Prospectiva (Frutal-MG).

Aplicação solução baseada em voz para portadores de necessidades especiais.

Rogério Marchi Marano.

Rogério Marchi Marano

APLICAÇÃO BASEADA EM VOZ PARA PORTADORES DE NECESSIDADES ESPECIAIS

COLEÇÃO
Produzir Cidadania
EDITORA
PROSPECTIVA

Rogério Marchi Marano

Aplicação solução baseada em voz para portadores de necessidades especiais

Frutal-MG
Editora Prospectiva
2016

Capa: Jéssica Caetano
Foto de capa: http://imagem.band.com.br/zoom/f_318172.jpg
Revisão: O autor
Edição: Editora Prospectiva
Editor: Otávio Luiz Machado
Assistente de edição: Jéssica Caetano
Conselho Editorial: Antenor Rodrigues Barbosa Jr, Otávio Luiz Machado e Rodrigo Portari.
Contato da editora: editoraprospectiva@gmail.com
Página: https://www.facebook.com/editoraprospectiva/
Telefone: (34) 99777-3102
Correspondência: Caixa Postal 25 – 38200-000 Frutal-MG

MARANO, Rogério Marchi.

Aplicação solução baseada em voz para portadores de necessidades especiais. Frutal: Prospectiva, 2016.

ISBN: 978-85-5864-031-2

1. Reconhecimento de voz. 2. Software. 3. Portadores de necessidades especiais. I. Marano, Rogério Marchi. II. Universidade do Estado de Minas Gerais. III. Título.

AGRADECIMENTO

A Deus, que me deu forças para chegar até aqui.

À minha mãe, que lutou junto a mim para que eu realizasse este projeto.

Ao professor Sylvio, pela paciência, orientando-me de forma eficiente.

DEDICATÓRIA

A todos os meus familiares e amigos que acreditaram em minha capacidade de realizar este projeto.

SUMÁRIO

LISTA DE SIGLAS E ABREVIATURAS

API - Application Programming Interface
CEFET-SC – Centro de Educação Tecnológica de Santa Catarina
CMN - Cepstral Mean Normalization
CMU - Carnegie Mellon University
DCT - discrete cosine transform
FFT - Fast Fourier Transform
FST - Free Subliminal Text
HMM - Hiddem Markov Models
HP - Hewlett Packard
IBGE - Instituto Brasileiro de Geografia e Estatística
LPC - Linear Predictive Encoding
MERL - Mitsubishi Electric Research Labs
MFCC - Mel-Frequency Cepstral Coefficient
MIT - Massachusetts Institute of Technology
PDA - Personal digital assistants
SDK - software development kit
PDS - Processador Digital de Sinais
PLP - Perceptual Linear Prediction
T.I – Tecnologia de Informação
UCSC - Universidade da Califórnia, em Santa Cruz
Ulbra - Universidade Luterana Brasileira
VoIP – Voz sobre IP

NOTA DO EDITOR

Mais uma produção acadêmica de interesse da sociedade faz parte do trabalho de Rogério Marchi Marano.

Como trabalho de conclusão do curso de Sistemas de Informação da Universidade do Estado de Minas Gerais (UEMG) – Unidade Frutal, também contou com a orientação do Professor Sylvio Barbon Jr.

A versão original impressa poderá ser consultada na Biblioteca da Unidade de Ubá. Nossa alegria é imensa por contar com a autora no trabalho de popularização da ciência e da divulgação científica. Quando nos permitiu publicar o trabalho para torná-lo acessível para consulta gratuitamente na *internet* contribuiu para a ampliação da cultura do acesso livre ao conhecimento e da transparência das atividades universitárias.

Professor Otávio Luiz Machado
Editora Prospectiva

1 – INTRODUÇÃO

Este projeto trará um estudo sobre comando de voz para que seja possível interagir com o meio eletrônico apenas contando com a sua voz, de modo a facilitar tarefas do dia-a-dia que nem sempre são fáceis para pessoas com determinadas limitações. Um dos grandes feitos da nova era da tecnologia é a implementação de sistemas para ajudar portadores de necessidades especiais, implementando em forma de softwares usabilidades que facilitam a interação destas pessoas com o mundo virtual e tecnológico. Usuários com deficiências motoras e visuais possuem dificuldades para realizar tarefas simples do dia-a-dia, como a utilização de aparelhos eletrodomésticos, computadores, ou simplesmente acender uma lâmpada.

No Capítulo 1 abrangeremos o foco de nosso projeto, mostrando um breve histórico sobre a tecnologia via voz. Também falaremos sobre a utilidade de tal tecnologia para os portadores de necessidades especiais, noções sobre voz, e, por último, entraremos na parte teórica sobre processadores digitais de voz.

Em seguida, no Capítulo 2, mostraremos como essa tecnologia vem se difundindo no mundo,

mostrando suas vantagens e onde elas estão sendo aplicadas.

Por fim, falaremos no Capítulo 3 sobre o software que utilizaremos para realizar nosso projeto, denominado Sphinx4, dando ênfase na utilização de suas ferramentas.

1.1 - RESUMO

Hoje em dia a tecnologia abrange vários tipos de áreas. Vivemos num mundo onde a agilidade e facilidade que os meios tecnológicos nos trazem são fundamentais para o bem estar do ser humano. Um desses caminhos é a usabilidade para os portadores de necessidades especiais, trazendo a eles o mínimo de condições para realizar as tarefas que uma pessoa sem essas limitações realizaria. Este projeto trará um estudo sobre comando de voz para que seja possível interagir com o meio eletrônico apenas contando com a sua voz, de modo a facilitar tarefas do dia-a-dia que nem sempre são fáceis para pessoas com determinadas limitações.

1.2-NOVA TENDÊNCIA:

Um dos grandes feitos da nova era da tecnologia é a implementação de sistemas para ajudar portadores de necessidades especiais, implementando em forma de softwares usabilidades que facilitam a interação destas pessoas com o mundo virtual e tecnológico. Usuários com deficiências motoras e visuais possuem dificuldades para realizar tarefas simples do dia-a-dia, como a utilização de aparelhos eletrodomésticos, computadores, ou simplesmente acender uma lâmpada. A tecnologia de voz visa fazer com que o sistema reconheça a fala humana e a processe de forma que saiba através do que foi reconhecido, que decisão tomar.

A voz humana é produzida pela vibração do ar expulso dos pulmões pelo diafragma, que passa pelas pregas vocais e é modificado pela boca, lábios e língua. É uma função humana que está intimamente ligada à necessidade do homem em se comunicar. Está associada à comunicação verbal e pode variar quanto à intensidade, altura, inflexão, ressonância, articulação, entre outras características.

A partir destas características, será criada uma solução para comandar o hardware de um computador através da voz humana.

O projeto aqui proposto tem como objetivo estudar os princípios desta tecnologia.

Através deste estudo, podemos chegar a uma série de tecnologias que poderão ser implementadas pelo controle de voz.

Nosso principal intuito é que ela seja usada não só como uma forma de conforto e praticidade, mas também para beneficiar portadores de necessidades especiais, que com um simples comando de voz ou o som de uma palma, podem acender, por exemplo, a luz de um cômodo. Para aplicações mais avançadas, controlar eletrodomésticos, carros, máquinas de construção, e fazer muitas outras atividades do dia-a-dia.

Imaginem, por exemplo, uma pessoa com problemas motores podendo trabalhar em qualquer serviço que antes exigiria dele uma maior destreza corporal, podendo realizar todo seu trabalho apenas com a voz.

Este projeto é um primeiro passo para o aprendizado desta tão útil tecnologia, para que cada vez mais as pessoas, independente de suas dificuldades e limitações, possam encontrar com

maior facilidade, uma forma de melhor realizarem diversos tipos de tarefas.

Uma das tecnologias usada será o processador digital de sinais (PDS), que tem como função converter sinais de voz analógicos (como a voz humana) em sinais digitais (reconhecidos pelos aparelhos eletrônicos). O sistema captará a frequência do som emitido, que pode ser alto ou baixo, e então reconhecerá as características dele, comparando com o(s) de seu banco de dados. Caso estas características sejam compatíveis, ele então executará a ação desejada.

1.3 - JUSTIFICATIVA

Vivemos num mundo onde grande parte da nossa realidade ainda não está adaptada para portadores de necessidades especiais, trazendo várias dificuldades a eles para se deslocar de um ponto ao outro, usufruir de um estabelecimento de entretenimento, usar equipamentos domésticos, entre outros.

Como é explicito no mundo de hoje, a tecnologia está cada vez mais presente em nossas vidas, realizando tarefas de forma rápida e eficiente, com grande comodidade e praticidade. Mas nem

sempre os equipamentos disponíveis para estas tarefas estão acessíveis aos portadores de necessidades especiais. Como já podemos notar, a tendência da nova era da tecnologia de informação nos deixará cada vez mais dependentes de seus avanços. Hoje é cada vez mais comum usarmos estas tecnologias para realizar tarefas básicas do nosso cotidiano, como fazer compras, realizar transações comerciais e bancárias, nos comunicarmos com nossos amigos e parentes que estão distantes, e também alguns que são menos comuns, mas que tendem a virar uma tendência, como estudo à distância e conferências empresariais.

Este software estuda a possibilidade de proporcionar aos portadores de necessidades especiais maior facilidade para interagir com meios tecnológicos, como luzes domiciliares, computadores, entre outros, fazendo com que suas limitações não sejam uma barreira para usufruir de tarefas comuns para a maioria das pessoas e que apesar de muitas vezes serem simples, são de grande importância para nosso bem estar. Com ele, os dispositivos tecnológicos poderão ser ativados apenas com os sinais captados de nossa voz.

1.4 - EMBASAMENTO TEÓRICO

Na Turquia, o professor de informática da ULBRA (Universidade Luterana Brasileira) Adriano Petry fez uma pesquisa sobre reconhecimento e comando de voz à distância. Nela ele diz ser possível fazer as máquinas receberem tarefas à distância, apenas usando a voz. Também afirma ser possível fazer transações comerciais pela internet, tudo usando o comando por voz.

1.5 - OBJETIVOS

- **1.5.1 - OBJETIVO GERAL**

Ativar determinado dispositivo de hardware usando comando de voz.

- **1.5.2 - OBJETIVO ESPECÍFICO**

Desenvolver um software para que portadores de necessidades especiais possam dar comandos ao hardware do computador usando a voz, fazendo com que o equipamento responda a tarefa solicitada convertendo o sinal analógico da voz humana para o digital, via rede.

2 - CAPÍTULO 1 – RECONHECIMENTO E PROCESSAMENTO DE VOZ

2.1 - Histórico

Conforme citado por MENEZES, 2008, segundo o CEFET-SC – Centro de Educação Tecnológica de Santa Catarina, "os primeiros trabalhos de reconhecimento de voz tiveram início no século XVII, ainda sendo preciso esperar por mais meio século para que se começasse a apresentar resultados...".

Em uma cronologia podemos citar os seguintes trabalhos, segundo Gariba, 2002 *apud* MENEZES, 2008:

- 1930 - O americano R. J. Wensley contruiu o Televox, autônomo capaz de receber pela primeira vez ordens dadas via telefone, respondendo com alguns movimentos;
- 1952 – Daves cria um sistema inteiramente a cabos com capacidade de reconhecer dez números pronunciados por um locutor. Este sistema foi aperfeiçoado em 1958 para aceitar diversos locutores;
- 1956 – Olson e Belar propuseram um sistema com o nome de máquina de

escrever fonética, com capacidade de reconhecer uma dezena de palavras;

- 1958 – Denes define em duas etapas um sistema capaz de reconhecer primeiramente um som puramente acústico, e depois refinando o mesmo pela utilização de conhecimento linguístico;
- 1960 – Surgem as pesquisas sobre os métodos numéricos, que com a utilização dos computadores, têm uma nova dimensão;
- 1966 – Comparando as formas das palavras, sistemas em laboratório conseguem identificar corretamente 30 a 50 palavras emitidas por diferentes pessoas;
- 1968 – Alter e Reddy verificam a utilidade das informações linguísticas no reconhecimento da fala. Vicens e Tubach concretizam em 1969 e 1970, respectivamente, trabalhos neste enfoque.
- 1971 a 1976 – A ARPA (Advanced Research Projects Agency), financia um projeto americano sobre o tratamento da fala contínua com influência da inteligência artificial. Este projeto era capaz de compreender um vocábulo de mil palavras,

utilizar sintaxe artificial de escopo de uma tarefa precisa e responder a isso em um tempo próximo do real.

- 1975 e 1976 - surgem respectivamente o DRAGON e o HARPY, que eram capazes de trabalhar com um discurso contínuo de um único usuário com um vocábulo de mil palavras, obtendo uma taxa de acerto entre 84% e 97%;
- 1985 – a IBM lança o TANGORA. Uma versão que sacrifica a fala contínua para um acerto de 97% e vocabulário de vinte mil palavras;
- 1987 – Com uma precisão de 97%, os Laboratórios Bell reconheceram os dígitos de um telefone;
- 1988 – o SPHINX (que será citado mais abaixo) reconhece com precisão de 96% fala contínua, independente do locutor, com vocábulo de mil palavras. Tudo isso em tempo real;
- Final da década de 80 – Teuvo Kohonen, da Universidade de Tecnologia de Helsinki, desenvolve uma máquina de escrever por voz. Para isso ele utilizou a

- combinação de DSP (processadores digitais de sinais) com sistemas baseados em regras e redes neurais. Obteve taxas de 92% a 97% utilizando casos extremos de conversações fala-texto, contínua, através de vários locutores e grandes vocábulos de ¼ segundo de resposta;
- 1994 – A chegada do reconhecimento por voz no mercado é anunciada por vários artigos, devendo chegar a um bilhão de dólares até 1999. Entre os sistemas desenvolvidos para este fim estão o Personal Ditaction System, da IBM, e o Dragon Dictate, da Dragon System.

Com os avanços desta tecnologia, já contamos hoje com implementações comerciais ou em fase de teste, em comando de voz para maquinários receberem tarefas à distância, ser realizadas transações comerciais pela internet, fazer chamadas em aparelhos celulares, prédios moverem seus cômodos, computadores, robôs e elevadores realizarem tarefas, luzes apagarem ou acenderem, entre outros.

2.2 - Portadores de Necessidades Especiais

Segundo o art. 6°, parágrafo X da Consulta Pública N.° 494, de 19 de Janeiro de 2004 do Conselho Diretor da Agência Nacional de telecomunicações – Anatel, sobre a proposta de plano geral de metas para a universalização do serviço de comunicações digitais destinados ao uso do público em geral prestado no regime público, "Pessoas Portadoras de Necessidades Especiais: são àquelas que possuem perda ou anormalidade de uma estrutura ou função psicológica, fisiológica ou anatômica que gere incapacidade para o desempenho de atividade, dentro do padrão considerado normal para o ser humano;".

Segundo o Censo 2000 do Instituto Brasileiro de Geografia e Estatística – IBGE, um total de 24,6 milhões de pessoas se declararam portadoras de necessidades especiais. Suas dificuldades se apresentam de diversas formas, como na visão, locomoção, audição, fala, entre outros.

2.3 - Usabilidade para Portadores de Necessidades Especiais

Os portadores de necessidades especiais ainda encontram diversas dificuldades de acesso em seu dia-a-dia. Apesar dos esforços realizados até hoje, ainda está longe do mundo se tornar um mundo de total usabilidade. A tecnologia de informação tem ganhado seu espaço na implementação de soluções que contribuam para amenizar as dificuldades destas pessoas.

Hoje em dia já temos sites totalmente adaptáveis para isso, softwares de reconhecimento da escrita, tato, voz, entre outros. A T.I. está ajudando a adaptar o mundo para que este seja mais acessível aos portadores de necessidades especiais. É importante que os profissionais desta área estejam cada vez mais empenhados quanto a usabilidade de seus serviços e produtos.

2.4 - A Voz

A voz humana é produzida pela vibração do ar expulso dos pulmões pelo diafragma, que passa pelas pregas vocais e é modificado pela boca, lábios e

língua. É uma função humana que está intimamente ligada à necessidade do homem em se comunicar. Está associada à comunicação verbal e pode variar quanto à intensidade, altura, inflexão, ressonância, articulação, entre outras características.

Existem dois períodos para a voz. O período *pitch*, se o sinal for periódico, e caso contrário, *unvoiced speech*.

No *pitch* encontram-se basicamente as vogais, enquanto no *unvoiced speech*, os demais sons. Dependendo de como o ar passa pelo trato vocal ou nasal, pode-se observar e classificar os sinais vocais da seguinte forma:

Fricatives**:** *Unvoiced speed* que surge quando há fricção do ar, geralmente causando uma turbulência do ar entre a língua e os dentes superiores, como o FI da palavra "figura".

Plosives**:** *Unvoiced speed* impulsivo, como na palavra "taco", *onde* temos o TA.

Whispers**:** Na palavra "ré" (manobra do carro), temos uma *unvoiced speed* que cria uma barreira nas cordas vocais permanecendo parcialmente fechadas ou sem oscilação.

Voiced Fricative**:** Fonemas *voiced* (de excitação periódica), misturado com ruídos criados

atrás dos dentes e contra o palato, como o VA da palavra “vaso”.

***Unvoiced Fricatives*:** Igual ao *voiced fricative*, mas sem as vibrações simultâneas das cordas vocais com a fricação, como o SE da palavra “seja”.

***Voiced Plosives*:** Fonemas *voice* misturados com ruído impulsivo, como por exemplo o TOU da palavra “estouro”.

***Unvoiced Plosives*:** Igual ao *voiced plosives*. Como exemplo, temos o BO da palavra “boxe”.

2.5 - Interpretação da Fala

Nossa audição funciona de forma bastante elaborada. Nosso ouvido é classificado em três partes, sendo elas o ouvido externo, responsável por coletar os sons e conduzi-los através do canal auditivo ao ouvido médio. Por sua vez, o ouvido médio converte a pressão do ar para movimentos de um fluído e o leva ao ouvido interno, mais especificamente para uma estrutura associada à membrana basilar, chamada *cochlea*. No ouvido interno, os sons são separados de acordo com as frequências e a movimentação fluídica é convertida em impulsos elétricos no nervo auditivo, logo após, sendo captado pelo cérebro.

2.6 - Processador Digital de Sinais (PDS)

Uma das tecnologias usada será o processador digital de sinais (PDS), que tem como função converter sinais de voz analógicos (como a voz humana) em sinais digitais (reconhecidos pelos aparelhos eletrônicos) e que chamamos de Transformada. O sistema captará a frequência do som emitido, que pode ser alto ou baixo, e então reconhecerá as características dele, comparando com o(s) de seu banco de dados. Caso estas características sejam compatíveis, ele então executará a ação desejada.

O objetivo do processamento de sinais é extrair a informação carregada por um sinal. Este método de extração depende do tipo de sinal e de sua natureza, representando matematicamente os sinais com operações algorítmicas, podendo ser sua representação em termos de função base no domínio das variáveis independentes, originais ou em termos de funções base no domínio da transformada. O processo de extração de informações também segue a mesma representação. Esta tecnologia manipula o processamento do "vocoding", (transmissão de um conjunto de parâmetros característicos do sinal de

voz com o objetivo de possibilitar a sua futura síntese em um receptor), que utiliza o modelo de predição linear (LPC – *linear Predictive Coding*), apresentando algumas vantagens em se tratando da retirada de ecos, ruídos, reconhecimento de voz e qualidade de equalização.

2.7 - Tipos de Sinal

Vários podem ser os tipos de sinal, dependendo de sua natureza e valores das funções. Eles podem ser gerados por uma fonte única ou por múltiplas fontes, tendo no primeiro caso um sinal escalar e no segundo um vetor, também chamado sinal em multicanais. O sinal de uma dimensão (1-D) é uma função de uma única variável independente. Já um sinal bidimensional (2-D) é uma função de duas variáveis independentes. Um sinal multidimensional (M-D) é uma função com mais de uma variável. O sinal de voz é um sinal unidimensional, sendo a variável independente, o tempo, como explicada no tópico a seguir.

2.8 - Quantização

Ao converter um sinal analógico para digital, uma sequência de amostras da variação de voltagem do sinal original é criada. Para ser armazenada pelo computador, esta escala é convertida para valores binários (0 e 1), sendo então este sinal chamado de sinal discreto.

Veja o exemplo nas figuras 1 e 2:

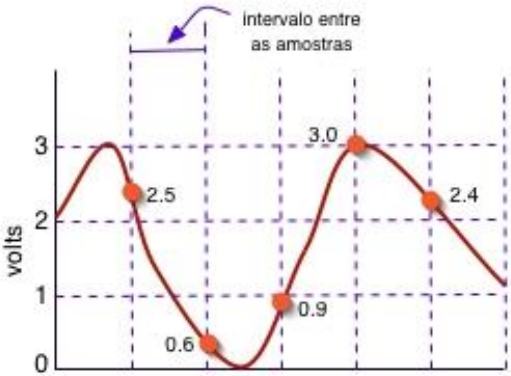


Figura 1: Gráfico demonstrando as ondas sonoras, com as amostras, seus valores e intervalos

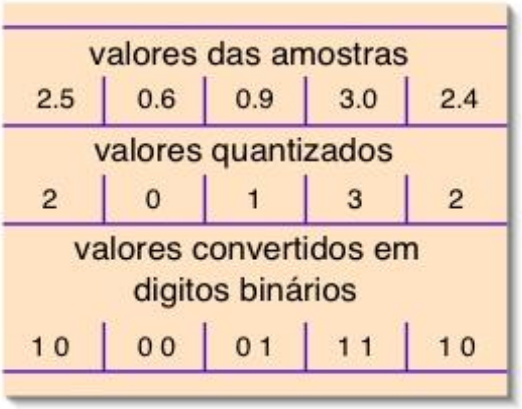


Figura 2: Tabela com os valores das amostras da quantização e sua descrição binária

2.9 - Taxa de Amostragem

Medidas em intervalos fixos, os números de vezes que as amostras são realizadas em uma unidade de tempo (a cada um segundo) é chamado de Taxa de Amostragem. É geralmente medida em Hz (Hertz).

Podemos exemplificar usando a taxa de Hertz de um arquivo de áudio. Se ele possui uma taxa de amostragem de 44.300 Hz, isso significa que a cada um segundo de som são tomadas 44.300 medidas da

variação do sinal. Quanto maior a taxa de amostragem, mais precisa é a representação do sinal, e também maior será o espaço utilizado para armazenar este arquivo.

2.10 - Tratamento de Aliasing em voz

Quando o correspondente analógico de um sinal digital é amostrado em uma taxa de Hertz insuficiente, surgem os *aliasing*, ou sinais "fantasmas", como demonstrado na figura 3.

As amostras são representadas pelos pontos verdes e a onda azul, o efeito *aliasing*.

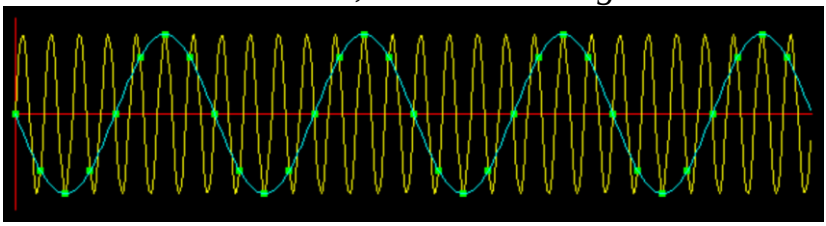


Figura 3: A faixa amarela representa uma onda de 17.500 Hz digitalizada com uma taxa de amostragem de 20.000 Hz.

Para se tratar este inconveniente, os intervalos das ondas da faixa amarelo acima deveriam conter

pelo menos duas amostras cada. Para isso, devemos usar variáveis do tipo *double*, por exemplo.

Uma amostra contendo apenas um bit poderia receber apenas os valores 0 e 1. Já em uma representação contendo três bits, uma amostra receberia 8 valores diferentes (000, 001, 010, 100, 110, 101, 011, 111).

Um CD de 16 bits (2^{16}) teria uma resolução binária com 65.534 valores. Ou seja, quanto maior a taxa da amostragem e da resolução, mais o som digital se aproxima do original.

2.11 - Transformada de Fourier

A transformada de Fourier desempenha um papel de grande importância em vários ramos das ciências exatas. As séries de Fourier são um caso particular da transformada de Fourier e permitem decompor uma função periódica qualquer na soma de um número infinito de funções senoidais com diferentes frequências e amplitudes. Além de poder ser empregado diretamente em um grande número de problemas, sendo a transformada discreta de Fourier bastante conveniente e diretamente associada ao processamento digital de sinais, por possuir

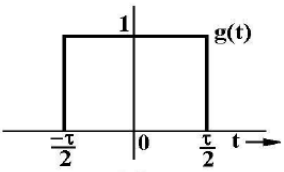
algoritmos capazes de computá-la com grande eficiência.

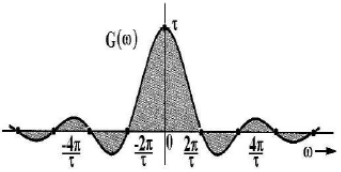
Transformada Direta de Fourier (G(ω))

$$G(\omega) = \int_{-\infty}^{\infty} g(t)e^{-j\omega t}dt$$

$G(\omega) = \Im[g(t)]$

$g(t) \Leftrightarrow G(\omega)$ → par de transformada de Fourier





$$G(w) = F(g(t)) = \int_{-\infty}^{\infty} g(t) e^{-jwt} = dt =$$

$$\int_{-\frac{\tau}{2}}^{\frac{\tau}{2}} 1 * e^{-jwt} dt = \frac{e^{\frac{jw\tau}{2}}}{-jw} = \int_{-\frac{\tau}{2}}^{\frac{\tau}{2}} = -\left[\frac{e^{-jwt}}{jw}\right]_{-\frac{\tau}{2}}^{\frac{\tau}{2}} = -\left(\frac{e^{\frac{-jw\tau}{2}} - e^{\frac{+jw\tau}{2}}}{jw}\right) =$$

$$\frac{e^{\frac{-jw\tau}{2}} - e^{\frac{-jw\tau}{2}}}{jw} * \frac{2}{2} = \frac{2}{w}\left(\frac{e^{\frac{jw\tau}{2}} - e^{\frac{-jw\tau}{2}}}{2j}\right) = \frac{2}{w} sen\left(\frac{w\tau}{2}\right) * \frac{\tau}{\tau} = \frac{\tau sen\left(\frac{w\tau}{2}\right)}{\left(\frac{w\tau}{2}\right)} =$$

$$\tau \operatorname{sin} c\left(\frac{w\tau}{2}\right) => G(w) = \frac{\tau sen\left(\frac{w\tau}{2}\right)}{\frac{w\tau}{2}}$$

$$rect\frac{t}{\tau} = g(t) <=> \tau \operatorname{sin} c\frac{w\tau}{2} = G(w)$$

$w = 0$

$$G(w) = \frac{\tau sen\left(\frac{w\tau}{2}\right)}{\frac{w\tau}{2}}$$

$$G(0) = \tau \frac{sen0}{0}$$

$$\lim_{w->0} \tau \frac{sen\left(\frac{w\tau}{2}\right)}{\left(\frac{w\tau}{2}\right)} = \lim_{w->0} \tau \frac{\cos\left(\frac{w\tau}{2}\right)}{\left(\frac{\tau}{2}\right)} * \frac{\tau}{2} = \tau$$

$w \neq 0$

$$G(w) = \tau \frac{sen\left(\frac{w\tau}{2}\right)}{\frac{w\tau}{2}} = 0$$

$$sen\left(\frac{w\tau}{2}\right) = 0$$

$$\frac{w\tau}{2} = k\pi =$$

$$w = \frac{2k\pi}{\tau}$$

2.12 - Limitações no Reconhecimento da Fala

É importante observar os fatores de como a entrada da voz no sistema é disposta.

Entre os exemplos estão:

- A qualidade do microfone: onde o espectro do sinal convertido pode trazer erros na cadeia do processo de reconhecimento;
- Modo de fala isolada ou contínua: relacionada com a capacidade de processamento do software e sua qualidade quanto a separação das palavras reconhecidas;
- Se a fala é lida ou espontânea: uma leitura terá uma probabilidade de reconhecimento da fala maior, em se tratando de sua velocidade e minimizando os vícios de linguagem;
- Independência ou não do usuário que fala: o computador deve reconhecer um ou mais usuários, diferenciando seu sexo ou idade. Também pode-se considerar o estado emocional do usuário com relação a stress, nervosismo, ansiedade, etc.;
- Tamanho do vocabulário: a quantidade de palavras ditas deve ser suportada pela capacidade de processamento e armazenamento do computador. Também se

pode aplicar conceitos de Inteligência Artificial para que o computador adicione ao seu banco de dados palavras que não eram reconhecidas por ele até então;

- Modelo de linguagem: A língua portuguesa, por exemplo, é bastante complexa, exigindo maior atenção às regras gramaticais implementadas no software;
- Perplexidade: o computador deve analisar o sentido da frase para determinar o significado de palavras iguais com sentidos diferentes;
- Ruído no ambiente: o computador deve distinguir a voz do usuário, que é o sinal, com os demais ruídos do ambiente.

2.13 – Ferramentas

- **Sphinx -** É uma biblioteca do software livre JAVA. Esta biblioteca de reconhecimento de voz faz o reconhecimento de todos os usuários sem que seja necessário gravar todos os tipos de vozes e frequências, se aproximando bastante do som original. Foi criado na Universidade de Carnegie Mellon e é hoje um dos melhores e mais versáteis sistemas de reconhecimento no mundo. Baseado em

HMM, tem como função primeira aprender as características dos sons, e em seguida utilizar o que aprendeu para encontrar a mais provável sequência de unidades de som como saída. Para utilizá-lo serão necessários o Sphinx Trainer e o Sphinx Decoder. Sua vantagem é reconhecer todos os tipos de voz, sem que seja necessária a prévia gravação da voz do usuário. Já uma desvantagem é sua biblioteca trabalhar com a verificação de grupos de palavras, fazendo a aproximação de uma palavra inesperada, ocasionando instabilidade. A língua portuguesa ainda é um problema neste tipo de implementação, mas como iremos usar apenas os comandos de voz *on* (ligar) e *off* (desligar), não será um problema aqui existente. Sua interface foi desenvolvida com base nas interfaces e protocolos Windows, sendo de fácil reconhecimento para o usuário.

- **Sphinx Trainer -** É um conjunto de programas responsáveis por tarefas bem definidas, junto a um conjunto de scripts que organiza a ordem em que os programas são chamados. Ele aprende os parâmetros dos modelos de unidades de som usando exemplos

de sinais de fala. Esta ação se chama *training database*. A escolha dos *training database* será escolhida pelo usuário e enviada para o Trainer através do *transcript file*. Em seguida, mapeia cada palavra através de um "dicionário" para obter a melhor associação para cada sinal. Segundo o site "CMU Sphinx Project Page", os elementos fornecidos para o Trainer são treinador de Código fonte, os sinais acústicos, o arquivo correspondente para a transcrição, o dicionário de línguas e o arquivo para o dicionário.

- **Sphinx Decoder -** Elaborado para gerar um único executável para o reconhecimento da tarefa, acionando corretas as entradas dos dados. Estas entradas são o modelo da formação acústica, um modelo de índice de arquivos, um modelo de linguagem, um dicionário de línguas, um dicionário de arquivos e um conjunto de sinais acústicos que devem ser reconhecidos (test data). Também segundo o site "CMU Sphinx Project Page", os elementos fornecidos para o Decoder são o decodificador de código fonte, o dicionário de línguas, o arquivo para o dicionário, o modelo de língua e o teste de dados. Além destes

componentes, é preciso ter os modelos acústicos que você treinou para o reconhecimento, fornecendo-os para o decodificador, onde o *trainer* irá gerar um modelo de índice apropriado para os arquivos. Estes índices contêm identificadores numéricos para cada estado de cada HMM, sendo utilizados pelo *trainer* e o *decoder* o conjunto correto de parâmetros para os estados HMM. Com isso o correspondente índice do modelo de arquivo deve ser utilizado para a descodificação.

3 – CAPÍTULO 2 (A TECNOLOGIA VIA VOZ NO MUNDO)

3.1 – Tecnologias via-voice

A tecnologia via voz é hoje algo presente em nosso dia-a-dia. O que antes era apenas ficção científica nos filmes e seriados hollywoodianos, hoje se encontra como uma tecnologia comum de se encontrar. Estacionamentos, computadores pessoais, elevadores, aparelhos celulares, entre diversos outros utilitários presentes em nossa vida já contam com tal tecnologia. Entre as ferramentas mais conhecidas no mercado estão o Sphinx, já comentado no capítulo anterior, o IBM Via Voz, Dragon Systems e Phillips. Entre outras tecnologias de implementações semelhantes estão o IVOS (Intelligent Voice Operating System) e o Voice Insert ActiveX, ambos para conversão de voz em texto. Também, o player de mídia, Voice Automated Media Player, que responde aos comandos de um player comum, apenas com o usuário utilizando sua voz.

Neste capítulo, vamos falar sobre algumas das diversas tecnologias existentes hoje, estando ou não sendo comercializados.

3.2 - O crescimento mundial do comando de voz

No imaginário das pessoas que assistiram filmes como *2001: Uma Odisseia no Espaço*, com o cérebro eletrônico HAL; R2D2 e C3PO em *Star Wars*; o super automóvel KIT, em *A Super Máquina*, a empregada Rosie do desenho *Os Jetsons*, entre tantas outras obras, já podia se ter uma idéia do que seria a tecnologia do século XXI. Computadores falantes já são uma realidade em laboratórios de pesquisa do mundo todo e seu uso são utilizados em geral para a segurança. A voz, impressão digital e a íris dos olhos, por serem únicas em cada ser humano, tornam-se uma espécie de senha biológica para acesso a ambientes e documentos de uso restrito.

O maior empecilho para que os sistemas de reconhecimento de voz chegassem aos lares e escritórios era o alto custo dos equipamentos aptos a realizar tal tarefa. Há alguns anos, um equipamento desse porte poderia custar até 20 mil dólares. Somente nos tempos atuais, estes computadores tornaram-se mais acessíveis, sendo o passo que faltava para que as ideias existentes fossem colocadas em prática, surgindo então diversas ferramentas que utilizam este sistema de comunicação entre o homem e a máquina.

Hoje em dia, já se pode encontrar uma vasta extensão desta tecnologia para diversos dispositivos. A coreana Samsung foi uma das primeiras empresas a oferecer no Brasil um celular com chamada de voz. A ideia é mudar o pensamento de que o homem deve adaptar a tecnologia àquilo que a máquina quer, mas sim o contrário. Assim, cada vez mais será descartado o uso de mouses e teclados.

3.3 - Exemplos de aplicação via voz

Citaremos aqui, diversas tecnologias que já utilizam comando por voz. Começarei pelo professor Adriano Petry, que em 2008 criou um projeto onde, usando apenas um microfone de um PC conectado a internet, mostra ser possível fazer aparelhos eletrônicos residenciais, máquinas industriais e de escritório, entenderem comandos de voz, mesmo a distância. Também afirma ser possível fazer transações comerciais pela internet.

O Windows Vista já conta com comando via voz, onde o usuário pode abrir programas, imprimir documentos, editar documentos e mensagens de correio eletrônico, preencher formulários na Web, entre outros.

O Windows XP também conta com diversos softwares de comando de voz. Em sites de downloads podemos encontrar diversos deles, entre os quais estão o Typle, o Tesponding Heads, Sannus Agent Lite, Speech SDK (kit de desenvolvimento para programas com suporte a reconhecimento de voz), e o Eddie the DJ (controlador de player via voz). Todos estes aqui citados são gratuitos.

O aparelho celular Nokia N95 utiliza comando de voz para localizar contatos na agenda e abrir aplicativos, sem a necessidade de abrir os aplicativos. Dentro da ferramenta "Comando de Voz" do aparelho é possível adicionar novos comandos, editá-los e reproduzi-los, sendo possível deixá-lo da forma que melhor for conveniente ao usuário.

O Suíte Vollard, primeiro prédio residencial giratório do mundo, situado na cidade de Curitiba, possui em seus apartamentos um eixo central fixo onde ficam localizados cozinha e banheiro, enquanto os outros cômodos podem se deslocar, mudando de lugar e dando ao apartamento um novo formato, de acordo com a vontade e criatividade do morador. Tudo isso é possível com o usuário usando apenas comando de voz para realizar tais modificações.

Ainda falando sobre moradia, em residências comuns, como casas, também já é possível utilizar a

voz para ligar equipamentos como ar-condicionado, banheira de hidromassagem e até mesmo lâmpadas. Além disso, com a central de automação instalada, é possível utilizar a voz para abrir portas de armários e adaptar a cama para uma posição mais confortável. Mesmo estando fora de casa, o usuário pode acionar o sistema por meio de um telefone.

O famoso site de busca Google lançou o serviço de voz para iPhone, o Google Mobile app for iPhone. O aplicativo permite fazer pesquisa através do celular sem usar nem uma tecla para digitar o que se deseja, bastado apenas falar o que se procura. E o sistema não se limita apenas a Web. Além do sistema de busca na internet, também pode ser usado para fazer buscas de contatos armazenados na agenda.

Indo pela mesma empreitada está o site Yahoo, que segundo o responsável pelos esforços de internet móvel da empresa, Marcos Boerries, o Yahoo deseja tornar milhões de links mais acessíveis pelo celular, aprofundando a navegação nas páginas da internet. A mais recente versão do aplicativo do grupo é o oneSearch, fechando acordo com várias operadoras de telefonia móvel em todo o mundo, tendo como objetivo proporcionar o serviço de internet

móvel a cerca de 600 milhões de usuários celulares, e pretende chegar aos 750 milhões com o acordo fechado como a Vivo aqui no Brasil.

A empresa já começou a ampliar os resultados de busca em celulares, permitindo aos provedores fornecerem informações altamente categorizadas, com o intuito de terem maior controle sobre o que chega ao usuário e de que maneira. Através da proposta de uma busca semântica na rede, o Yahoo pretende que com isso os computadores reconheçam e categorizem as informações que aparecerem em um website. Os usuários do oneSearch poderão utilizar comandos de voz para, além do reconhecimento se voz ou buscas em listas existentes, buscar ofertas de vôo, nomes de sites, restaurantes, notícias ou horários de partidas esportivas.

Nos aviões de caça *Mirage*, alguns comandos já são acionados pela voz do piloto, especialmente em manobras que necessitem o uso das duas mãos.

Os sistemas de reconhecimento de fala são adotados como atendentes virtuais em companhias telefônicas e empresas aéreas. Ao invés de digitar o ramal, através do bocal do telefone, basta dizer o nome do departamento ou da pessoa solicitada.

Elevadores com acionamento pela voz, que funciona como um elevador convencional, mas pode

ser chamado por intermédio de um aparelho PDA (Palm Top), celular ou por microfone. Elevadores como este, além de oferecer conforto e praticidade, facilitam a utilização de portadores de necessidades especiais em utilizar este meio de transporte.

O super-robô da Honda, conhecido como Asimo é um grande exemplo de reconhecimento de voz. Assim que ele tem uma primeira interação com uma pessoa, esta lhe diz: Oi Asimo, meu nome é "fulano", e ele então olha para a pessoa que falou, respondendo com o pedido que ela espere até que ele memorize o rosto da mesma. A partir daí, o Asimo interage com a pessoa como já a conhecendo e atende aos comandos dados a ele via voz.

Entre as façanhas do robô estão andar, correr, desviar de obstáculos, chutar bola de futebol ao gol, subir escadas, entre outras. Atualmente, o Asimo já está sendo desenvolvido para um patamar mais elevado de comunicação, o do pensamento, onde a pessoa utiliza um capacete eletrônico conectado ao BMI (Brain-Machine Interface), para dar o comando ao robô.

Tecnologias via voz também já são uma realidade em sistemas de segurança de carros. Com um sistema como o Block III o usuário pode, através de um celular Nokia, ter total controle sobre o carro,

podendo fazer o corte de combustível do mesmo, travar as portas, ser avisado por discagem caso o alarme for acionado, desligamento da bateria, entre outras funções.

Em 2003, a Intel lançou um software de auxílio a sistemas de reconhecimento de voz que permite que o computador realize tarefas similares a leitura dos lábios. O AVSR (Audio Visual Speed Recognition) promete melhorar a exatidão dos programas de reconhecimento de voz, especialmente com relação a ruídos existentes no ambiente. A meta é permitir aos PCs sincronizar os dados de vídeo capturados por uma câmera, ao som, trazendo melhorias ao reconhecimento de voz.

Estudantes da Fudan University, em Xangai, na China, criaram um robô que tem a capacidade de aprender coisas novas a partir de comandos de voz dados por humanos. O sistema inteligente assemelha-se com o aprendizado de uma criança e também funciona como TV, mudando de canal após comando de voz. A máquina possui um componente interno que tem como função receber e reconhecer vozes, além de memorizar comandos. O usuário pode colocá-lo na cozinha, por exemplo, e dizer: “Aqui é a cozinha”. O robô então memoriza a localização do

local e formula um mapa eletrônico para se guiar, quando lhe for solicitado, ao local aprendido por ele.

Comandos de voz já são utilizados para vender até mesmo ações da bolsa de valores. Também já é possível realizar pesquisas de opinião pública. Utilizando discadores automáticos, os institutos de pesquisa ligam para os entrevistados, que conversam com uma máquina que transforma as respostas orais em dados.

O serviço de auxílio a lista por reconhecimento de voz já é usado por empresas de telefonia, assim como em opções de serviço, onde o usuário, ao invés de discar no teclado do telefone o número referente ao que deseja, basta falar o que deseja para ser encaminhado a um atendente responsável. Alguns serviços mais simples, como emissão de segunda via, desbloqueio de linha e verificação de mudança de endereço podem ser feitos pela própria máquina.

Segundo Roberto Aragão, diretor de tecnologia da Velip, a tecnologia APIs Velip permite enviar diversos tipos de torpedo de voz on-line como:

- retorno do momento do atendimento para sincronismos com vídeos;
- conferência que liga para dois destinos, unindo-os conforme programação;

- interativos com respostas pelo teclado e áudios conforme as respostas;
- personalizados com áudios gerados por TTS, com nome do destinatário ou outras informações faladas no áudio;
- em lote, na qual o sistema armazena as requisições e dispara em um momento único também comandado pelas APIs;
- customizados conforme necessidade do projeto.

Entre as vantagens do uso do torpedo de voz estão não utilizar as linhas do callcenter, redução de custos, velocidade, relatórios on-line, atinge cadastros tanto de telefones fixos como celulares.

Além disso, as APIs ativam retornos com informações on-line de cada envio com tempos, valores, status de atendimento e códigos do cliente, permitindo um controle total do processo.

Com esta tecnologia as empresas podem preparar produtos sofisticados como:

- Vídeos sincronizados com torpedos de voz;
- Click-to-Call, solicitando uma ligação grátis para um callcenter ou destino determinado;

- Torpedo conferência, oferecendo, por exemplo, gratuitamente, uma chamada ou conferência;
- Solicitações de informes por áudio em totens ou celulares.

A Microsoft lançou o Game Voice, que permite ao jogador adicionar comandos de voz aos games durante suas partidas, substituindo comandos dados pelo teclado.

A IBM apresentou um programa de inteligência artificial que transforma a voz do usuário em sinais para comunicação usados para portadores de necessidades especiais auditivas. Desenvolvido por pesquisadores de diversas universidades inglesas, com o apoio da empresa, o software possui um avatar (boneco digital), que mostra os sinais correspondentes ao som de voz captado.

O programa foi chamado de Sisi (Say It Sign It), algo que se pode entender como *diga-o* ou *sinalize-o*. O módulo de reconhecimento de voz usado interpreta a voz de uma pessoa falando normalmente, como em uma palestra, sem que se tenha a preocupação com o tom da voz.

O Voice Command, também da Microsoft, transforma aparelhos PDA em um equipamento que

com apenas uma palavra, traz seus contatos, faz ligações e busca seu calendário. Pode ainda inicializar músicas e programas.

O jogo que simula a vida de forma virtual, *Second Life*, passou a ter a possibilidade de utilizar comando de voz para conversar, bastando para isso apenas um microfone acoplado ao computador. Esta plataforma nos leva a pensar numa nova possibilidade de interface gráfica para Internet, alternativa à Web.

O VR Commander permite ao usuário adicionar comando de voz para controlar a interface de jogos, da ferramenta CAD ou qualquer outro aplicativo do Windows, simulando teclas digitadas. É possível executar arquivos ou script e inserir textos ou comandos de qualquer tamanho. O poderoso sistema já é inclusive utilizado em alguns aviões avançados. Sua tecnologia de cancelamento de ruído permite que o software trabalhe até mesmo com dispositivos *Bluetooth* e VoIP.

A NASA desenvolveu, juntamente com a Xerox, um sistema via voz batizado de Clarissa, com o intuito de ser usado por astronautas na Estação Espacial Internacional. Este sistema funciona em um computador portátil convencional. Segundo o site CNET News, o sistema é uma versão real do HAL

9000, o computador falante do filme “2001: Uma Odisseia no Espaço”.

Segundo Beth Ann Hochey, líder do projeto, o Clarissa, desenvolvido no centro de pesquisa da NASA em Ames, no Vale do Silício, é um assistente virtual de equipe operado por voz, que possibilita aos astronautas serem mais eficientes com suas mãos e olhos, dando total atenção às tarefas a serem realizadas enquanto usam comandos falados. O sistema responde aos comandos de voz, lê em voz alta procedimentos conforme eles são executados, ajudando assim, o astronauta a não se perder, e também avisa sobre alarmes e timers.

A metodologia usada pela Xerox permite que o sistema Clarissa analise com maior precisão as expressões. Também é capaz de reconhecer palavras, sentenças e o contexto das palavras e pode agir de acordo com comandos falados de formas diferentes. O sistema enxerga cada palavra individualmente dentro de uma sentença e através de um algoritmo, faz com que a máquina aprenda o peso de cada parte da sentença, cortando assim metade dos erros de Clarissa em diferenciar conversas paralelas de ordens diretas.

3.4 - IBM Via Voice

Lançado pela IBM, o Via Voice é capaz de reconhecer palavras em português e compreender comandos básicos como salvar, imprimir ou abrir um documento. Além disso, este aplicativo ainda possui uma função chamada de síntese de fala, que é a capacidade do computador em reproduzir um texto pré-selecionado em linguagem oral, sendo esta função especialmente importante para portadores de necessidades especiais. O programa possui um vocábulo de 60 mil palavras e está adaptado a nove idiomas, entre eles o inglês americano e britânico, francês, chinês e espanhol, além do já citado português.

3.5 - Vantagens do Sphinx

Como já foi citado anteriormente, o Sphinx é uma das ferramentas para comando de voz mais conhecidas. Entre as vantagens que este software nos trás, citadas no site *sphinxbrasil.com*, podemos citar:

- A facilidade, amigabilidade, flexibilidade e autonomia;

- A capacidade de analisar e interpretar dados quantitativos e qualitativos (análise léxica, análise de conteúdo, codificação de textos para posteriores cruzamentos e explorações, dicionários temáticos agregadores de filtros, verbatim segmentado, entre outros);
- Sua aplicação compatível em todas as áreas, profissionais ou acadêmicas. (Sistemas - Marketing - Finanças – Gestão - Controladoria – Comercial – Industrial - Qualidade - Recursos Humanos, entre outros);
- Total integração com funções e estágios que cobrem todas as etapas do processo de pesquisa, coleta e análise de dados: concepção do questionário, formatação do formulário, coleta, entrada ou importação de dados, tratamento e análise dos dados, geração de relatórios, divulgação e publicação dos resultados; Módulo de digitação (Sphinx Operador);
- Novas tecnologias: pesquisa Web, multimeios, cartografia, filtros dinâmicos, segurança de dados e acesso controlado, triagem na web, legendas e ações personalizadas, regras de integridade, paineis dinâmicos, digitalização automatizada via scanner, entre outros;

- Interface com outros sistemas: relatórios automáticos para Word® e Powerpoint®, importação e exportação de dados, abertura de dados via ODBC;
- Possui diversos recursos para apresentação dos resultados de pesquisas e de análises, com relatórios, gráficos e análise científica, com facilidades tais como modelos de análises fornecidos pelo software (estilos, planos de tabulação, árvores de segmentação ilustradas com variáveis, gráfico de relações, desvios e restrições, ações e legenda personalizada, etc.), tudo muito simples, intuitivo e rápido.
- Pode-se fazer vários cruzamentos, AC, ACP, regressão, correlação, médias, tabela de características, análise léxica, análise de conteúdo, etc.
- Tanto o software como toda a documentação técnica estão em português. Website, manuais, suporte, soluções, vídeos, capacitação, autodemos, etc.

Capítulo 3 - Sphinx-4

4.1 - Introdução

O Sphinx-4 é um sistema de reconhecimento de fala escrito em linguagem de programação Java. Criado através em colaboração conjunta entre a Sphinx group at Carnegie Mellon University, Sun Microsystems Laboratories, Mitsubishi Electric Research Labs (MERL), e Hewlett Packard (HP), com as contribuições da Universidade da Califórnia, em Santa Cruz (UCSC), e o Massachusetts Institute of Technology (MIT).

É uma evolução do Sphinx-3, desenvolvido para ser mais flexível, tornando-se assim uma excelente plataforma para realizar o reconhecimento de fala.

Foi projetado com um elevado grau de flexibilidade e modularidade. Possui três módulos principais: o *FrontEnd*, o *Decoder*, e as *Linguist.* O *FrontEnd* possui um ou mais sinais de entrada com parâmetros para sequência de características. O *Linguist* traduz qualquer tipo de modelo de idioma padrão, juntamente com informações de pronúncia do dicionário e estruturas a partir de uma informação ou conjuntos de modelos acústicos, em um

SearchGraph. O *SearchManager* utiliza características do *FrontEnd* e os *SearchGraph* da *Linguist* para realizar a decodificação real, gerando *Results.* A qualquer momento, antes ou durante processo de reconhecimento, o pedido pode emitir controles de cada um dos módulos, tornando-se um parceiro efetivo no processo de reconhecimento.

O Sphinx-4 como a maioria dos sistemas de reconhecimento de fala, possui um grande número de parâmetros configuráveis, tais como pesquisa do tamanho do sinal, para ajustar o desempenho do sistema. O Sphinx-4 *ConfigurationManager* é usado para configurar tais parâmetros. Mas ao contrário de outros sistemas, o *ConfigurationManager* também dá ao Sphinx-4 a capacidade de carregar e configurar os módulos, em tempo de execução, obtendo assim um sistema flexível e acoplável.

Para dar aos aplicativos e desenvolvedores a capacidade de monitorar as estatísticas de decodificação, tais como a taxa de erros das palavras, velocidade e uso de memória, o Sphinx-4 proporciona uma série de ferramentas. Assim como em outros sistemas, as ferramentas são altamente configuráveis, permitindo aos usuários executar um amplo leque de sistema de análise.

Além disso, as ferramentas também possuem um ambiente interativo que permite aos utilizadores modificarem os parâmetros do sistema enquanto este estiver em execução, permitindo uma rápida interação com várias definições de parâmetros. Também fornece utilitários de apoio para dar suporte aos resultados dos níveis de aplicação de reconhecimento.

4.2 - FrontEnd

O objetivo do *FrontEnd* é parameterizar um sinal de entrada (como áudio) em uma sequência de recursos de saída. Paralelamente, compreende uma ou mais cadeias de processamentos de sinais substituível comunicando módulos denominados *DataProcessors* que suportam múltiplas cadeias, permitindo computar simultaneamente diferentes tipos de parâmetros, a partir da mesma, ou de diferentes sinais de entrada. Permite assim a criação de sistemas capazes de serem descodificados simultaneamente, utilizando diferentes tipos de parâmetro, tais como *MFCC* e *PLP*, ou mesmo tipos de parâmetro derivados da não intervenção de sinais, tais como o vídeo. Veja a figura 4:

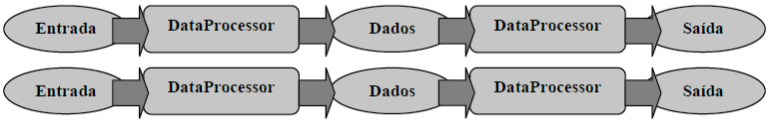


Figura 4: Processamento do *FrontEnd*

Cada *DataProcessor* do *FrontEnd* prevê uma entrada e uma saída que pode ser ligado a outro *DataProcessor*, permitindo arbitrariamente longas sequências de dados. As entradas e saídas de cada *DataProcessor* são objetos genéricos, que encapsulam dados processados de entrada, bem como marcadores que indicam o ponto final de detecção dos dados classificados como eventos. O último *DataProcessor* em cada cadeia é responsável por produzir um objeto composto de dados, que são sinais, características, a serem utilizados pelo *Decoder*.

O Sphinx-4 possui a capacidade de produzir sequências paralelas de características. No entanto, a medida que permite um número diferente de fluxos simultâneos.

A comunicação entre os blocos segue um fluxo determinado. Baseado neste fluxo, uma determinada

entrada requisita de um *DataProcessor*, em oposição ao *DataProcessor* já determinado, onde um módulo propaga a sua produção.

A capacidade de analisar do início para o fim ou vice-versa em uma pilha, não só permite o decodificador executar, assim como ocorre com o *Viterbi*, mas também permite que o decodificador realize outros tipos de pesquisas, tais como depth-first and A*.

Dentro do quadro genérico *FrontEnd*, o Sphinx-4 fornece um conjunto de *DataProcessors* que implementam sinais comuns de técnicas de processamento. Estas aplicações incluem suporte para leitura a partir de uma variedade de formatos de entrada para o modo de funcionamento batch, sistema de leitura do dispositivo de entrada de áudio para o modo de operação em tempo real, preemphasis, janelamento com uma levantada, transformada cosseno, por exemplo, janelas Hamming e Hanning), transformada de Fourier discreta (via FFT), mel frequency filtering, bark frequency warping, discrete cosine transform (DCT), linear predictive encoding (LPC), end pointing, cepstral mean normalization (CMN), mel-cepstra frequency coefficient extraction (MFCC), e

perceptual linear prediction coefficient extraction (PLP).

Usando o ConfigurationManager, os usuários podem utilizar o Sphinx-4 DataProcessors juntos, em qualquer forma, bem como incorporar ao DataProcessor suas próprias implementações do projeto. Como tal, a natureza modular e acopláveis não se aplica apenas no nível mais alto da estrutura Sphinx-4, mas também se aplica ao mais alto nível dos próprios módulos (ou seja, o FrontEnd é um módulo de encaixe, mas também é constituído de módulos próprios de acoplagem).

4.3 - Linguist:

Um *Linguist SearchGraph* gera o que é usado pelo decodificador durante uma pesquisa, ao mesmo tempo em que esconde as complexidades envolvidas na geração deste gráfico. O *Linguist* é um módulo de encaixe que permite ao usuário configurar dinamicamente o sistema com diferentes implementações *Linguist*.

Uma aplicação típica do *Linguist* é construir o *SearchGraph* utilizando linguagem estruturada, representado por um determinado modelo de linguagem.

O *Linguist* também pode utilizar um dicionário (tipicamente uma pronúncia léxico) para mapear palavras do modelo de linguagem em sequências de elementos de modelo acústico. Ao gerar o *SearchGraph*, o *Linguist* também pode incorporar sub-unidades de palavras com os contextos de comprimento arbitrário, se fornecido. Ao permitir que diferentes implementações do *Linguist* sejam plugados pelo runtime, o Sphinx-4 permite diferentes configurações de reconhecimento para requisitos individuais de sistemas. Existem três componentes acopláveis de *Linguists*: o Modelo de Linguagem, o Dicionário e o Modelo Acústico.

4.4 - Modelo de Linguagem

O Modelo de Linguagem, módulo do *Linguist*, prevê palavras com estruturas no nível linguístico, que podem ser representadas por qualquer quantia de palavras. Estas implementações normalmente dividem-se duas categorias: graph-driven grammars e modelos estocásticos N-Gram. O graph-driven grammars representa uma palavra baseada em seu formato na frequência, onde cada nó representa uma única palavra e cada saída representa a probabilidade da detecção de uma palavra. Os modelos estocásticos

N-Gram fornecem probabilidades de palavras dadas à observação da palavra anterior (como a palavra arco-íris).

As implementações do Modelo Acústico do Sphinx-4 possuem uma variedade de formatos, incluindo os seguintes:

- **SimpleWordListGrammar:** define uma gramática baseada em uma lista de palavras. Um parâmetro opcional define se há "loops" na gramática ou não. Se a gramática não se repete, então, esta será utilizada com palavras isoladas no reconhecimento. Será utilizado para apoiar reconhecimentos de palavras equivalentes e com gramáticas de iguais probabilidades.
- **JSGFGrammar :** suporte ao formato de gramática Speech API (JSGF) para Java, que define um estilo de FBN, de plataforma independente.
- **LMGrammar :** define uma gramática baseada em um modelo de linguagem estatística. Ele gera uma gramática por palavra e funciona bem com gramáticas pequenas, com cerca de até 1000 palavras.
- **FSTGrammar :** suporta um formato de gramática ARPA FST.

- **SimpleNGramModel :** fornece suporte para modelos ASCII N-Gram do formato ARPA. O SimpleNGramModel não faz nenhuma tentativa de otimizar o uso de memória, por isso funciona melhor com pequenos modelos de linguagem.
- **LargeTrigramModel :** fornece suporte para modelos verdadeiros de N-Gram gerados pela CMU-Cambridge Estatistical Langabari Modeling Toolkit. O LargeTrigramModel otimiza memória de armazenamento, permitindo trabalhar com grandes arquivos, de 100MB ou mais.

4.5 - Dicionário:

O Dicionário prevê a pronúncia das palavras encontradas no Modelo de Linguagem.

Nestas pronúncias, as palavras são quebradas em sequências de sub-unidades. A interface do Dicionário também suporta a classificação das palavras e permite que uma única palavra se encontre em várias classes. O Sphinx-4 fornece atualmente implementações da interface do Dicionário para apoiar o CMU Pronouncing Dicionary. Existem várias implementações para otimizar os padrões de

utilização, com base no tamanho do vocabulário ativo.

4.6 - Modelo Acústico:

O Modelo Acústico fornece um mapeamento entre uma unidade de fala e um HMM que podem ser analisadas pelo *FrontEnd*. Assim como acontece em outros sistemas, o mapeamento pode tomar decisões contextuais e definir a palavra pronunciada.

Normalmente, o *Linguint* transforma as pausas entre cada palavra em uma sub-unidade. O *Linguist* então passa as sub-unidades e os seus contextos ao Modelo Acústico, recuperando os resultados HMM associados a essas unidades.

Em seguida, utiliza esses resultados junto com o Modelo de Linguagem para a construção do SearchGraph. Ao contrário da maioria dos sistemas de reconhecimento vocal, que representam os resultados HMM como uma estrutura fixa na memória, o Sphinx-4 HMM é meramente um resultado parcial. Em vez de uma estrutura fixa, uma aplicação do Modelo Acústico pode utilizar os resultados HMM como fatores distintos.

As interfaces do Modelo Acústico não restringem a HMM quanto ao número de estados, o

número ou transições fora de qualquer estado, ou a direção de uma transição (para frente ou para trás). Além disso, Sphinx-4 permite que o número de estados de um HMM varie de uma unidade para outra num mesmo Modelo Acústico. Cada estado é capaz de produzir uma pontuação a partir de uma característica observada. O código real para calcular a pontuação é feito pelo próprio Estado, escondendo assim a sua aplicação no restante do sistema, permitindo ainda divergentes funções de probabilidades e densidade ao ser utilizado pelo estado HMM. Os elementos que compõem um determinado estado HMM, como Gaussian mixtures, transition matrices, e mixture weights, podem ser compartilhados por qualquer um dos estados HMM para melhorar o grau de precisão no reconhecimento.

Os usuários podem configurar Sphinx-4 com diferentes implementações baseadas no Modelo Acústico, de acordo com suas necessidades. Sphinx-4 prevê atualmente a execução de um único Modelo Acústico, que é capaz de carregar e utilizar modelos acústicos gerados pelo Sphinx-3.

4.7 - SearchGraph:

Modelos *Linguists* podem ser implementados de maneiras muito diferentes, e suas topologias dos espaços gerados pela pesquisa *Linguists* podem variar muito, sendo todos os espaços de pesquisa representados por uma *SearchGraph*.

Um *SearchState*, representa um estado de emissão e não emissão. Estados de emissão podem ser pontuados com as características acústicas recebidas. Enquanto os de não emissão são, geralmente, utilizadas para representar níveis construtores da linguagem, tais como palavras e fonemas.

A interface do *SearchGraph* é propositadamente genérica, permitindo uma ampla gama de formas de execução, aliviando as pressupostas limitações de hard-wired encontradas em sistemas de reconhecimento anteriores. Em particular, os lugares não inerentes do *Linguist* restringem-se ao seguinte:

- Topologia global de pesquisa espacial;
- Tamanho fonético do contexto;
- Tipo de gramática (ou regra estocástica em que foi baseada)

- Modelo de profundidade linguística N-Gram

Uma característica fundamental do *SearchGraph* é que a implementação dos SearchState não precisam ser corrigidas. Cada implementação *Linguist* prevê a sua própria aplicação concreta das *SearchState*, que podem variar baseadas nas características particulares do *Linguist*. Um *Linguist* possui um grande e complexo vocabulário, no entanto, podemos construir uma representação compacta interna do *SearchGraph*. Neste caso, o *Linguist* gera um conjunto de *SearchStates*.

A maneira pela qual o *SearchGraph* é construído afeta o carregamento da memória (footprint), velocidade, precisão e reconhecimento. A concepção modular do Sphinx-4, no entanto, permite diferentes formas de se utilizar a compilação do *SearchGraph*, sem alterar outros aspectos do sistema. A escolha entre a construção estática e dinâmica da linguagem HMM depende principalmente do tamanho do vocabulário, complexidade e modelo de linguagem desejado e da memória do sistema, podendo ser feito pelo aplicativo.

4.8 - Decodificador:

O papel principal do Sphinx-4 *decoder* é usar características do *FrontEnd* em conjugação com o *SearchGraph,* a partir do *Linguist,* para gerar resultados hipotéticos. O decodificador inclui um encaixe *SearchManager* e outros códigos que simplificam o processo de decodificação de uma aplicação. Como tal, o mais interessante dos componentes do decodificador é o *SearchManager*.

O decodificador simplesmente informa ao *SearchManager* que reconheça um conjunto de frames. Em cada etapa do processo, a *SearchManager* cria um resultado que contém todas as indicações de reconhecimento. Para processar o resultado, o Sphinx-4 também fornece utilitários capazes de produzir um índice e um resultado de reconhecimento. Ao contrário de outros sistemas, no entanto, os pedidos podem modificar a procura de espaço e do objeto Result por etapas, permitindo que a aplicação se torne um parceiro no processo de reconhecimento. Da mesma forma que o *Linguist,* o *SearchManager* não é restrito a uma determinada aplicação. Implementações de *SearchManager* podem realizar pesquisas de algoritmos tais como o

frame-synchronous Viterbi, A*, bi-directional, e assim por diante.

Cada aplicação *SearchManager* utiliza um algoritmo token. Um Sphinx-4 token é um símbolo, o objeto que está associado com uma *SearchState* e contém a pontuação global acústica e a linguagem do caminho em um determinado ponto, uma referência ao *SearchState*, e outras informações relevantes. O referente *SearchState* permite que a *SearchManager* relacione um token para o seu estado de saída distribuído, dependente do contexto da unidade fonética, pronúncia, palavra e estado da gramática. Qualquer hipótese parcial termina com um token ativo.

Implementações de um *SearchManager* podem construir um conjunto de arquivos, por vez, sob a forma de um *ActiveList*, embora a utilização de um *ActiveList* não seja necessário.

4.9 - Capacidade

- Capacidade de reconhecer falas discretas e contínuas;
- Inclui implementações de preemphasis, Hamming window, FFT, Mel frequency filter bank, discrete cosine transform, cepstral mean

normalization, e feature extraction of cepstra, delta cepstra, double delta cepstra features;

- Possui suporte para ASCII e versão binária do unigram, bigram, trigram, Java Speech API Grammar Format (JSGF), e ARPA-format FST grammars;
- Suporte para Sphinx-3 acoustic models;
- Suporte para pesquisa breadth first e word pruning;
- Utilitários para o pós-processamento dos resultados de reconhecimento, incluindo obtaining confidence scores, generating lattices e embedding ECMAScript into JSGF tags;
- Ferramentas Standalone incluindo displaying waveforms and spectrograms e generating features from audio;

4.10 - Desempenho

O Sphinx-4 é um sistema muito flexível, sendo capaz de realizar diversos tipos de tarefas em reconhecimento. Sendo assim, é difícil especificar seu desempenho e precisão com números. Em vez de usar esses recursos para qualificá-lo, são regularmente realizados testes para determinar a

forma como ele responde em diversas tarefas de tarefas. Os tópicos abaixo mostram estas tarefas e os seus resultados mais recentes, sendo cada tarefa mais difícil que a anterior:

- Dígitos Isolados (TI46): Executa o Sphinx-4 com os dados dos ensaios pré-gravados para obter métricas de desempenho de reconhecimento com apenas uma palavra de cada vez. O vocabulário é apenas a fala dos algarismos de 0 a 9, com a expressão contendo apenas um dígito. (TI46 refere à "NIST CD-ROM da Texas Instruments desenvolveu-46-Word Palestrante-Dependente Isoladas Word Speech Database").
- Dígitos Conectados (TIDIGITS): Teste para reconhecer mais de uma palavra em um momento (fala contínua). O vocabulário é apenas a pronúncia dos algarismos de 0 a 9, com uma única expressão contendo uma sequência de dígitos. (TIDIGITS refers to the "NIST CD-ROM Version of the Texas Instruments-developed Studio Quality Speaker-Independent Connected-Digit Corpus".) (TIDIGITS refere-se à "NIST CD-ROM da Texas Instruments

desenvolveu Estúdio Quality Speaker-Independent Conectado dígitos Corpus".)

- Pequeno Vocabulário (AN4): Abrange um vocabulário de aproximadamente 100 palavras, com dados de entrada variando entre palavras, letra por letra.
- Vocabulário Médio (RM1): Abrange um vocabulário de aproximadamente 1.000 palavras.
- Vocabulário Médio (WSJ5K): Abrange um vocabulário de aproximadamente 5.000 palavras.
- Vocabulário Médio (WSJ20K): Abrange um vocabulário de cerca de 20.000 palavras.
- Vocabulário Grande (HUB4): Abrange um vocabulário para cerca de 64.000 palavras.

A tabela 1 compara o desempenho de Sphinx 3/3 com Sphinx-4.

Teste	S3.3 ER	S4 WER	S3.3 TR	S4 TR (1)	S4 TR (2)	Tamanho do Vocabulário	Modelo de Linguagem
TI46	1,217	0,168	0,14	.03	.02	11	Reconhecimento de dígitos isolados
TIDIGITS	0,661	0,549	0,16	0,07	0,05	11	contínua dígitos
An4	1,300	1,192	0,38	0,25	0,20	79	trigram
RM1	2,746	2,88	0,50	0,50	0,41	1.000	trigram
WSJ5K	7,323	6,97	1,36	1,22	0,96	5.000	trigram
HUB4	18,845	18,756	3,06	~ 4,4	3,95	60.000	trigram

Tabela 1: Tabela de desempenho entre o Sphinx 3/3 com Sphinx-4

Note que o teste de desempenho no trabalho HUB4 não é completo

Chave:

- WER – trabalha a taxa de erro (%) - (quanto menor, melhor)
- RT - Tempo Real - Relação de tempo de processamento de áudio - (quanto menor, melhor)
- S3.3 RT - Resultos para configuração de CPU simples ou dual.
- S4 TR (1) - Resultados de uma única configuração de CPU.
- S4 TR (2) - Resultados de uma configuração de CPU dual.
- Esses dados foram coletados em uma CPU dual UltraSPARC (R)-III rodando a 1015 MHz e com 2G de memória.

4.11 - Requisitos:

O Sphinx-4 foi testado nos sistemas operacionais Solaris, Mac, Linux e Windows.

Deve possuir instalado na máquina o Java 2 SDK, Standart Edition 5.0 ou melhor, Ant 1.6.0 ou superior (só é necessário caso deseje desenvolver a partir do código fonte), e o Subversion (SVN – para ambiente Linux).

4.12 - Demonstrações do Sphinx:

Para ajudar o usuário, o Sphinx-4 possui uma série de programas de demonstração JAR, já construídas na versão binária (versão sphinx4-()-bin.zip), podendo executá-las diretamente.

Capítulo 4 – Documentação

5.1 – Caso de uso e diagrama de sequência:

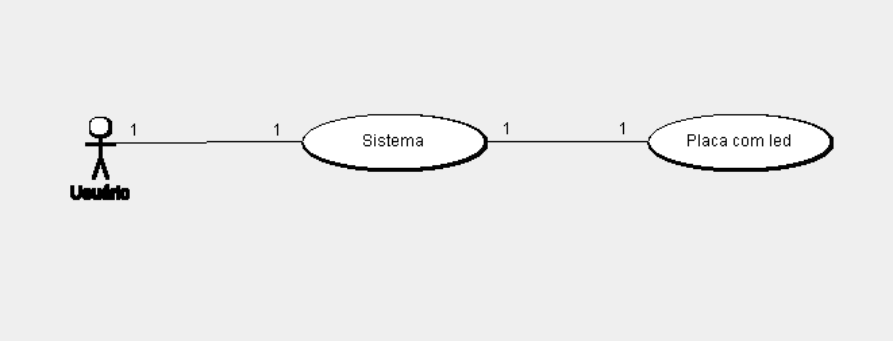


Figura 5: Caso de uso

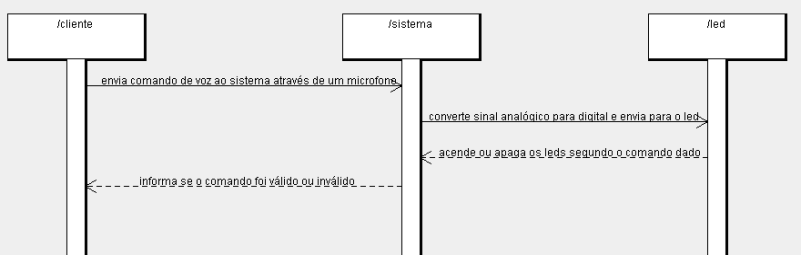


Figura 6: Diagrama de sequência

CONSIDERAÇÕES FINAIS

Este projeto visa acessar um dispositivo diretamente conectado a uma máquina de computador, visando o estudo de tecnologia via voice para a implementação de possíveis projetos futuros que possam ajudar portadores de necessidades especiais.

Aplicações viáveis para esta tecnologia como projetos futuros seriam no próprio site da UEMG, ou mesmo na biblioteca do estabelecimento. Funcionários e até mesmo alunos poderiam fazer busca de conteúdo didático apenas dizendo o nome do que se deseja encontrar, tendo como resposta do software o local onde o livro, artigo ou revista se encontra. Sendo assim, todo o planejamento do que se espera com este projeto está claro e a implementação de um modelo do que será o resultado final já está em andamento. O estudo e implementação deste software de reconhecimento de voz se mostra totalmente viável, tendo em vista seus benefícios e também por ser uma tecnologia atualmente bastante difundida.

REFERÊNCIAS:

PAIVA, Rafael Cauduro Dias de . *Monografia sobre Transformações em Sinais de Voz: Morphing e Modificação de Pitch*;

Comparison of Techniques for Environmental Sound Recognition;

PELLENZ, Marcelo E. *Processamento Digital de Sinais (PDS)*;

CORRÊA JÚNIOR, Rivaldo Guedes; OLIVEIRA, Jansen Carlos de. *A Tecnologia do Processador Digital de Sinal (PDS) Aplicada ao Rádio Definido por Software (RDS) - 3G*;

ALBUQUERQUE, Márcio Portes de; ALBUQUERQUE, Marcelo Portes de. *Processamento Digital de Sinais*;

MENEZES, Juliano Leonel de. *Monografia sobre Controle de Processos por Voz*;

http://www.cin.ufpe.br/~if143/projetos/saulofft1d/index.html

http://cmusphinx.sourceforge.net; Acesso em 20 de maio de 2009, às 00:35.

http://info.abril.com.br/aberto/infonews/052008/23052008-4.shl; Acesso em 11 de fevereiro de 2009, às 13:52.

http://sistemas.anatel.gov.br/SACP/Contribuicoes/TextoConsulta.asp?CodProcesso=C499&Tipo=1&Opcao=realizadas; Acesso em 10 de março de 2009, às 15:36.
http://student.dei.uc.pt/~guilhoto/downloads/voz.pdf; Acesso em 20 de fevereiro de 2009, às 12:40.
http://supertrunfonet.tripod.com/trunfonticiadofuturo/id5.html; Acesso em 21 de junho de 2009, às 13:15.
http://www.cefetsc.edu.br/~gariba/VOZ.PRN.pdf; Acesso em 17 de março de 2009, às 09:03.
http://www.cts.org.br/includes/Til2003.pdf; Acesso em 12 de março de 2009, às 00:45.
http://www.eca.usp.br/prof/iazzetta/tutor/audio/a_digital/a_digital.html; Acesso em 18 de abril de 2009, às 10:33

Anexo 1

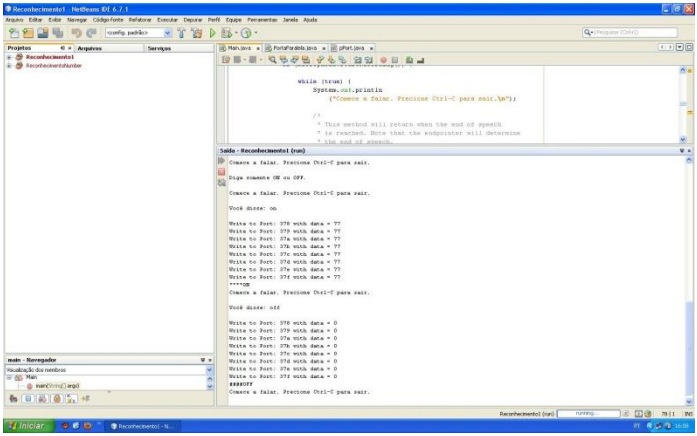


Figura 7: Tela principal do sistema

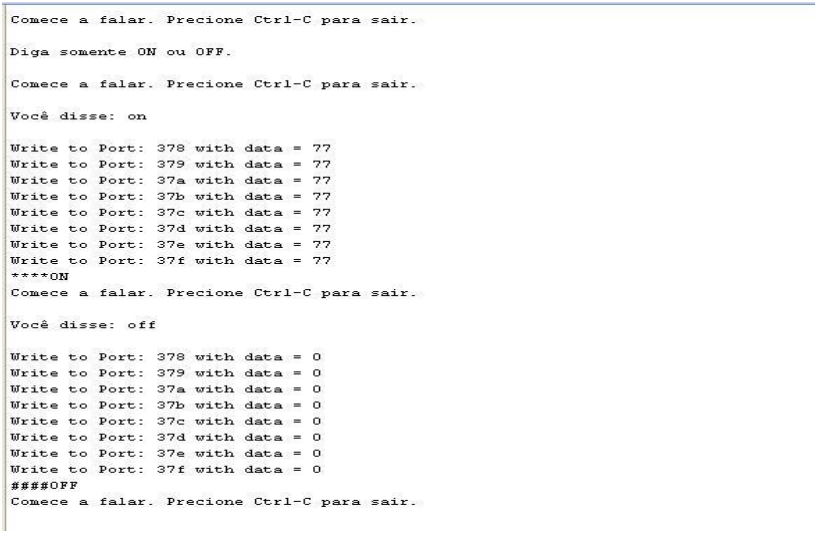


Figura 8: Tela com visão ampliada do sistema


Editora Prospectiva