Supporting Learning in Conversations using Personal Technologies

Paul D Rudman
University of Birmingham
pdr889@bham.ac.uk

Mike Sharples
University of Birmingham
m.sharples@bham.ac.uk

Chris Baber
University of Birmingham
baberc@eee-fs8.bham.ac.uk

Abstract

We describe the design and evaluation of the CASL (Conversation Augmentation by Semantic Links) software. The learner creates a concept map on a mobile computing device while conversing with a (relative) expert. The device also holds a complete map of the topic (initially hidden from view). As the learner creates nodes, the device compares these with the hidden map and displays matching nodes with their immediate links.
It is theorised that this extra information will help the learner to guide (scaffold) the conversation from the edge of their existing knowledge towards the topic area(s) that are most useful for their current needs. Preliminary results show the software to be effective in providing the necessary environment, with distinct changes to the conversation based on the augmentation.

1. Introduction

Our focus is learning outside of formal teaching, where one adult learns from another as they discuss a topic together, especially over the telephone, and take notes. For example, one person in a company attends a training course. Later, another employee needs to learn a small portion of the course material, and may ask the attendee. In such circumstances it can prove difficult for the learner to ask appropriate questions, since (s)he knows only their problem, whereas the course attendee will have knowledge of the course but not necessarily the experience to select the most useful information based only on the learner's questions. The object of this research is to find a way to support such conversations, assisting the learner to guide the conversation more effectively towards their learning objective.

We propose that the learner uses a hand-held computing device to build a concept map as the conversation, and their learning, progresses. As the learner adds nodes to the concept map, the computer system compares each new node to a hidden 'teacher's map' - a complete concept map of the domain (which for the example above would have been prepared as part of the training course). When a match is found, semantically linked nodes in the hidden map are displayed for the learner. It is envisaged that these prompts will help the learner to explore the other person's domain knowledge in a more structured and effective manner.

We describe the design and evaluation of the CASL (Conversation Augmentation by Semantic Links) software. CASL follows from an earlier pilot study using a manual simulation [1]. Its main functionality is a very-easy-to-use node and link editor, with automatic word matching and display of related nodes from a secondary (hidden) concept map.

2. Theoretical basis

2.1. Zone of Proximal Development

The theoretical basis for this work begins with Vygotsky's 'Zone of Proximal Development' (ZPD) [2]. This determines learning as taking place in the ZPD between what is understood and what cannot be understood at present, even with the help of another person.

The intention is to apply the ZPD concept to adult learning during a conversation, since similar factors are involved to children's learning [3, pg.2]. The other person would act as the 'more knowledgeable peer', with the computer using the learner's emerging notes to help locate his/her current ZPD.

2.2. Scaffolding

Each new note (node on the concept map) then can provide the basis for estimating the domain area currently being learned, while the ZPD concept suggests that semantically linked nodes may relate to areas that can be learned with help (from the other person). The system's objective is to prompt the learner to discuss domain areas that (s)he is ready to assimilate. It is not the system's role to decide which potential area should be discussed. Rather, the system suggests areas that logically follow from that currently under discussion (i.e. the latest note taken).

In this way, the learner is presented with a continually updated scaffold [4] of related ideas to discuss. This enables the learner to ask about any of the domain areas that relate directly to the current area, not just those (s)he already knows, can extrapolate from existing knowledge or are mentioned by the other person. This scaffolding can help the learner to direct the conversation in whatever they consider to be the most useful direction to move their ZPD towards their overall learning objective.

3. The software

An Object Oriented approach was used for the design and programming, with the CASL software being written in JAVA, making it easy to port to a number of mobile devices.

3.1. Concept maps

For the software to monitor the learner's current area of study, there must be some form of externalisation of the learning process by the learner. Attempts have been made to categorise the many forms of external representation (see Blackwell and Engelhardt [5] for an overview). From this and other work it was decided to ask the learner to create a concept map of the domain as (s)he learns.

While this may assist the learner to visualise the domain, the main purpose is to provide the system with constantly updated information on the current focus of the learning process. It also serves a secondary purpose of providing a convenient tool for an expert or teacher to create - in advance - a comprehensive 'teacher's map' for the computer to refer to.

3.2. Example

Figure 1 shows an example of the CASL software (only the top-left quarter of the screen display is shown). The very top-left of the screen is reserved for displaying the automatic prompts. The remainder of the screen is initially blank and available for the learner's concept map. When a node is attended by the learner (created, amended, moved or 'hovered on' - if the hardware allows this), a word matching algorithm searches the hidden map and displays the text from the nearest matching node, along with up to four linked nodes.

The example in Figure 1 shows a learner's concept map of an artificial domain (an imaginary 'murder-mystery' - created for the purpose of evaluating the software). In the example, the node 'Medicine bottle and dropper' is highlighted (in blue, on the screen) since the cursor has recently 'hovered' over that box. The matching algorithm has found a match - in this case, also 'Medicine bottle and dropper'. The text of this matching node is displayed in the reserved area; underneath is shown the text from linked boxes in the hidden map.

In this case, the prompts 'Bedroom 1' (where the bottle was found) and 'Heart medicine' (the bottle's contents) have already been discussed (they appear on the learner's map). However, the prompt 'Fingerprints-medicine-Jane/Frank/???' does not yet appear on the learner's map. This is therefore something that the learner could, at this point, ask the other person about. Fingerprints on the bottle is something which the learner may not think to ask about. Similarly, the other person may not think to mention it - they may have forgotten, or perhaps think it was not relevant (to them, but possibly not to the learner). Thus, the learner is prompted to consider a relevant domain area which they may otherwise not find, or may reach by a round-about and time-consuming route.

3.3. Interface

The interface takes into account potentially restricted input devices, using only single-key mouse, or pen commands. In particular, there are no menus or buttons in order to minimise the extra cognitive load on the learner through using the software while carrying out a conversation.

Pressing on empty space creates a box at that location and allows input from a keyboard (real or virtual) or a handwriting recognition system, while pressing within a box allows the text to be edited. In either case, subsequently pressing outside the box, or using the RETURN key, ends input. Links are created by dragging from one box to another. Boxes may be relocated by dragging (the links follow the box), or deleted by dragging to a 'delete' area.

The text prompts that appear at the top of the screen may also be dragged into the main area, where they become boxes.

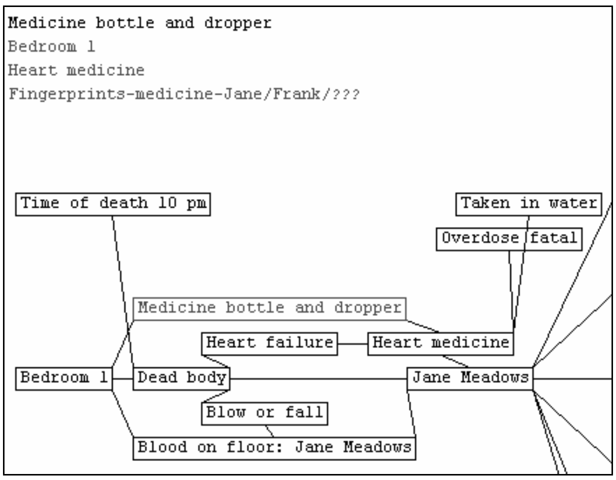

Figure 1 - Partial screen of the CASL software

3.4. Word matching

When a node is to be matched against the hidden map, it is first broken into individual words, with common words, such as 'and', removed. The remainder are compared against all words in the entire hidden map using a adaptation of the ‘Term Frequency inverse Document Frequency’ (TFiDF) algorithm [6]. This searches for the ‘term’ (word in the learner’s node) which appears with least frequency in the ‘document’ (hidden map) to find a matching node. If more than one node is found then the next-least common word is used as the term within the ‘document’ (nodes matched to the previous term) to reduce the number of matched nodes, as far as possible to one node. The text from this node is then displayed, followed by up to four linked nodes.

4. Evaluation

The CASL software has been evaluated using the artificial domain described above. A brief overview of the experiment is given here.

Eighteen undergraduate participants were allocated into pairs. First, they were introduced to the domain. One participant from each pair was given, on paper, all the information available (which (s)he retained for the whole experiment). The other participant (the ‘learner’) was shown a sub-set (which (s)he did not have access to during the main part of the experiment).

During the actual experiment, the learner had only the CASL software (with or without prompts enabled) or, in the third condition, pen and paper. The learners task was to create a comprehensive concept map of the domain and to solve the crime by obtaining information over a telephone link from the more knowledgeable person.

Observation of the participants found rapid learning of the interface with only a few, brief interruptions to the experiment due to interface-related problems. Informal interviews of the participants revealed a consistent liking for the system, with most finding it helpful and easy to use.

A full analysis of the results from this experiment is currently being prepared. However, it is clear that the pairs given prompts by the CASL software were able to cover a greater proportion of the domain in the time given. Indeed, several of the non-augmented pairs stopped, believing they had finished, without having considered one or more crucial aspects of the domain.

Based on this, other versions of the software are planned. It is intended to ensure the system’s ease of use and effectiveness for a future trial within a relatively realistic learning situation.

7. Acknowledgements

We would like to thank the EPSRC and British Telecommunications plc for their support of this research.

8. References

1. Rudman, P.D. *Supporting Conversations with Vicarious Learning using Personal Technologies*. in *Interact 2001*. 2001. Tokyo, Japan: IOS Press.
2. Vygotsky, L.S., *Mind in Society*. 1978: Cambridge MA: Harvard University Press.
3. McKendree, J., *et al. Why Observing A Dialogue May Benefit Learning: The Vicarious Learner*. in *PEG '97*. 1997. Sozopol, Bulgaria.
4. Bruner, J.S., *Child's talk: learning to use language*. 1983, Oxford: Oxford University Press.
5. Blackwell, A.F. and Y. Engelhardt, *A Meta-Taxonomy for Diagram Research*, in *Diagrammatic Representation and Reasoning*, M. Anderson, B. Meyer, and P. Olivier, Editors. 2002, Springer-Verlag.
6. Salton, G., *Automatic Text Processing. The Transformation, Analysis and Retrieval of Information by Computer*. 1988: Addison-Wesley, Reading, MA.