Issues in the Development of an Intelligent Human - Machine Interface

A.P. Breen

BT Applied Research and Technology
Main Laboratory building
Floor 3, Room 44, B62
Martlesham Heath, Ipswich, IP5 3RE, UK
apb@dwarf.bt.co.uk

Abstract

A significant component of any intelligent environment is the human - machine interface. It is highly likely that in the future such an interface will, for the majority of applications, closely model human to human communication. In fact we may expect that the human - machine interface will increasingly mimic the behavior and appearance of humans. Two years ago BT set up the Maya project. The aim of this project was to research into the spoken language and kinesic[1] aspects of such an interface and to provide an effective computational research framework. Due to scale of the problem, Maya collaborates closely with a number of other groups within BT, these include speech synthesis (Page and Breen 1996) or (Edgington, Lowry, Jackson, Breen and Minnis 1996) recognition (Pawlewski 1996), understanding (Wyard 1996) and virtual humans (Breen 1996) or (Breen, Bowers and Welsh 1996). This paper provides an insight into the underlying principles governing developments within the Maya project.

The paper begins with an introduction to a number of the issues affecting natural human - machine discourse. It then briefly describes the computational framework being developed within the Maya project and uses the example of speech synthesis to argue that advanced research into spoken language and Kinesics is best achieved within such an integrated framework.

Introduction (So what do people really want?)

Over the last two decades, researchers have regularly predicted the imminent appearance of listening and speaking machines in our every day lives. This technology however, still has a long way to go before it can be considered widespread in society. Some services are starting to appear, but this slow change in fortune has more to do with dramatic changes in the computer and telecom's industries, than it has to do with any significant breakthrough in speech and language research. The simple fact is, that while some applications can now be handled adequately with the current generation of systems, the advanced applications, those wanted by the vast majority of people and which would sit comfortably within an intelligent environment are still many years away.

An obvious question to ask at this point is "So what do people really want?". One way to answer this question, is to divide communication with a machine into three broad application areas:

- Truly natural discourse with a machine
- True summary
- Domain sensitive database retrieval.

These three classes are designed merely for explanation and are not meant to represent a comprehensive categorization of human - machine communication.

Truly natural discourse with a machine

In its widest sense, this represents the Holy Grail for many speech and language researchers, where the term "natural" in this context is defined as meaning a man-machine interaction which exhibits all the components of a conversation between two humans speaking on the some topic and in the same speaking style.

Discourse in its most general sense contains both verbal and non-verbal cues. A great deal of information is born in the speech signal, but a significant amount of information is also contained in pausing and gestures. Truly natural conversation with a machine will only be possible once all the components, both verbal and non-verbal are considered and mastered. Simply stated, the computer must be sensitive to the emotional and physical state of the human interlocutor. To do this, it must have access to as many different modalities as possible. It is not enough for a computer to have a pleasant visual and acoustic persona. This persona must reflect the mood of the interlocutor, the content of the discussion, and the social context.

Truly natural discourse, as described above, while necessary is not sufficient. As the title of the paper suggests, a machine must exhibit a degree of intelligence as well. The ability to "understand" spoken requests or retrieved information, is fundamental requirement of any interface and as such is the subject of the next two sections.

True summary

Many applications require, or are enhanced through, the use of data summarization. Document summary, for example, is

[1] The relationship between body movements (e.g. blushes, shrugs, head or eye movements) and spoken communication.

a growth area, but the current generation of summarisers do not attempt to analyze and re-interpret the text. Instead, they use statistical techniques to extract highly relevant portions of existing text. True summarisation would attempt to understand the contents of the document and re-state it in a way appropriate to the discourse and the communication medium. As an example, consider the short E-mail given below:

Hi Donald,
Got your E-mail regarding the meeting on Friday with Warner Bros. I'll be there.

Regards,
Daffy.

Consider now a brief spoken dialogue, taken from an imaginary advanced automated E-mail enquiry system:

User: Do I have any E-mails from Daffy about the Warner Bros. meeting?
E-mail system: Yes you have one from Daffy. He says he can make the meeting on Friday.

Here the E-mail system has interpreted the users request and generated an appropriate reply. The example above demonstrates that improved usability can be achieved through the appropriate use of summarization and language generation.

Alternatively, if the user had said:

User: Show me any E-mails for Daffy about the Warner Bros. meeting.

In this example, the system must interpret the spoken request "show me" appropriately and respond by displaying all the E-mails from Daffy about the meeting. Notice also, that the system must recognise that the words "the Warner Bros. meeting", refer to a particular meeting. Finally, for this spoken request to be performed correctly, the machine must be "aware" of the types of output media available and choose the most appropriate.

Domain sensitive database retrieval

Many texts should not be summarised, but still need to be treated differently depending upon their specific characteristics. For example, one application of speech synthesis systems is as reading machines for the blind. Such reading machines are insensitive to the type of material they are reading. A book is read in the same impassive style as a news paper. Clearly there is a need to provided synthesis systems with the ability to control the style and emotion of synthetically generated spoken language. However, the only way a speech synthesis system can perform this task is through an appreciation of the style and an interpretation of the content of the text being read. In addition, truly natural readings can only be achieved once the synthesis system has an appreciation of the needs of the audience.
Clearly Speech synthesis should not be seen as an isolated component independent of the language generated, but as part of the larger process of expressing meaning and emotion through spoken language.
From the discussions above, it should be clear that a human - machine interface must do more than simply mimic the attributes of human - human communication, it must be able to deduce intention and respond in a manner appropriate to the request. For a machine to achieve this, it must be "aware" of the limitations of the communication medium and of its own ability to best present information within the constraints of the medium.

The Maya Project

As stated above, Maya's eventual goal is to produce systems which not only mimic the semantic, pragmatic, linguistic, paralinguistic and kinesic attributes of a human being, but provide an enhanced communication medium, which has access to more information than a human, and is able to interpret and present information in a way superior to a human. Such a goal is a long way off, and we must all live in the real world. The Maya project therefore is developing an infrastructure which enables advanced research to co-exist with demonstrable solutions. Clearly no single machine or program can currently hope to accommodate this degree of complexity. The solution is to provide component services, which exist as part of a distributed computer system. These component services, (e.g. speech synthesis, recognition and understanding) exist independently of any particular application, on a variety of computers and are written in a number of different computer languages. Such services communicate through a unifying standard interface language. Currently the system uses the Object Management Group's (OMG) Common Object Request Broker Architecture (CORBA).

A schematic view of the Maya system is shown in figure 1. Maya is composed from a set of co-operating services. These services come in two forms, core services (facilitators) and component services. The facilitating services help to control the flow of information about the system and co-ordinate the behaviour of component services. The number and type of component services available within a particular configuration of Maya will vary depending on the modality of the application being built. As a minimum, the Maya system will contain speech recognition, parsing, dialogue and speech generation services. Each of these component services may in turn be composed of a number of co-operating sub-processes. As an example, the speech generation component within the Maya has at its core, BT's Laureate text-to-speech system (Page and Breen 1996).

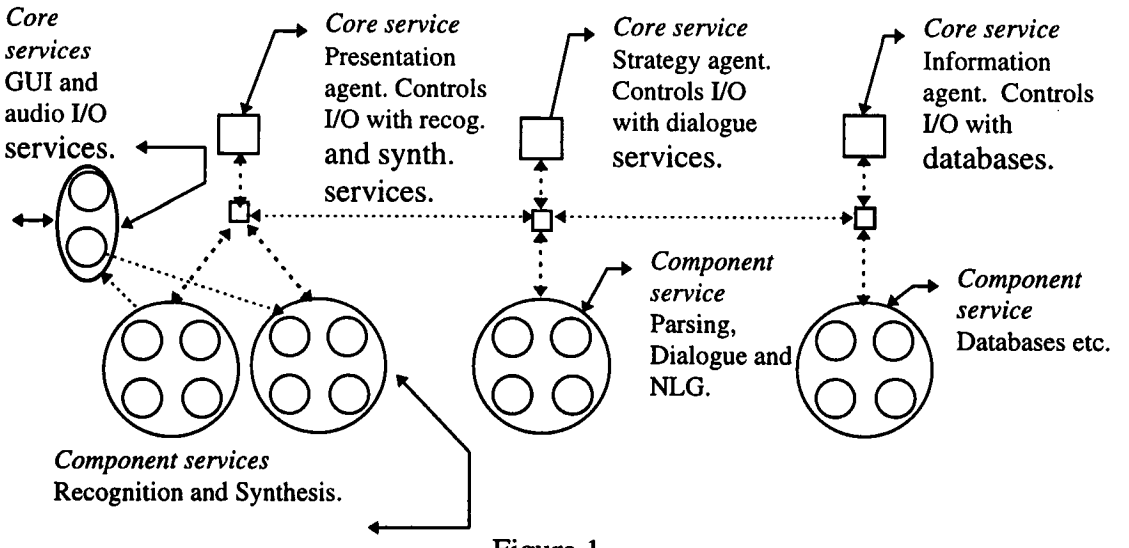

Figure 1.

The future of Speech Generation

The introduction suggested that the requirements of an intelligent human - machine interface could be highlighted through a discussion of three different but related application areas. One Implicit requirement, derived from these discussions, was that an intelligent interface would be composed from a number of different collaborating systems, each system controlling a specific aspect of the interface. The second section provided a brief overview of a computational framework, designed to support research into such an intelligent interface. One significant system (component service) within this framework was the speech generation component.
It is suggested in this paper that the best way to effectively carry out research into such component services, is to consider component specific issues within a complete multi-modal framework, as realized by the Maya system. This section will attempt to provide some justification for this belief through a discussion of the future requirements of the speech generation component.

The vast majority of speech generation systems available today are in fact Text-to-speech (TTS) systems. Which as the name suggests take as their input plain text. Plain text is however, a poor encoder of information, which means that TTS systems are forced to undertake some form of shallow text analysis. The majority of typographical ambiguities found in unrestricted text are handled by a process known as text normalisation, while gross cases may be handled using a limited set of escape sequences. Plain text however does not contain sufficient information to correctly resolve many types of ambiguity. As a result, a large number of the decisions made by the text normalization process are in effect arbitrary. This problem is not restricted to the process of text normalisation, but seriously effects all aspects of the speech synthesis process. To illustrate this point, consider the following example. Imagine that as part of some dialogue, an automated system has generated the question given below:

Did you say fifty pence?

The meaning of this question changes with word emphasis. For example, if emphasis was placed on the word *pence*, the system is asking the user to confirm that the amount was in pence rather than pounds, whereas, if emphasis was place of the word *fifty*, the system is asking the user to confirm that the number was fifty as opposed to some other amount. By default, with only plain text to work with, a text-to-speech systems would typically place the emphasis on "pence".

A TTS system may be asked to convert text from a news article, a poem, a discourse, or even a train time table. Typically, little or no pre-processing will be performed on the text. Is it any wonder then, that the quality of speech produced by such systems is so stilted.

TTS researchers are faced with two choices. Either to increase the level of linguistic analysis conducted on text within the TTS system, or to encourage users to extend the type of information presented to a speech synthesis system.

If the first choice is taken, TTS systems will be forced to balloon in size, effectively taking on the majority of the tasks involved in the interpretation and presentation of information. This is an unpalatable choice for many synthesis researchers, as linguistic analysis, while being a necessary requirement for the production of synthetic speech, is by no means a sufficient requirement. Knowing what to say is not the same as knowing how to say it.

Researchers in speech synthesis still have a lot to learn about the human production system. Unfortunately working with plain text is effectively stifling research into production.

If the second choice is taken, researchers are obliged to investigate effective methods of encoding linguistic and para-linguistic information, which in itself is currently an ill defined and complex task.

The following two sections discuss some of the ways researchers are starting to address the problem of presenting extended information to a speech generation system. The latter section discusses some of the issues facing speech generation systems embedded within a multi-modal framework.

Mark-up

The simple example considered in the last section could easily be handled by attaching a synthesizer dependent emphasis flag to the desired word e.g.

Did you say <Esc>E fifty pence?

The above example demonstrates this using an invented escape sequence. In this example, the escape sequence "<Esc>E" is used to trigger emphasis on the following word. Such escape sequences or flags are commonly used to a greater or lesser degree in most text-to-speech systems. Flags are often used to modify the behavior of the text normalisation and word pronunciation processes, but only a comparatively few systems have flags to modify the emphasis applied to a particular word.
To date, embedded flags of the sort considered above, are not sophisticated enough to provided a realistic mechanism for significantly modifying the behavior of a text-to-speech system. However, a number of researchers (Slott 1996), (Taylor and Isard 1997) or (Sproat, Taylor, Tanenblatt and Isard 1997) are starting to investigate more advanced flag sets. Many of the proposed systems are based on the widely used SGML[2] standard.

An SGML style mark-up language designed specifically for speech synthesis has a number of advantages:

- Marked up text remains readable and carefully designed tags are intuitive
- Tags are easy to apply
- Mark-up systems based on SGML are straight forward to implement and provide a good basis for developing a system independent standard.

However, while text-to-speech mark-up languages have many advantages, they also have disadvantages. Designing an effective and consistent set of tags is problematic. For example, it would be possible and even desirable to produce a tag set which included mark-up for dates, times etc. But such a set would not be significantly different from existing systems. As a result, the only benefit of producing such as set would be the gain obtained through a degree of standardisation. If mark-up is to be used to improve the ability of speech synthesis systems to control style and emotion, a more adventurous set of tags must be devised. Is it possible or desirable to have abstract tags such as "happy", "sad" and "assertive"? Consider the following example.

<assertive> did you say <emphasis> fifty </emphasis> pence? </assertive>

If such a mark-up set was devised, would it be possible to mix levels of abstraction and styles? e.g.

<sad><assertive><noun_phrase><det>the</det><emphasis ><adjective>lead</adjective></emphasis><noun>soldier</ noun></noun_phrase></assertive></sad>

In this example, mark-up tags are used to provide information on style, emotion, focus and syntax! If we assume that each tag is processed independently, it is not hard to envisage a point where one tag conflicts with another tag, producing unpredictable results. It may be the case that the complexity of tag sets is limited by the need to control processing side effects.
One final problem with SGML style mark-up is that it is verbose (as can be seen from the simple sentence given above). Even with carefully designed tags, a comparatively short piece of text could become unworkably large and unreadable.

Speech synthesis with structured data types

It was suggested above, that mark-up was potentially useful, but not sufficient as a means of control. In addition to mark-up, speech synthesis systems should accept typed data, and produce typed data. In this scenario, text is replaced by structured data types. These structures would contain domain, linguistic or para-linguistic information in addition to the data to be processed. The process data contained within the structure may be plain text, marked-up text or in more advanced systems an abstract form of linguistic knowledge.

The type information sent to the speech synthesiser would not only affect the generation process, but the type of output produced by the speech synthesiser as well. For example, if the synthesis system was presented with information relevant to the generation and control of a synthetic persona, then the data produced by the synthesis system would be in a format compatible with the type of information needed to control the computers visual persona. In other words, the specified structured data type would dictate the type of processing done and the type of data produced by the system.

[2] Standard Generalised Mark-up Language.

This paper has repeatedly suggested that advances in the fields of speech synthesis, recognition, understanding and kinesics are best undertaken within a multi-modal framework. In addition, it has been suggested that speech synthesis researchers face a choice between extending the number of text analysis tasks undertaken by the a TTS system or extending the type of information presented to the system. The following example attempts to highlight these issues and hence re-enforce the argument that TTS synthesis systems can best advance when developed within a larger computational framework and support extended input data types.

Consider a case where a synthesis system is presented with an E-mail data type and has within it an E-mail pre-processor. Given that the system now knows that this information has the characteristics of an E-mail, it can apply the appropriate text pre-processing stages needed to correctly interpret and format E-mail. This is shown schematically in figure 2.

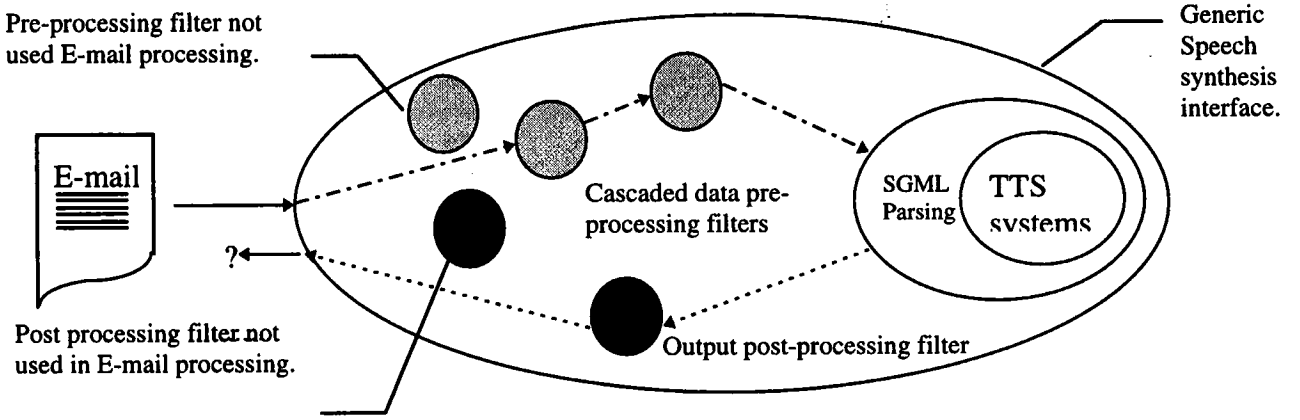

Figure 2.

However, E-mail may contain a bewildering array of encoded data, (e.g. graphics, formatted documents) in addition to plain text. How is a speech generation system to handle this? One solution would be to provide an E-mail pre-processor which was able to detect specific types and replace them with information in a form suitable for speech synthesis. For example, graphical data could be replaced by text simply stating that the E-mail contained an image. However, in a multi-modal environment, a user would expect to have graphical data present on a screen, not lost and replaced by a spoken message! In order for this to happen, a process external to speech synthesis must first interpret the E-mail. In effect a meta-synthesis layer or presentation broker is needed which first determines the number and type of modalities available to the user before committing any data to a particular presentation process. Such a broker is very similar in principle to the facilitating services used within Maya.

Hence through a process of requirement specification for one modality (synthesis), the need for a multi-modal framework such as Maya re-appears.

Summary

This paper has presented one possible approach to the development of an intelligent human - machine interface. It has suggests that research into the development of such an interface will substantially benefit all related areas of study such as speech recognition, synthesis and kinesics. In addition, it has been suggested that the next generation of synthesis systems will inevitably take more account of the type of information being presented to them; that the interfaces to such systems will become more generic, and that the type of processing conducted as part of the synthesis process will become more diffuse and data orientated.

Acknowledgements

The ideas expressed in this paper are the product of many interesting discussions with colleges at BT Labs. In particular I would like to thank the members of the Maya project: Simon Downey, Maria Fernández, Rumman Gaffur, Ed Kaneen and Steve Minnis for their contributions.

References

Page J.H., Breen A.P. 1996. The Laureate Text-to-Speech System - Architecture and Applications. BT Technol. J.:14:1.

Edgington. M., Lowry. A., Jackson. P., Breen A. P., Minnis.

S. 1996. Overview of Current Text-to-Speech Techniques: Part I - II, BT Technol. J.:14:1.
Pawlewski. M. et al. 1996. Advances in Telephony-Based Speech Recognition", BT Technol. J.:14:1.
Wyard. P. et al. 1996. Spoken Language Systems - Beyond Prompt and Response, BT Technol. J.:14:1.
Breen. A. P. 1996. The face of talking Machines in a Multi-media World. British Telecommunications Engineering:15..
Breen. A. P., Bowers. E., Welsh. W.: "An Investigation into the Generation of mouth Shapes for a Talking Head", ICSLP '96..
Slott, J. M. 1996. A Generalised Platform and Markup Language for Text to Speech Synthesis. Ph.D. diss., Dept Electrical Engineering and Computer Science, MIT.
Taylor P, Isard A. 1997. "SSML: A Speech Synthesis Markup Language", Speech Communication:21:123-133.
Sproat. R., Taylor. P., Tanenblatt. M., Isard. A.1997. A Markup Language for Text-to-Speech Synthesis. Eurospeech '97: 4.