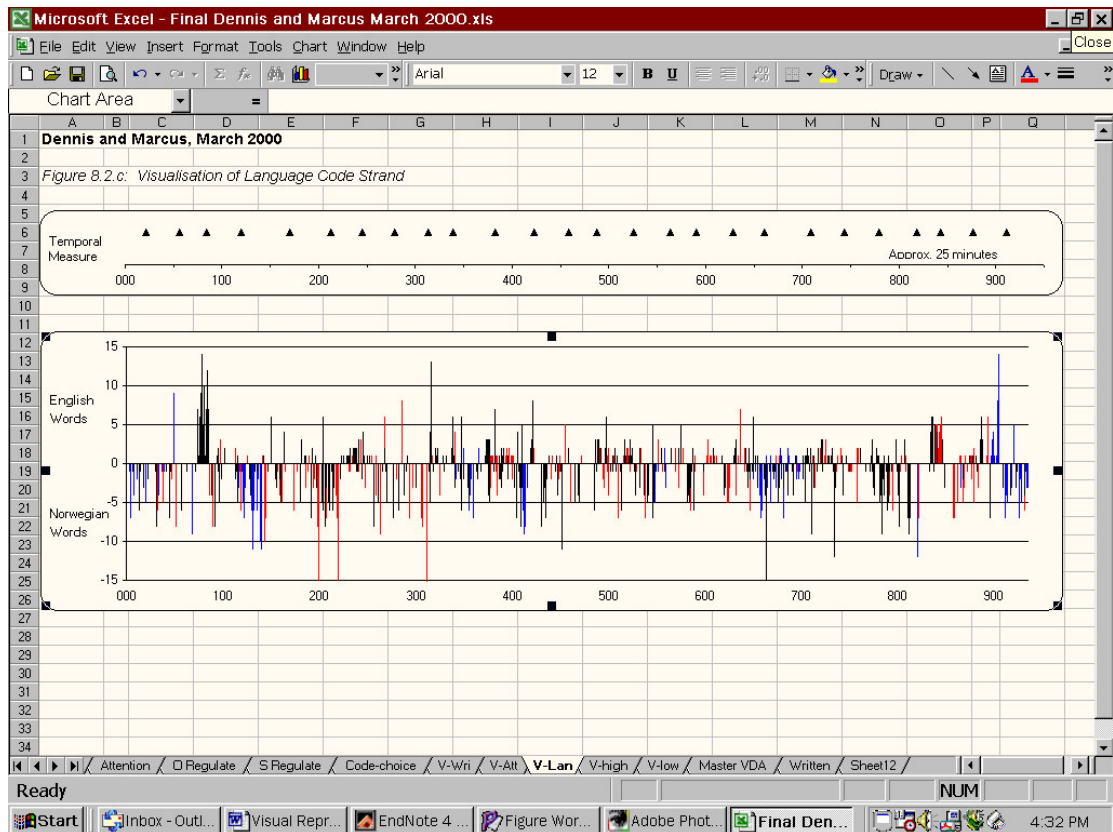


Figure 5.7: Formatted visualisation in Microsoft Excel

Note that the top part of the visualisation in figure 5.7 includes a second panel. This panel is a *time measure* included in the basic visualisation of each activity strand and thread. The purpose of this time measure is to give an indication of how long different features in the visualisations last. Preparing this time measure involved creating an additional worksheet in MS Excel, complete with transcription. A data series, with one data point for every minute that passed in the learner interaction, was coded in this worksheet. This was done by listening to the tape-recorded interaction with a watch, and recording those points in the interaction when another minute had passed. The time measure, therefore, could be compared to mile markers along a highway. That is, the time measure represents minute markers in the visualised learner interaction. The time measure also includes a caption indicating the approximate number of minutes the entire dialogue-writing activity lasted. Figure 5.8 is a magnified illustration of the time measure.

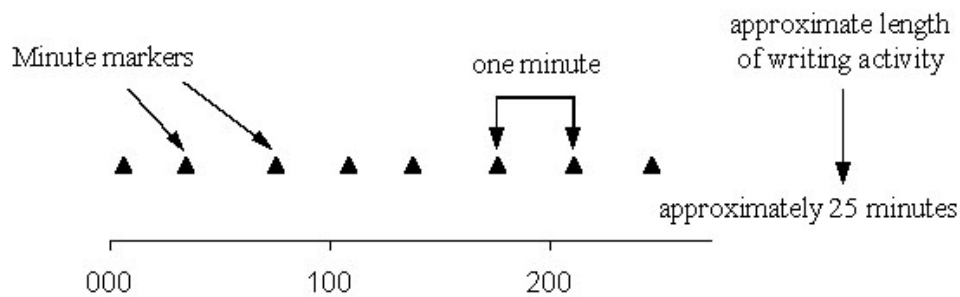


Figure 5.8: Visualisation of a time measure

At this stage the basic visualisations of each activity strand were printed, and further analysis could take place (cf. chapters 6 and 8). The modular approach to visualising the activity strands and threads allowed a great deal of flexibility in presenting the visualisations, without having to go back and redo the time-consuming procedure every time (cf. discussion in sub-section 5.2.2). That is, the visualisations of individual activity strands and threads could be combined through positioning one above the other in a worksheet.

The visualisation of all the activity strands and threads followed the same overall procedure described in the present sub-section. Section 5.3 includes a detailed description of the conventions used in the visualisation of each activity strand and thread.

5.3 Coding and Visualisation of Each Activity Strand and Thread

This section describes the coding and visualisation of each of the activity strands and threads identified in the learner interaction data (cf. sub-section 5.2.1). The discussion of each activity strand and thread is organised into two sub-sections. The first sub-section includes a discussion of why and how the relevant strand or thread was identified in the data, as well as a detailed description of the coding of the strand or thread. Appendices that contain sample MS Excel worksheets, as well as extracts of transcribed learner interaction data support this description. Another sub-section provides an introduction to the conventions used in visualising the relevant strand or thread. This introduction is supported by sample visualisations.

5.3.1 Coding of the Writing Strand

In all four successive role-play tasks (cf. section 4.5) the pupils had to compose and write a dialogue between two fictional characters. In the written texts that the pupils produced, this dialogue had a recognisable turn-taking structure, constituting a conversation between the two fictional characters. The only exception to this turn-taking structure was a title, which pairs