Universidade de Coimbra
Faculdade de Ciências e Tecnologias
Departamento de Engenharia Informática

Reconhecimento de voz

Sistemas Multimédia
2001/2002

Paulo José dos Santos Guilhoto
Susana Patrícia Costa de Sousa Rosa

Departamento de Engenharia Informática
Faculdade de Ciências e Tecnologia da
Universidade de Coimbra

Trabalho de síntese realizado
para a disciplina de
Sistemas Multimédia
no âmbito da
Licenciatura em Engenharia Informática

Este trabalho está disponível em
http://student.dei.uc.pt/~srosa/voz

Paulo José dos Santos Guilhoto
guilhoto@student.dei.uc.pt

Susana Patrícia Costa de Sousa Rosa
srosa@student.dei.uc.pt

Novembro de 2001

Índice

Introdução

Há uns anos atrás, o reconhecimento de voz era considerado por muitos como sendo apenas obra de ficção científica. Contudo, desde o final da década de 1950, instituições de pesquisa já estudavam meios de fazer com que a voz fosse processada pelo computador. Actualmente, o reconhecimento de voz tornou-se real e passou a ser um dos tópicos mais férteis no seio da investigação. De facto, muitas aplicações estão actualmente a ser desenvolvidas ou a surgir no mercado. Um dos exemplos mais vulgares, com o qual a maioria de nós certamente já teve contacto, é a tecnologia de chamada por voz, presente em alguns telemóveis existentes no mercado.

Este trabalho pretende assim, representar o "estado da arte", pelo que, para além de temas como o funcionamento do reconhecimento de voz, a complexidade dos aspectos técnicos envolvidos e a evolução dessas técnicas, também irão ser explorados temas como os produtos actualmente existentes, suas características e potencialidades, e as aplicações que estão a tirar partido desta tecnologia.

O que é o reconhecimento de voz?

Frequentemente, a expressão "reconhecimento de voz" é utilizada com vários sentidos, que, na verdade, referem-se a tecnologias distintas. O processamento da voz pode ser aplicado em quatro áreas principais: comandos por voz, fala natural, síntese de voz e autenticação de voz. Cada uma destas é descrita a seguir:

- O reconhecimento de palavras (utilizado nos comandos de voz) caracteriza-se por processar apenas um pequeno trecho de fala, de modo a identificar que tipo de acção o sistema deve tomar. Este processamento torna-se simplificado, uma vez que o sistema já sabe de antemão quais os comandos disponíveis para o utilizador. Este é o caso de centrais de atendimento telefónico, onde o utilizador pode usar a voz em vez de pressionar botões.
- O reconhecimento de fala natural (ou fala contínua) envolve uma ou mais frases, isto é, várias palavras que tenham um sentido semântico. A fala reconhecida é então convertida em texto. O tipo de aplicação mais comum para essa tecnologia é o ditado de documentos, para uso em processadores de texto, escrita de e-mails, etc.
- A síntese de voz é o processo contrário ao do reconhecimento da fala. O sintetizador recebe um texto na forma digital e transforma-o em ondas sonoras, ou em outras palavras, fazendo uma leitura em voz alta. Um programa de síntese de voz é útil nas situações em que o utilizador não pode desviar a atenção para ler algo ou não tem acesso ao texto escrito, seja porque a informação está distante ou porque o utilizador tem alguma deficiência visual.
- A autenticação baseia-se, por sua vez, no facto de que a voz é única para cada pessoa e pode ser utilizada para identificar alguém. Os sistemas de autenticação podem ser aplicados para permitir o acesso de uma pessoa a uma determinada função.

Sendo assim, e dado que a temática deste trabalho é apenas o reconhecimento de voz, o leitor deverá ter em atenção que os capítulos seguintes apenas irão cobrir a matéria referente aos dois primeiros conceitos acima referidos.

Como é que funciona?

O reconhecimento de voz evoluiu bastante ao longo dos últimos anos. Inicialmente, o reconhecimento processava-se apenas em modo discreto, isto é, o utilizador tinha que fazer uma pausa entre cada palavra ditada. Actualmente, o utilizador já tem a possibilidade de efectuar ditados contínuos ao computador. Para além disso, o reconhecimento também se tornou mais inteligente, uma vez que possui um conjunto de regras gramaticais incorporadas, permitindo assim perceber melhor o que está a ser ditado.

O reconhecimento de voz usa diferentes técnicas para reconhecer a voz humana. Funciona assim como uma *"pipeline"* que transforma os sinais áudio digitais provenientes da placa de som em fala reconhecida. Estes sinais passam por diversas etapas, ao longo das quais são aplicados método matemáticos e estatísticos de forma a tentar compreender o que está a ser ditado.

Transformação do PCM

Quando o utilizador fornece um comando de voz pelo microfone, este é transmitido para a placa de som, sendo o sinal analógico amostrado 16.000 vezes por segundo. De seguida, este é transformado para digital através de uma técnica chamada *Pulse Code Modulation (PCM)*. Esta forma digital do sinal não é nem mais nem menos do que uma *stream* de amplitudes representando o sinal analógico. No entanto, o software de reconhecimento de voz não pode trabalhar directamente com base nesta *stream*, dado que é muito complicado procurar padrões que possam ser relacionadas com o que está actualmente a ser ditado. É neste sentido que esta é transformada num conjunto de bandas de frequência discreta, usando uma técnica chamada *Windowed Fast Fourier Transform (FFT)*. Esta consiste numa amostragem do sinal áudio em cada centésimo de segundo, mas desta vez no domínio da frequência. Assim, a *stream* de entrada é agora representada por um conjunto de bandas de frequência discretas, podendo ser facilmente identificadas as componentes de frequência de um som. A partir destas componentes, é possível aproximar-nos do que o ouvido humano ouve.

O próximo passo envolve o reconhecimento destas bandas de frequência. Para isso, o software de reconhecimento de voz possui uma base de dados de milhares de frequências ou "fonemas". Um fonema é a mais pequena unidade de fala de um idioma ou dialecto. A dicção de fonemas é diferente entre si, tal que, ao substituir um fonema numa palavra, esta passa a ter um significado completamente diferente. Por exemplo, se substituirmos o fonema "b" na palavra "bato" pelo fonema "m", o significado será alterado para "mato". A base de dados de fonemas é usada para comparar e identificar as banda de frequência áudio que foram amostradas. Se, por exemplo, a frequência de entrada tem um som igual ao "t", o software irá tentar compará-lo com o respectivo fonema na base de dados. Ao encontrar o fonema correspondente, é atribuído ao sinal de entrada o número identificador do fonema na base de dados, também chamado de *"feature number".*

Graças às transformadas de Fourier e à base de dados de fonemas, tornou-se assim possível passar de um vector PCM com 16.000 entradas para um vector com apenas 100 entradas (por segundo), transformando o processo de reconhecimento em tempo real fazível.

Reconhecimento de padrões acústicos

Aparentemente, o processo é simples. A cada banda de frequência seria associado o seu fonema correspondente. O software iria de seguida juntar os fonemas em palavras, e o computador passaria a compreender a voz humana. Aparentemente. No entanto, o processo é muito mais complicado quando começamos a olhar para ele com mais detalhe. Podem existir tantas variações no som causado pelo modo como as palavras são pronunciadas, que é quase impossível encontrar a entrada na base de dados correspondente ao som. A juntar a isto, diferentes pessoas pronunciam a mesma palavra de forma diferente. Para complicar ainda mais a situação, o ambiente também adiciona a sua componente de ruído: numa situação em que o utilizador está na secretária do seu escritório, com o consequente ambiente ruidoso que daí advém, o reconhecedor de voz pode apresentar resultados diferentes de uma situação em que o utilizador está numa sala silenciosa com um microfone de alta qualidade. Sem esquecer que o próprio som de um fonema varia conforme os fonemas que o rodeiam e que o som produzido por um fonema altera-se desde o início da sua pronúncia até ao seu fim. Sendo assim, o software tem que usar técnicas complexas para aproximar o som de entrada e descobrir quais os fonemas que estão envolvidos.

O ruído de fundo e os problemas de variação são solucionados com recurso a métodos estatísticos, ao permitirmos que um *feature number* seja usado por mais do que um fonema. Isto apenas é possível porque a duração de um fonema é longa se comparada com a frequência de amostragem de um centésimo de segundo. Isto quer dizer que enquanto um fonema está a ser pronunciado, estão a ser atribuídos vários *feature numbers*. Admitindo que o software já foi treinado, o que quer dizer que já foram estabelecidas as probabilidades de ocorrência de um determinado *feature number* num fonema, resta-nos calcular a probabilidade do conjunto de *feature numbers* registados ser realmente o fonema *(ver exemplo na figura 1).*

Durante o processo de treino, o software registou as seguintes estatísticas, para os fonemas "b" e "m", respectivamente:

- No fonema "b", a probabilidade de aparecer o som associado ao *feature number* #52 é de 55%, 30% para o #189 e 15% para o #53.
- No fonema "m", a probabilidade de aparecer o som associado ao *feature number* #52 é de 10%, 10% para o #189 e 80% para o #53.

Vamos usar a análise dos dados obtidos ao longo do treino durante o processo de reconhecimento. Assumindo que foram "ouvidos" 6 *feature numbers* (#52, #52, #189, #53, #52, #52) durante o reconhecimento, vamos calcular a probabilidade de o conjunto ser o fonema "b" ou "m", respectivamente:

- 55% * 55% * 30% * 15% * 55% * 55% = 0.41%
- 10% * 10% * 10% * 80% * 10% * 10% = 0.0008%

Verifica-se assim que o fonema pronunciado foi o "b".

Figura 1 – Exemplo de como é reconhecido um fonema isolado

O reconhecedor de voz também necessita de saber quando é que um fonema acaba e outro começa. Para isto é usada uma técnica matemática denominada *"Hidden Markov Models" (HMM).* Admitindo que o reconhecedor de voz registou os *feature numbers* correspondestes a dois fonemas consecutivos de uma palavra, torna-se complicado, a olho humano, distinguir onde começam e acabam os fonemas, sobretudo se estes dois fonemas tiverem algum *feature number* em comum *(ver figura 2).* É aqui que entra a técnica das HMM's, que

consiste na explosão combinatória das possibilidades de qualquer fonema ser seguido de qualquer outro fonema, ligados por meio de transições com pesos associados, até que se torna possível distinguir com clareza onde começa e acaba o fonema. Contudo, este método não é viável se não forem aplicadas limitações, como se irá ver de seguida, dadas as gigantescas quantidades de memória necessárias.

Supondo que o fonema "a", com probabilidades da ocorrência de *feature numbers* de 75% para #82, 15% para #98 e 10% para #52, surge depois do fonema "b" numa determinada palavra, gerando assim a seguinte sequência de *feature numbers:* #52, #52, #189, #53, #52, #52, #82, #52, #82, etc.

Tomando em atenção que o *feature number* #52 aparece nos fonemas "b" e "a", torna-se difícil distinguir um do outro. Apenas podemos afirmar com certeza que o fonema "b" antecede o fonema "a", dada a localização dos *feature numbers*.

Figura 2 – Exemplo de como distinguir fonemas consecutivos

Também é de referir a importância da existência de um fonema "silencioso", também ele caracterizado por *feature numbers*, de forma a identificar pausas nas palavras ditadas.

Ainda existe um outro motivo de preocupação: o som de um fonema depende geralmente do fonema anterior e posterior. O software de reconhecimento de voz consegue superar este problema criando "tri-fonemas", que são fonemas no contexto de fonemas consecutivos. Dado que a língua portuguesa tem 28 fonemas, existem 28*28*28 = 21.952 tri-fonemas, o que constitui um número demasiado elevado em termos de esforço de computação, daí que os tri-fonemas que tenham um som semelhante sejam agrupados, acabando por se transformar num só.

Muitos outros problemas subsistem, tal como a evolução do som de um fonema não ser constante, o próprio reconhecedor de voz não saber quando é que o utilizador vai começar a falar. Contudo, não nos vamos debruçar sobre eles.

Modelos linguísticos

O software de reconhecimento de voz é agora capaz de identificar fonemas que são pronunciados. O passo seguinte consiste em reconhecer palavras, comparando as combinações de fonemas com as palavras contidas no dicionário utilizado pelo programa. Contudo, isto não é assim tão simples: o utilizador pode enganar-se ao pronunciar uma palavra que não faz parte do dicionário; o software de reconhecimento de voz pode enganar-se ao reconhecer uma palavra; ainda não se sabe onde começa uma palavra e acaba a outra; o processamento de voz continua incomportável quer a nível de exigências de CPU como de memória. É por isto tudo que se torna necessário para o reconhecedor de voz restringir as possibilidades do que está a ser ditado, através do recurso a modelos linguísticos e gramáticas adequadas.

Se o sistema for programado para fazer o reconhecimento de comandos, o modelo linguístico é um pouco mais simples do que se fosse para reconhecimento da linguagem natural. Nesse caso, o dicionário contém todas as formas possíveis de se pronunciar cada comando registado no sistema. Para a fala contínua, é preciso que o sistema utilize um dicionário com todas as palavras da língua ou pelo menos com todas as palavras usadas com maior frequência.

As condicionantes não ficam por aqui. O dicionário deve também registar as classes gramaticais de cada vocábulo. Para além disso, é preciso construir um modelo gramatical com as construções possíveis e a probabilidade de ocorrência de cada tipo de construção. Com a ajuda destas informações, o software de reconhecimento de voz tem condições para concluir que as frases "foi à três anos" e "foi a três anos" não estão correctas, mas sim "foi há três anos". Mais uma vez, para o caso do ditado discreto (reconhecimento de comandos), as gramáticas são muito mais simples, uma vez que o sistema sabe com antecedência quais as palavras que está à espera. *(Para mais informações sobre o reconhecimento de linguagem natural, ver a secção dedicada ao NLU – Natural Language Understanding).*

Antes disto, o software deve processar os fonemas para identificar correctamente o agrupamento de palavras. Os fonemas contidos em "foi há três anos" sugerem termos como *"foia"* e *"trêzanos"*, que não estão no dicionário de vocábulos conhecidos, logo, são descartados. Por outro lado, ao ouvir a palavra "comunicação", o reconhecedor de voz terá que decidir se ouviu "comunica acção" ou um único termo. O modelo gramatical vai indicar qual é a construção mais plausível. Os programas mais recentemente implementados fazem a análise de toda a frase, para ampliar a precisão do reconhecimento, tomando assim vantagem do facto da linguagem apresentar uma estrutura. Po exemplo, supondo que o reconhecedor de voz tem dúvidas entre a escolha das palavras "hora" e "ora", mas sabe que a palavra anterior é "uma", então está na possibilidade de efectuar a escolha certa porque sabe que a sequência "uma hora" faz mais sentido do que a sequência "uma ora". A técnica que está por detrás disto usa "trigramas", baseados em modelos estatísticos, que calculam a probabilidade de uma determinada sequência de palavras ocorrer.

A maioria dos pacotes de reconhecimento da fala vem com dicionários que contêm cerca de 150 mil palavras do português. Os sistemas na língua inglesa também trabalham com aproximadamente a mesma quantidade de termos no dicionário.

Treino do software

Apesar de todas as técnicas mencionadas acima, para que serve uma grande base de dados de fonemas se estes não correspondem à nossa voz ou pronúncia? Por exemplo, o que é que acontece quando uma base de dados desenvolvida no Brasil é colocada à venda em Portugal?

Este é um verdadeiro problema, e a única solução é possibilitar ao utilizador "treinar" os modelos acústicos. No caso de um sistema mono-utilizador, basta fornecer um texto pré-determinado ao utilizador para este ditar. No caso de sistemas multi-utilizador, por exemplo no caso de uma central telefónica, não é viável pedir a cada utilizador falar durante 15 minutos para treinar o reconhecedor de voz. A solução para este caso é efectuar um treino conjunto do sistema por várias pessoas, sendo os pesos das transições das HMM's ajustado de acordo com a média dos modelos, de forma a tornar o sistema capaz de reconhecer o maior número de utilizadores possível.

Natural Language Understanding (NLU)

Para um computador, um conjunto de palavras não possui nenhum significado intrínseco. Pegar nos resultados do reconhecimento de voz e extrair informação útil sobre a qual o computador pode agir, não é tarefa fácil. Dado que, mesmo as áreas do cérebro humano que processam a linguagem ainda são largamente desconhecidas, os primeiros linguistas aplicados à área da computação tiveram que começar do nada.

A área de *Natural Language Understanding (NLU)* tem vindo a ser desenvolvida na sua maioria por companhias telefónicas e organizações ligadas à área da internet e redes IP. A razão para tal é sobretudo a fraca qualidade do sinal áudio transmitido nas linhas telefónicas, o que dificulta em muito a aplicação de centrais telefónicas automáticas. É aqui que entra o NLU, usando 75% de reconhecimento de voz certificado para o transformar em 85% ou 90% graças à análise contextual.

Para este propósito, a análise da estrutura gramatical da fala tem pouco interesse. Por isso, e no âmbito deste trabalho, iremos apenas fazer uma análise muito superficial destas técnicas.

Fundamentos

De um ponto de vista mais informático, o NLU funciona ao longo de um processo muito similar com o processo de compilação de um programa, só que ao contrário. Isto é, em vez de adoptar uma metodologia *"top-down",* em que a compilação pára sempre que detectar a falta de um ponto-e-vírgula (p.e), opta-se por uma abordagem *"bottom-up",* tipicamente pessimista, que assume de antemão que algo estará errado no seu *input* e prepara-se para ter o melhor desempenho possível.

Se tivessemos optado por aplicar a abordagem *"top-down parsing"* à linguagem natural, muitos problemas iriam surgir, nomeadamente:

- Todas as linguagens naturais coexistem com a ambiguidade, excepções às regras gramaticais, e inconsistência. A tarefa de definir uma gramática de *parsing* para cada linguagem torna-se praticamente impossível.
- Não se pode esperar que todas as pessoas falem de um modo pré-definido, porque isso tornaria o sistema demasiadamente frágil.

O NLU utiliza uma aproximação mais robusta e objectiva perante este problema:

- Uma aplicação só deverá definir uma gramática para o menor subconjunto da linguagem natural apropriada ao seu domínio. Desta forma, muita da ambiguidade pode ser colocada de parte.
- Em vez de usar o *"top-down parsing"*, o NLU irá utilizar o *"bottom-up parsing"*, tentando interpretar pequenos fragmentos de palavras soltas e combinando-as de modo a obter uma "imagem" global do que se está a tentar dizer.

Por exemplo, na frase "Eu quero ir para Lisboa esta tarde", o NLU interpreta as palavras "esta tarde" como sendo a descrição de um instante no tempo, a palavra "Lisboa" como sendo o nome de uma cidade, e a palavra "para" seguida do nome de uma cidade como sendo a identificação de um destino. O resto da frase é descartado porque já temos elementos para extrair o sentido do que foi pronunciado.

Se algo correr mal, nomeadamente na entrada de voz, e o sistema reconhece a fala "Elu queru iri para Lisboa espa tarde", os fragmentos mais importantes "para Lisboa" e "tarde" continuam a ser bem compreendidos (possivelmente com alguma incerteza, porque não sabemos a que dia é que o utilizador se está a referir) e 80-100% da fala é bem interpretada enquanto conteúdo, embora apenas 43% das palavras tenham sido exactamente identificadas.

Gramáticas

Ambos métodos de *parsing* requerem um conjunto de regras não ambíguas, completas e objectivas de modo a serem codificadas numa gramática. Algumas décadas atrás, o linguista e político Noam Chomsky definiu diversas classes de linguagem e as gramáticas que as suportam. Por exemplo, as "expressões regulares" podem ser processadas de um modo linear, uma palavra de cada vez, enquanto que as gramáticas "sem contexto" que definem linguagens de programação são recursivas por natureza. Todo o sistema NLU é baseado nestes ideais.

Por exemplo, passemos a considerar uma gramática muito simples, no âmbito de um sistema de controlo de viagens:
<CIDADE> -> Lisboa
<CIDADE> -> Funchal
<CIDADE> -> Porto
<CHEGADA> -> [indo] para <CIDADE>
<PARTIDA> -> [a partir] de <CIDADE>
<VIAGEM> -> <PARTIDA> <CHEGADA>
<VIAGEM> -> <CHEGADA> <PARTIDA>

Os termos dentro de *'<>'* são as variáveis. As palavras dentro de *'[]'* são opcionais. As regras da *'<VIAGEM>'* especificam que uma viagem é definida seja com a cidade de partida seguida pela de destino seja pelo contrário. A regra *'<CHEGADA>'* especifica que podemos definir um destino quer através de *'para'* seguido de uma cidade, quer dizendo *'indo para'* seguido de uma cidade.
Desta forma, o sistema é capaz de compreender frases como "Para Funchal a partir de Lisboa" ou "A partir de Funchal, para Porto".

Mesmo que a entrada de voz fosse destorcida e esta resultasse em "Para bsdvger de Lisboa", o sistema seria capaz de conseguir identificar a cidade de origem e fazer apenas uma pergunta em relação ao destino, evitando assim perguntas tipo "Por favor, repita!".

Entraves à implementação

Quando as pessoas falam, são geralmente bastante descuidadas no que toca à formulação, produzem erros fonéticos, como já tivemos oportunidade de ver na secção anterior. Muitas fontes de erro são introduzidas nos dados de entrada do NLU:

- As pessoas não falam a um ritmo constante, há hesitações, pausas e interjeições nos momentos mais inoportunos.
- Frequentemente, quando uma pessoa fala, muda de opinião a meio de uma frase. Por exemplo, um ouvinte humano percebe que "não, desculpe, eu quis dizer sexta-feira" se refere a uma referência anterior, contudo para um computador já não é tão perceptível.

- Por vezes, as pessoas duplicam frases ou dizem frases sem nexo sequencial, etc. Um exemplo é o caso em que ouvimos uma entrevista na TV, apercebemo-nos que a discussão, diálogo ocorre naturalmente, mas, se tivermos acesso à mesma entrevista por escrito, depressa nos iremos perceber que há coisas que não têm nexo.
- As pessoas assumem que o ouvinte é capaz de interligar os pronomes e locuções como "o", "ele" e "aquele" aos conceitos respectivos. Mais uma vez, isto é extremamente complicado para um computador.

Como escrever uma gramática

Para tornar o sistema o mais robusto possível, os linguistas que escrevem as suas próprias gramáticas terão que criar regras que aceitem os erros mais comuns de entrada como sendo válidos, num determinado domínio. Também se espera que as regras de mais alto nível (tal como *'<viagem>',* no exemplo anterior) suportem formulações suficientes para manipular a maioria dos casos (com suporte para aceitar lixo, se for caso disso, como dados válidos).

Quando possível, os linguistas devem tentar limitar os seus domínios em expressões regulares, que podem ser analisadas com complexidade O(n), enquanto gramáticas "sem contexto" requerem $O(n^3)$. Infelizmente, isto é raro acontecer excepto em alguns casos.

Contudo, isto não é assim tão óbvio. Cada vez que adicionamos uma regra para cobrir um novo caso, esta corre o risco de introduzir alguma ambiguidade e colocar em risco a funcionalidade do sistema. O crescimento na complexidade é exponencial; uma vez que o sistema atinja uma centena de regras, torna-se quase impossível introduzir o suporte para uma nova frase sem comprometer o já comprovado funcionamento de dez outras.

Como consequência, alguns investigadores estão a começar a experimentar a aprendizagem automática de regras gramaticais. Por exemplo, no nosso caso acima, uma versão inteligente do sistema deveria "descobrir" que o nome de uma cidade a seguir à palavra *'para'* é geralmente indicativo de um destino, desde que tenhamos treinado o sistema com vários exemplos onde é explícito que *'para Lisboa'* indica que queremos ir para lá. Por enquanto, os resultados são interessantes, mas não atingiram ainda qualidade suficiente para passarem a ser disponibilizados comercialmente.

Aplicações

Como se pode imaginar, o reconhecimento de voz apresenta ganhos significativos no processo de transcrever documentos para processadores de texto. Tirando vantagem do facto de que o ser humano ser capaz de ditar sete vezes mais rápido do que escrever, conseguem-se, nalguns casos, ganhos de produtividade de 60%. Contudo, as vantagens da tecnologia de reconhecimento de voz não estão limitadas aos ambientes de escritório, como se poderia pensar. Quando se está em viagem, por exemplo, pode-se utilizar um gravador digital para ditar memorandos, mensagens de correio electrónico, ou notas de reunião que podem ser depois transcritas para vários programas. E uma vez que os gravadores estão a ficar cada vez mais pequenos, quase nem damos conta quando transportamos um no bolso.

As possíveis aplicações não ficam por aqui. Um exemplo é a indústria da saúde, que se depara actualmente com factores críticos de sucesso como a redução das despesas e aumento da eficiência. O reconhecimento de voz pode aqui ajudar as equipas médicas eliminando a necessidade de transcrever manualmente os relatórios médicos, bastando para isso o uso de um pequeno aparelho portátil, parecido com um gravador, enquanto se procede ao diagnóstico dos pacientes.
Do mesmo modo, numa seguradora, os ganhos podem ser imensos. Imaginemos que um inspector de seguros está a avaliar um sinistro no terreno. Ao usar o reconhecimento de voz, é possível que, ao mesmo tempo que este está a descrever o caso, os dados do processo comecem a chegar à seguradora, permitindo que o cliente veja o seu caso resolvido muito mais rapidamente.

Um exemplo típico de aplicação do reconhecimento de voz são as centrais telefónicas automáticas, nas quais o utilizador pode dizer naturalmente que deseja falar com uma determinada pessoa e o sistema repassa a chamada para o posto correspondente. Em caso de dúvida, por exemplo no caso em que se pede para falar com uma determinada pessoa e existem mais pessoas com o mesmo nome na empresa/organização, o sistema interage com o cliente dando alternativas para que seja feita a escolha.

Ainda assim, o reconhecimento de voz está a começar a aparecer onde menos se esperava. É exemplo disso a área dos jogos, onde se prevê que num futuro próximo comecem a surgir jogos totalmente comandados por voz. De início, supõe-se que as técnicas de reconhecimento comecem a ser aplicadas em jogos com pouca interacção e onde, sobretudo, haja poucos sons ambientes, uma vez que estes iriam dificultar em muito o processo de reconhecimento de voz. Os jogos de estratégia são os mais fortes candidatos a serem alvo da aplicação destas técnicas.

Soluções existentes

Dragon Systems, Lernout & Hauspie (L&H), IBM e Philips são as maiores empresas que actuam neste mercado. As duas primeiras fazem parte do mesmo grupo, desde que a belga L&H (http://www.lhsl.com) comprou a norte-americana Dragon (http://www.dragonsys.com) em 2000. Há pouco mais de um ano, a Dragon detinha cerca de 60% do mercado de sistemas de reconhecimento de voz, no mundo, com a IBM e L&H disputando cerradamente o segundo lugar. A Dragon, no entanto, nunca investiu no desenvolvimento do reconhecimento da língua portuguesa, e a L&H não fez mais do que incluir o nosso idioma entre as opções de dicionário, no tradutor Power Translator. A IBM cedo demonstrou interesse em criar uma versão para português do seu produto IBM ViaVoice (www.ibm.com/speech), embora só numa versão orientada para o português brasileiro. A Philips comercializava ainda à bem pouco tempo o produto FreeSpeech 2000 (http://www.speech.philips.com), orientado para utilizadores em ambientes doméstico e de escritório. Contudo, e talvez porque as perspectivas de futuro não eram muito prometedoras, a Philips decidiu enveredar no desenvolvimento de produtos exclusivamente para profissionais.

De um modo geral, todos os produtos, para além de permitirem que o utilizador dite os seus textos num processador de texto do tipo Microsoft Word, permitem criar macros para a introdução de texto ou ditado com qualquer aplicação Windows. De igual modo, todos eles (excepto o NaturallySpeaking) permitem comandos de voz fiáveis para a inicialização de programas e para correr macros de teclado/rato previamente criadas. Paralelamente, todos os produtos disponibilizam algumas funcionalidades Web com capacidade de voz.

O Dragon NaturallySpeaking caracteriza-se pela sua facilidade de utilização, incluindo a formatação e a navegação na Web através de comandos de voz. Além disso, como conta com uma optimização melhorada para novos processadores e com melhoramentos a nível do reconhecimento de voz e de comandos, o NaturallySpeaking disponibiliza uma exactidão impressionante (cerca de 95%). Este programa só é prejudicado pelo facto de apresentar um suporte limitado em termos de linguagem natural com outras aplicações além do Microsoft Word. Em sua defesa, a sua funcionalidade NaturalWeb permite inserir facilmente URLs e seleccionar ligações nas páginas através da voz. O NaturallySpeaking conta ainda com atalhos de formatação e de ditado intuitivos, bastando dizer "cap", por exemplo, ou "all cap" para definir maiúsculas.

Com uma taxa de exactidão de 94%, o L&H Voice Xpress disponibiliza tudo o que de básico se pode esperar de um programa de reconhecimento de voz. O ponto forte do programa reside nos seus comandos intuitivos de linguagem natural para o Microsoft Word, para o Excel e para o processador de texto simples do Voice Xpress. No entanto, as suas funcionalidades de correcção são limitadas, faltando-lhe, por exemplo, a reprodução áudio.

O IBM ViaVoice, é o mais avançado produto de reconhecimento de voz da IBM e disponibiliza bastantes funcionalidades, tanto para ditados como para o controlo de aplicações. O ViaVoice é ainda capaz de guardar o áudio das últimas mil palavras ditadas para reprodução, algo que nos pode ajudar a corrigir enganos. Esta funcionalidade é essencial se os utilizadores quiserem delegar a edição a outra pessoa. Com uma exactidão de 98%, este produto é actualmente o mais fiável do mercado.

Histórico do desenvolvimento de software de reconhecimento de voz

Final da década de 1950	Primeiras pesquisas tecnológicas para o reconhecimento de voz.
1964	IBM apresenta um sintetizador de voz para a fala de dígitos.
1978	A Texas Instruments lançou o primeiro chip dedicado à síntese de voz.
1993	IBM lança o primeiro software comercial para reconhecimento de voz, o IBM Personal Dictation System, para OS/2.
1993	Apple apresenta um conjunto de rotinas para Mac, para reconhecimento e síntese de voz.
1993	Universidade Federal do Rio de Janeiro desenvolve Dosvox, com síntese de voz em português, para deficientes visuais usarem PC's com DOS.
1994	Dragon Systems apresenta o Dragon Dictate para ditados.
1996	IBM apresenta o MedSpeak/Radiology, primeiro produto para reconhecimento da fala contínua em tempo real.
1996	OS/2 Warp é o primeiro sistema a embutir comandos de voz.
1997	Dragon Systems lança o primeiro programa de uso geral para reconhecimento da fala contínua em inglês.
1997	IBM lança o ViaVoice, para fala contínua.
1998	IBM lança ViaVoice em português.
1998	MicroPower lança DeltaTalk, sintetizador de voz em português.
1999	Philips lança FreeSpeech 2000, com reconhecimento da fala em português.
1999	Lotus e Corel acrescentam recursos de voz a seus pacotes de aplicativos.
2000	L&H adquire Dragon Systems e lança L&H Dragon NaturallySpeaking 5.0.
2001	Telemar lança Vocall, primeiro serviço de voz aberto ao público, com síntese e reconhecimento da fala, para e-mails e agenda.
2001	L&H é colocada à venda, por se encontrar em grave crise financeira.
2001	Microsoft acrescenta recursos de voz (para ditados e comandos) ao Office XP. Na versão em português, essa facilidade está ausente.

Fontes

- Philips Speech Processing
  www.speech.philips.com
- Indiatimes InfoTech – How speech recognition works
  http://www.indiatimes.com/infotech/help/software/voicereg.html
- Revista PC World – Biometria: Reconhecimento de voz
  http://pcworld.terra.com.br/pcw/testes/tecno_hard/0030.html
- Revista PC World – Ouça e seja ouvido! O computador já conversa com você
  http://pcworld.terra.com.br/pcw/testes/tecno_hard/0040.html
- GIGNews.com - Speech Processing for Games
  http://www.gignews.com/fdlspeech1.htm
- Samsung Electronics – Research Center in Russia
  http://research.samsung.ru/surveys/1999-11/05-1.html
- ZDNet Portugal – PC Magazine
  http://www.zdnet.pt/pcmagazine/analises/software/0002/voz1.shtml
- Guia do PC
  http://www.guiadopc.com.br/testes/viavoice.htm

- http://ciips.ee.uwa.edu.au/~roberto/research/speech/local/howsr.htm