Using interviews to investigate implicit knowledge in computer programming

Rebecca Mancy, Norman Reid
Centre for Science Education, University of Glasgow, 11 Eldon Street, Glasgow G3 6NH
Rebecca.Mancy@educ.gla.ac.uk, N.Reid@mis.gla.ac.uk

Abstract: The notions of implicit learning and implicit of tacit knowledge first became popular in the 1960s (Reber, 1967; Polanyi, 1967). However, they have rarely been applied to education. Most empirical studies of implicit learning and knowledge have used experimental paradigms. We argue that in order for these notions to be investigated in real-world educational situations, other research methods are necessary. In light of this, we report the results of an interview study of implicit knowledge in computer programming education. Several cues that helped us to determine that knowledge was implicit are explained and illustrated with examples. We conclude the interviews can be useful in investigating nonconscious knowledge and offer some observations concerning implicit knowledge about programming that became apparent from this study.

Introduction

Implicit learning first appeared in cognitive psychology in 1967 with the work of Reber (1967), the same year that the philosopher of science, Michael Polanyi, published *The Tacit Dimension* (Polanyi, 1967). Although implicit *knowledge* has been investigated in both philosophy and psychology, we shall concentrate on the psychological literature as we feel this offers greater insights into the *processes* by which this knowledge is gained. As a consequence, we use the vocabulary associated with that area of research. Reber introduced the Artificial Grammar Learning paradigm with experiments that use strings of letters generated by a complicated rule base (the strings were generated using a finite-state machine). Artificial Grammar Learning experiments usually take place in two parts: a learning and a testing phase. Participants see letter strings generated by the artificial grammar and are required to carry out a task on these words (for example, they may be asked to repeat them back correctly). Following the learning phase, subjects are informed of the existence of the grammar underlying the patterns in the letter strings. During a second phase (the testing phase), participants see letter strings that they have to classify as conforming to the rules of the grammar or not. Alternatively, they may see pairs of letter strings, one of which conforms to the rules of the grammar, whereas the other does not, and must choose between the strings (for further discussions of the procedure, see Pacton, Perruchet, Fayol and Cleeremans, 2001).

The empirical finding from such experiments is that participants are able to correctly categorise grammatical strings at above chance level, even though they are unable to articulate the rules governing the material (Reber, 1967). In other words, participants act as though they possess abstract rules, but if this is the case, these rules are not declarative, and participants usually claim no conscious awareness of the rules. We use the term 'high-level abstraction' to describe knowledge structures that embody such regularities beyond reference to specific concrete examples. Implicit learning of regularities has been shown to be robust with numerous studies unambiguously demonstrating the same phenomenon. Similar results have been found for other implicit learning tasks where underlying rules are apparently learnt but that participants cannot articulate. Examples include a simplified version of the artificial grammar task where four-digit strings were used, each containing an invariant digit (McGeorge and Burton, 1990), interactive computer-based tasks with complex underlying mechanisms (e.g. the Berry and Broadbent sugar factory and "Clegg" tasks (Berry and Broadbent, 1984), and serial reaction tasks participants appear to be able to predict one instance from those preceding it (Nissen & Bullemer, 1987).

The diverse nature of these tasks indicates that implicit learning is not simply a specific process confined to grammar learning. What all of these tasks have in common is the need to acquire sensitivity to a complex underlying pattern, although as noted by Seber, different mechanisms may be at work when detecting different kinds of regularities. For example, some have claimed that learning such regularities in natural language may be supported by an innate Universal Grammar (Chomsky, 1965), but this argument presumably does not apply to the acquisition of programming knowledge. Although most empirical studies of implicit learning have concentrated on local perceptual patterns (Goswami, 2004), Kuhn and Dienes (2005) found implicit learning of a non-local musical rule, leading us to think that it may be useful to investigate less superficial forms of implicit learning.

Implicit and Explicit Learning in Education

The distinction between knowledge that is acquired and encoded with or without conscious awareness is relevant to education both during learning and also assessment. If we assume that both modes of learning are possible, the question in education becomes one of identifying those aspects of a task that are learnt more efficiently via one or the other process, in order to best exploit these two modalities. In doing so, it is necessary to be clear about not only the kinds of knowledge that are best gained implicitly or explicitly, but similarly how we can encourage this learning to take place. Teaching strategies that foster implicit and explicit learning are likely to be different. However, the reasons to use one strategy or another may vary: for example, it may be that in foreign language education, better long-term proficiency is gained by implicit learning of grammatical structure (and this would need to be confirmed), so despite the fact that teachers are fully able to explain the rules explicitly, and that learners can integrate these to an extent in either way, an implicit learning strategy may be preferable. However, other regularities - for example the plan structures in programming - may not be conscious or declarative for the teacher, and as such he or she can only hope to set up situations in which they can be learned implicitly.

From the point of view of testing, educators need to be aware that students may have declarative knowledge of some aspects of their learning, whilst other aspects remain impossible or difficult to consciously describe. Implicit and explicit knowledge need to be tested differently: we can ask students to recall their declarative knowledge; that which is stored implicitly can only be reliably examined by testing performance on pertinent tasks. It may be that this distinction is made naturally, since, for example, testing second language proficiency using questions that probe students' declarative knowledge of grammar would for most appear counter-intuitive, and comparing skiers on their ability to recount what they are doing when they ski is nigh on farcical. However, this is no guarantee that inconsistencies will not occur and as such clearly distinguishing the two types of knowledge could guard against such problems.

Investigating real-world implicit learning

Despite the importance of the distinction between implicit and explicit learning in education, it has proven very difficult to find studies in the education literature that discuss implicit learning. Perhaps it is not surprising that little research into implicit learning has been carried out in the field of education. As Claxton (1998) points out:

> In many areas of public and even private life, the Cartesian model has led to the assumption that explicit, articulate thinking is the most powerful form of cognition — the one to be trained most assiduously in schools, colleges and universities, and the one to be relied upon most heavily in formal situations of learning, problem-solving or decision-making. Ways of knowing that are hazy or metaphorical have consequently tended to be skimped, rushed, concealed or denigrated. (Claxton, 1998)

Mathews (1997) explains that it is the narrow focus of research on implicit learning that is to blame for its lack of penetration into areas such as education:

> Scientific debates tend to focus our attention on narrower and (hopefully) clearer issues, as authors try to isolate phenomena and prove a point. [...] Our research methods bias us to look for "it", the one true feature that will clearly separate implicit from explicit learning. As students of human knowledge, we should know better. (Mathews, 1997, p39)

Mathews claims that by concentrating almost exclusively on fine-tuning the distinction between implicit and explicit learning, researchers are failing to apply what we do know to real-life situations such as education. Similarly, since the research in this area is strongly rooted within experimental psychology, most studies of implicit learning have attempted to control as many parameters as possible, thereby concentrating on tasks that are far removed from the complex learning that takes place in real-world learning situations. Indeed, whilst rigour should clearly be maintained, in order to elucidate the issues arising in educational situations, we may need to emphasise synthesis and studies of complex learning as opposed to critical analysis of pure cases (Mathews, 1997). This may mean the development new research methods, or adaptation of techniques such as interviews. However, there are methodological issues with eliciting knowledge that is unconscious via interviews.

Interviews as a data source in investigations of implicit knowledge

There are problems with using interviews in investigations of implicit learning, some of which have been discussed elsewhere. In interview studies, we ask participants to give a verbal report of some knowledge. In such interviews, subjects are essentially asked to retrieve information from short-term or long-term memory. However, verbal reports may depart from the actual knowledge possessed if not all information in memory is reported (Ericsson & Simon, 1980). Nisbett and Wilson (1977) suggest that what is reported by interviewees depends on their own perception of their understanding, and that subjects tend to offer explanations that seem plausible to them. If they are unsure of their knowledge, they may choose to report nothing or give an alternative explanation that seems more plausible than the real explanation. These situations could arise in the case of either implicit or explicit knowledge.

An additional hurdle when studying unconscious knowledge is that conscious and unconscious knowledge may co-occur, and probably often do in real-life learning situations. If a subject has conscious knowledge, he or she is likely to offer this knowledge when asked a question about it. However, we cannot imply from this that no unconscious knowledge is present. Additionally, the fact that an interviewee is able to offer explicit knowledge during an interview does not guarantee that this knowledge is operative in a task situation: it could be that in such situations, implicit knowledge determines behaviour. It is difficult to demonstrate unconscious knowledge except when behaviour is not explained by conscious knowledge. This problem has plagued research in implicit learning since the beginning.

However, the situation is not perhaps as desperate as it sounds. As Baars (1997), points out, experiential or verbal reports have been extremely important in the development of knowledge in areas such as sensory processes, imagery and immediate memory, so may also be useful for implicit knowledge.

Empirical study - identifying implicit knowledge from interview data

In an attempt to investigate whether interview data can be used to gain information about implicit knowledge, we devised a series of interview questions to investigate implicit knowledge in programming. The questions specifically addressed decision points in programming, and related to concepts that could be abstracted out from the regularities seen in examples. Specifically, students were asked how they went about choosing between possible language structures, and as a follow-up question, whether they had thought about this particular issue previously. Example questions are included as part of the excerpts in the following subsections of this paper. In some cases, we knew that explanations had been taught in lectures whereas for others no explicit teaching had been given. For example, students had been given an explicit rule by the programming lecturer to help them choose between different types of loop. The rule stated that if the number of loops was known in advance, then a `for` loop should be used, and a `while` should be used otherwise. On the other hand, students had not been given a simple explicit rule that would help them make a decision about when to use an array.

The students who participated in the interviews were first year students studying introductory computer programming in Ada during 2003-2004. The interviews were conducted towards the end of the course, some before and some after the Easter break. Ethics approval was granted for students to take part in the interviews, and twelve students were recruited as volunteers via requests in the lectures and by email.. All of the students were male, although this may simply represent the difference in the number of males and females taking the course. Twelve interviews were carried out using a semi-structured procedures, with interviews lasting approximately an hour, and participants being paid for their time and assistance.

The interviews were tape-recorded and transcribed in full by the investigators, with attention being paid to aspects such as hesitation. A grounded theory approach was used to identify statements that seemed to indicate implicit knowledge, and later to categorise them according to the indicators of the implicit status of that knowledge. The investigators deliberately excluded any statements that appeared to indicate implicit knowledge but where it was felt that the form of the response may have been influenced by the interviewer. The results of the categorisation of indicators of implicit knowledge are shown below and are accompanied by illustrative excerpts. The extracts are taken from interviews with students who performed well in the examinations. This was done because with students who perform poorly, it is difficult to know whether apparent lack of knowledge is due to an inability to articulate implicit knowledge, or rather an underlying lack of knowledge.

Little or no knowledge about how skill is utilised

Perhaps the most obvious sign of implicit knowledge is when an interviewee claims not to know how he or she achieves success in a particular skill. It may be that until the question was asked, the learner is not even aware that he or she has knowledge in this area. The excerpt below concerns the use of semicolons as 'statement terminators' in Ada (rather like the use of full stops / periods in English).

> **Interviewer** – How do you decide when to use a semicolon?
>
> **Victor - *... I don't, I just do it. Occasionally like ... erm, I just ... it's when, it comes naturally when I'm typing something.***

In this extract, Victor gives the impression that he does not explicitly know how he goes about using semicolons, or where they should be used. Quite simply, it 'comes naturally', apparently without any conscious consideration or deliberation.

Claims of difficulty in articulating knowledge

In the following excerpt, Halsey indicates that he is aware that he has some knowledge, but explicitly states that this knowledge is difficult to articulate. The question relates to the situations when an array should be used when writing a computer program.

> **Interviewer** – How do you decide when to use an array?
>
> **Halsey -** *... erm, from, to start you'd say, is there ... lots of the same type of ... variable ... needed,* [ahuh] *so something like ... it's got lots of different ... for ... er values for the same thing.* [sure] *... **I'm just trying to think. I'm terrible with wording things.***

Halsey appears to be aware that he possesses some knowledge about this concept, and is even able to partially describe the situations when using arrays might be appropriate. However, his explanation is incomplete and he also seems aware that it does not fully explain his ability to decide between different data structures. As a result, he makes the statement that he finds it difficult to express his ideas. It could be argued that this knowledge is conscious but that it is represented in a form that is difficult to articulate in words. However, when it was suggested to Halsey that it might be easier to explain things using a picture or by writing something down, he felt that would not help. As a consequence it appears that whatever the representation of Halsey's knowledge, it does not translate well to any of the forms of communication suggested.

Reliance on concrete examples

During the interviews, students frequently failed to be able to provide a high-level abstraction of a computer programming concept, and resorted to use of examples. The excerpt below is a continuation of the extract above.

> **Interviewer** – you can write anything down if it's easier for you, or draw pictures, or ... it's up to you, I mean ...
>
> **Halsey** – *It's probably easier talking. No, that's not much help!* [laughs] *er, ... actually, when you use an array it's ... useful for ... erm ... **lots ... of the same thing held like, if you're wanting to .... hold the number of cups of tea you drink in a day, you would have like you know the numbers would be day one to seven would be your day ... and inside each would be your number of cups of tea.***

After a few aborted attempts to state generically the conditions under which he might use an array, Halsey resorts to giving an example. He seems much more confident when describing his example, and is consequently more fluent in his description of the concrete example than of the abstract rule. This particular example was not considered in lectures, so it may be his own example, invented on-the-fly, although he may have read about it or seen it at school. It seems as though, for Halsey, the example embodies the concept. A similar phenomenon has been seen in mathematics education where students appear to use concrete examples as prototypes (Tall & Pinto, 2002). However, this example does not simply exist as an isolated example: Halsey is not entirely incapable of accessing anything pertinent to an abstraction, but the high-level abstraction is implicit in the sense that it is too imprecise or inaccessible in a form that would allow him to explain his ideas coherently without reference to a concrete example.

Use of language as an indicator of implicit knowledge

A further cue to identifying implicit knowledge of high-level abstractions is the language used by students in their attempts to describe concepts. The following extract concerns the use of code indentation in Ada and specifically the code structures with which indentation is used.

> **Interviewer** – Can you explain how you use indentation, how you indent your code, when you write it?
>
> **Elwin** – *ah, yeah, ... generally I stick to sort of saving and using the reformat button* [laughs] *it puts it in the proper kind of way erm specially, seems to be that if it's a loop, it'll bring it in a bit,* [ahuh] ... ***specialist thing*** *so it's got to come in (sic)* [ahuh] *just makes it easier to read, sort of when you're going down (sic) or like or it almost lets you know that you're,* ***sort of something different's coming*** [yeah], *in a way,* ***there's all sorts of things****, put lines in (sic)*
>
> **Interviewer** – so, it happens with loops; does it happen with anything else?
>
> **Elwin** – *er, yeah, it happens ...* `if-then` ***kind of statements****, er ... yeah, it happens in ... ...* ***a lot of things like that, I think it's mainly the things that are dependent on stuff*** *and <inaudible>* [ok], ***those kind of structures***.

Elwin does not simply offer a list of situations when it is conventional to use indentation. He seems to see the examples he does give as being representative of some higher-level concept or set, as evidenced by his use of personal terms such as 'specialist thing', or 'something different', as well as expressions such as 'things like that', 'kind of statements'. However, none of his explanations would allow us to determine whether a particular case would be a member of the set. Set inclusion therefore seems to be implicit in the sense that although Elwin is aware of some 'sameness' of the elements of it, he is not aware of what would constitute a rule for set inclusion.

Surprise

A final indicator of the existence of a high-level abstraction is surprise expressed when that abstraction is discovered explicitly. In the following extract, Elwin has been asked to describe what might constitute a condition of a `while` loop (for example, the "`X<3`" in "`while X<3 loop`")

> **Elwin** – *Oh, it could be say* `while` *... ... technical ones like* `not Found`*, where* `Found` *is sort of a Boolean value, er ...* ***I suppose it could be any kind of expression sort of to do with*** *...* `while` *E is equal to five, or whatever* [ahuh] *just something that's ...* ***take it there's gonna be a value to come back?*** [aha!] ***Perhaps, does it always have to come back with a Boolean value, then?***
>
> **Interviewer** – It does.
>
> **Elwin** – *Yeah, it does, doesn't it! [Sounds surprised]*
>
> **Interviewer** – You'd not thought about that before?
>
> **Elwin** – *No ... no! [laughs]*

In this example, it appears that Elwin's prior knowledge concerning the kind of code segments that can be used as conditions in a `while` loop was implicit, but that during the discussion this knowledge becomes explicit. In natural language learning literature, this phenomenon is known as 'noticing' (Schmidt, 1990; Truscott, 1998). It should perhaps be noted that the student may have been aware of this regularity in the past and subsequently forgotten it.

Discussion

The interview studies above indicate that implicit learning and abstract structural knowledge can be usefully investigated in real-world situations using research techniques that are adapted to such learning situations. We have shown that students can possess knowledge of regularities without conscious access to that knowledge. In interview studies, implicit learning is not simply evidenced by an inability to report anything pertinent. A whole host of subtle clues such as hesitations, the time taken to formulate ideas, language such as 'things like that' which indicates a higher-level concept, as well as surprise on noticing structural patterns can be used to gain evidence of implicit learning.

Furthermore, interviews offer much richer sources of information about the form of implicitly learned information than experimental studies. For example, this study has highlighted the fact that much implicit knowledge seems to lend itself to explanation using examples. It also appears that implicit knowledge is very loosely structured, with ideas probably linked associatively as opposed to logically. However, even in the highly logical area of computer programming, the fact that knowledge is stored associatively appears not to render it inoperable.

Another example would be that the use of interviews has allowed us to see that with real-world knowledge, a strict dichotomy between implicit and explicit knowledge may be inappropriate. A graduated scale of conscious status of knowledge, as shown below, may well be more useful as a framework for understanding implicit knowledge in educational settings (Mancy, 2006).

1 - No awareness of any knowledge (subject does not realise that he or she has any knowledge).

2 - Awareness of knowledge, but no awareness of its content (subject is aware that he or she must have some knowledge, but has no idea what that knowledge might be).

3 - Some awareness of knowledge, but this knowledge is structured in a way that makes it difficult to communicated.

4 - Structured declarative or explicit knowledge.

Pragmatically, therefore, it appears that interview studies can be useful for investigating implicit knowledge and learning in real-world situations. However, the rate of occurrence of relatively unambiguous evidence for implicit knowledge is fairly low making this mode of investigation somewhat inefficient. This could perhaps be improved by looking at the type and wording of questions used so that respondents are able to give pertinent answers. Investigating the type of questions that are likely to elicit useful information would therefore be a valuable route for further study.

References

Baars, B. (1997). Some essential differences between consciousness and attention, perception, and working memory. *Conscious Cognition*, 6(2-3), 363-371.

Berry, D. C., & Broadbent, D. E. (1984). On the relationship between task performance and associated verbalizable knowledge. *The Quarterly Journal of Experimental Psychology*, 36(A), 209-231.

Chomsky, N. (1965) *Aspects of the Theory of Syntax*. MIT Press.

Claxton, G. (1998). Knowing without knowing why. *The Psychologist*, May, 217-220.

Ericsson, K. A., & Simon, H. A. (1980). Verbal reports as data. *Psychological Review*, 87, 215-251.

Kuhn, G., & Dienes, Z. (2005). Implicit learning of non-local musical rules. *Journal of Experimental Psychology: Learning, Memory, & Cognition*, 31, 1417–1432

Mancy, R. (2006) *Explicit and implicit learning: the case of computer programming.* PhD Thesis, University of Glasgow.

Mathews, R. C. (1997). Is research painting a biased picture of implicit learning? The dangers of methodological purity in scientific debate. *Psychological Bulletin & Review*, 4(1), 38-42.

McGeorge, P., & Burton, A. M. (1990). Semantic processing in an incidental learning task. *The Quarterly Journal of Experimental Psychology*, 42A(3), 597-609.

Nisbett, R. E., & Wilson, T. D. (1977). Telling more than we can know: Verbal reports on mental processes. *Psychological Review*, 84, 231-259.

Nissen, M. J., & Bullemer, P. T. (1987). Attentional requirements for learning: Evidence from performance measures. *Cognitive Psychology*, 19, 1-32.

Pacton, S., Perruchet, P., Fayol, M., & Cleeremans. (2001). Implicit Learning Out of the Lab: The Case of Orthographic Regularities. *Journal of Experimental Psychology*, 130(3), 401-426.

Polanyi, M. (1967). *The Tacit Dimension*. London: Routledge & Kegan Paul Ltd.

Reber, A. S. (1967). Implicit learning of artificial grammars. *Journal of Verbal Learning and Verbal Behavior*, 6, 855-863.

Schmidt, R. W. (1990). The role of consciousness in second language learning. *Applied Linguistics*, 11, 129-158.
Seger, C. A. (1994). Implicit learning. *Psychological Bulletin & Review*, 115(2), 163-196.
Tall, D., & Pinto, M. (2002). Building formal mathematics on visual imagery: a case study and a theory. *For the learning of mathematics*, 22(1), 2-10.
Truscott, J. (1998). Noticing in second language acquisition: a critical review. *Second Language Research*, 14(2), 103-135.

Acknowledgments

We would like to express our appreciation to the students who participated in this study. We would also like to thank the Professor Rex Whitehead and the anonymous reviewers for their comments that allowed us to clarify various aspects of this article.