Talking to the computer

At the Information Technology & Systems faculty of TU Delft, Ir. Robert van Vark has come up with a method for travel information queries that can chat flexibly.

Traveller waiting on the platform at Amsterdam Central Station. In the event of delays, travellers have little access to current information.

Christian Jongeneel

 Whenever computers communicate with humans, for example on the telephone, the machine will conduct the dialogue along strictly defined lines. After all, computers can't cope with the rather casual way humans tend to converse. So what you get is a form to be filled in on the screen, or at best a question and answer routine with a limited range of options. At the Information Technology & Systems faculty of tu delft, Ir. Robert van Vark has come up with a method for travel information queries that can chat flexibly.

Suppose you're calling travel information and your call is answered by a computer telling you: 'If your departure station starts with A or B, press 1; if your departure station starts with C or D, press 2', etc. When you've made your choice, the computer continues: 'Your departure stations starts with C or D. If it is C, press 1, if it is D is, press 2'. You make your choice, and then you
'Your departure station starts with D. If the second letter of your departure station is A or E, press 1.' This is repeated for the destination station, and then for the departure or arrival time. If you make one small mistake at the end, you have to start all over again. There is little chance of such a service ever becoming very popular with the general public. So, the travel information service lines are staffed by operators who know exactly what callers mean when they say they want to depart from Rotterdam to Amsterdam at four the next morning. The problem is that the Dutch travel information number is called many more times than the staff can possible handle. In 1995 it got 13 million requests for information, only 10 million of which could be processed. So even back then, the service was looking for methods to automate, in part or in full. The solution was to train a speech recognition program using dialogues between travellers and operators. Some 5000 dialogues were painstakingly typed from audio tape, including redundancies, ums and ahs and the dialogue outlined above was streamlined quite a bit. Once the computer understands callers spoken replies, three or four questions will suffice to get the information it needs to provide a simple itinerary. Even so, the dialogue remains under the computer's strict control, so there is no real conversation, what you have is an oral questionnaire. There had to be a better way.

Hello

'We analysed about 500 calls,' says engineer Robert van Vark, who worked on the project, 'using a coding table to define the various elements of the call, such as greeting, question, answer, etc. The question was which dialogues could be processed automatically and how many.' Quite a few, as it turned out. Three-quarters of enquiries calls follow such a rigid pattern that they could be processed automatically. Not that they would have to follow a predetermined script; they could simply use the human form of dialogue. A good automated method ought to be able to cope equally well with opening sentences such as 'This is Smith speaking from The Hague' and 'I want to travel to Amsterdam tomorrow' as with the more experienced traveller's 'I want to depart from Delft at four o'clock tomorrow for Rotterdam'. One aspect Van Vark looked at was how operators handle their callers. Experienced callers will state their purpose right away, whereas new users require some coaching. In many cases, the interaction is very subtle, many people using a short guttural sound to indicate that they've understood what is being said. The research also showed that well-presented information is easier to remember. So instead of volunteering a long string of information all at once, you need to speak clearly and present the information in concise sections.

Van Vark: 'On paper, the dialogues sometimes look strange, but if you listen to the tape, you realise that this is indeed the way they tend to go. Operators are very good at assessing how to cut up the information into smaller bits for callers to digest.'

The call analysis was the starting point of Van Vark's...
post graduate research at the Data & Knowledge Systems group, which he hopes to complete early this year. The travel enquiries seemed the perfect research object for studying man-machine dialogues, since the subject area of the calls is very accurately defined. Artificial intelligence research has produced quite a few «artificial personae» that can handle wider-ranging dialogues, but they still make lots of mistakes, or give nonsensical answers, which makes them unsuitable for actual use - except for people who call enquiries just for a chat.

Targets
Van Vark's research supported the development of pita (Personal Intelligent Travel Assistant), part of the Seamless Multimodal Mobility research programme (see last page) of the trail research centre. He built a system that could handle dialogues on travel information in a flexible way. The general principle is as follows. The system starts with a main target, which is to fulfil the caller's need for information. The main target is then subdivided into a number of sub targets until a list of single targets remains. The single targets are then resolved by putting questions to the caller ('where do you want to travel to?'), or by consulting the built-in database ('at what time does the first train to Amsterdam leave?'). When every target has been achieved, the dialogue is finished.

'The dialogue manager consists of a number of modules,' Van Vark explains. 'Each time a caller says something, it is first processed by a module that tries to remove any ambiguities. For example, when the computer tells the caller to take the four o'clock train to Amsterdam, many people will repeat the word Amsterdam by way of confirmation. The module filters out this type of content because it does not constitute information that is required by the rest of the system. Another thing is that people who ask for travel directions to Amsterdam Amstel will use 'Amsterdam' later on in the conversation. The module has to understand that this does not mean that they want to travel to a different destination.'

The second module keeps track of the targets that have been achieved, for example whether the place of departure has been narrowed down to a single possibility. If users indicate the small town 'Alkmaar' as their destination, the system has to respond differently than it would if they asked for 'Amsterdam'. In the latter case the module has to add a new target to its collection of targets, i.e. to ask which particular station in Amsterdam users want. A smart system would start by assuming that the user wants Amsterdam Central Station, but this would still have to be verified by asking the user. The second module also has to detect when the user deviates from his original query, for instance by asking the system to repeat something, or by answering a departure time query with 'I want to arrive in Amsterdam at four'. Although this is not an answer to the question, it is useful information, so the module has to determine which of the targets it helps to achieve.

The third module controls the direction of the conversation. Depending on the course of the

The transcribed dialogues were coded by hand using this diagram. By processing a large number of dialogues in this way, typical characteristics of conversations were made visible.

Schematic diagram of Van Vark's dialogue manager. The blue modules are essential to flexible dialogue processing.

Impression of a future model WAP phone that will be able to inform travellers about the latest itinerary deviations. Such systems are expected to be able to provide a service to train, tram, bus, underground, and ferry passengers as well as motorists.
conversation, it uses predefined rules of reasoning to determine which of the targets to go for next. If the second module detects that the user has requested for information to be repeated and so has put the 'repeat previous' target on the top of the stack, the third module ensures that the sentence, 'Your train leaves Delft at three o'clock' is repeated.

Bits and pieces
By cutting up the complex problem of dialogue management into separate sub problems, Van Vark has made his system appear rather simple. Even so, the difficult task remains of ensuring that the system can handle every nuance. For instance, the dialogue manager will have to be able to detect from the way in which callers provide information whether they are experienced or unsure travellers. In the latter case the information must be asked for and presented in bits and pieces. This means that the second module has to cut the targets into even smaller sub targets. A traveller who says 'I want to leave Delft at four tomorrow for Amsterdam Amstel Station' can be given the full itinerary in one go. Other travellers will first be told at what time they have to be at the station in Delft. When the user confirms this, the computer will continue with the time the train will arrive at Amsterdam Central Station where the user has to change trains. What makes Van Vark's system so special is that instead of processing the targets one by one in a fixed order, it determines which target it is best to tackle next. This makes it very different from existing systems like the one used by Die Bahn (German railways).

Van Vark: 'Current systems are based on a single, large dialogue with all its variations included. If you want to add a single element, the system doubles in size because you have to cater for each of the variations, depending on the reply to the question. The advantage of my system is that it is easier to add a target without increasing the complexity of the system to any great degree.'

As it happens, Van Vark is the first to admit that his approach has drawbacks too. The modules have to communicate a lot with each other, which puts quite a strain on the computer. This is not such a problem in a test environment, but when a system has to handle thousands of calls concurrently, computer capacity becomes a major consideration. Halving the speed of the method means twice the number of computers to handle the same number of calls. Therefore, the benefits of the method must be weighed against the investment required to put it into practice.

Connections
The system is not fully operational yet. Van Vark's job is to demonstrate the viability of its operating principle. Industry will have to pick up the development to turn it into a commercial product - which is, after all, not a university's task. Even so, development on the system continues. Graduate students are currently engaged in actually connecting Van Vark's system to speech recognition software. To date, it has only been tested using written dialogues similar to the ones you could
expect if man and machine were to communicate through the Internet or a WAP (Wireless Application Protocol) mobile phone with limited Internet access. The next step might even be using a WAP phone to buy your train ticket. The ticket would be a code on the phone's display that the ticket collector can check to see if you've paid for the journey. An experiment using this form of ticket vending is currently taking place in Austria. Whatever the case, Van Vark will stick to handling the dialogue. Which is an advantage, because it means that his work can be used to connect to all kinds of different systems. The dialogue does not have to be conducted by phone, nor does the conversation have to be about travel information. The idea behind the Van Vark's dialogue manager can be applied to any dialogue with well-defined subject matter. If you are going to be asking relatively simple questions by telephone, you will find yourself being served increasingly often by a computer. And it will be a computer with human traits, not the kind that only understands the meaning of beeps signifying nought to nine.

For more information, please contact Ir. Robert van Vark, phone +31 15 278 1537, e-mail r.j.vanvark@cs.tudelft.nl, or Dr. Leon Rothkrantz, phone +31 15 278 7504, e-mail l.j.m.rothkrantz@cs.tudelft.nl

Seamless Multimodal Mobility

Seamless Multimodal Mobility, or SMM, is a research programme of the Delft-Rotterdam research centre, TRAIL, under the auspices of Prof. Dr. Ir. Piet H.L. Bovy. TRAIL is researching new, intelligent ways to tackle transport problems. The emphasis is on making travelling more dynamic by seamlessly switching between modalities. The role of PITA is to project the information more dynamically. So, instead of giving a single itinerary and waiting for further queries when things go wrong, the object is to provide optimum information all through the journey. As a simple example, if an office is ten minutes from the station, and rail traffic control knows that a certain train is running fifteen minutes late, travellers could be sent a message even before they left the office, saving them the extra wait on the platform. A major role in the research programme is being played by PITA (Personal Intelligent Travel Assistant). The original purpose of this concept was to act as a separate device, but the mobile phone and electronic diary boom makes integration with those devices the obvious solution. PITA provides users with continuously updated travel information so they can select the optimum route. Vice versa, travellers can ask PITA for information in a natural way. Before it becomes reality, an enormous amount of information will have to be collected and made accessible in a smart way. This is not much of a problem with train time tables, road maps, and similar information, but the challenge is to provide information about trains running late, traffic congestion, and...
roadworks. An example of the type of research forming part of the SMM programme is the intelligent dynamic route planning system developed by graduate student Gerritjan Eggenkamp. He has created an expert system that uses traffic congestion information collected every minute along part of the Dutch motorway network to determine the best route for motorists to follow. In some cases alternative routes will be quicker, in other cases it will be better to stay in the traffic-jam and sit it out.

Compared with existing methods, Eggenkamp's is faster. It does have the disadvantage that designing an expert system takes a lot of time.

It will be a while yet before we can use PITA to transfer seamlessly from car to train, from train to tram, underground, or coach-on-call. Once it happens, PITA can start contributing to its own accuracy. With many different people in contact with the central system, the system can use the information it gets from them to update its own knowledge about the current transport situation, and so provide still better advice.
Street map of Delft (under) and its network representation (above). Student Ronald Kroon used it to develop a method that can guide motorists to available parking spaces via the quickest routes.