Curso de Ciência da Computação
Universidade Estadual de Mato Grosso do Sul

ESTUDO E ANÁLISE DE MÉTODOS PARA RECONHECIMENTO DE PALAVRAS DITAS

Raiza Artemam de Oliveira
Willian Sousa Santos

Prof. MSc. André Chastel de Lima (Orientador)

DOURADOS-MS
2016

RESUMO

faca um resumo

Palavras-chave: Resumo. Palavras chaves . .

SUMÁRIO

Lista de siglas

ALSA - Advanced Linux Sound Architeture

API - Application Programming Interface

CDMA - Code Division Multiple Access

DCT - Discrete Cosine Transform

FFT - Fast Fourier Transform

GSM - Groupe Special Mobile

HMM - Hiden Markov Model

MFCC - Mel Frequency Cepstral Coefficients

PCM - Pulse Code Modulation

RIFF - Resource Interchange File Format

WAVE - Waveform Audio File Format

Lista de tabelas

Lista de ilustrações

1 INTRODUÇÃO

Nos primeiros sistemas computacionais a comunicação entre pessoas e máquinas era realizada através de terminais por linha de comando. Apenas especialistas conseguiam utilizar estes sistemas. Depois, no início da década de 70, com a criação do mouse e a introdução da interface gráfica os sistemas tornaram-se mais amigáveis ao usuário, podendo ser utilizados por pessoas comuns sem necessidade de conhecimento técnico. Com o passar dos anos a interação entre pessoas e máquinas tornou-se mais intuitiva com as diversas interfaces entre o usuário e o sistema. No fim da década de 70 iniciaram-se as pesquisas de reconhecimento de fala. Interfaces por meio de fala são utilizadas em diversas áreas, tais como: sistemas embarcados, automação residencial, operações bancárias, conversão fala texto e dispositivos móveis.

O reconhecimento da fala é um campo de estudo amplo e necessário as diversas tecnologias que utilizam desta como um meio de comunicação entre o usuário e o sistema. Utilizar a fala como entrada de um sistema torna a comunicação entre o usuário e o sitema mais direta, intuitiva, rápida e precisa.Como um campo de ampla aplicacação, o reconhecimento de fala tem diversos projetos em diferentes partes do mundo. Dentre os quais se destaca o projeto CMU Sphinx da universidade americana Carnegie Mellon. O projeto já tem cerca de 20 anos de pesquisas na área de reconhecimento de fala e de voz. Trata-se de um projeto open source voltado para linux, mas também conta com uma versão em java multiplataforma. O CMU Sphinx oferece suporte para várias linguagens, dentre elas o inglês, alemão, russo, francês e espanhol. O reconhecimento de fala pode ser classificado de acordo com o tamanho do vocábulario, de acordo com os algoritmos utilizados e de acordo com o tipo de fala a ser reconhecida (contínua ou discreta).

1.1 Justificativa

O reconhecimento de palavaras ditas é um campo de estudo de extrema importância para uma melhor comunicação entre usuários e sistema.

1.2 Objetivos

O objetivo deste trabalho é estudar os principais métodos de reconhecimento de fala. Analisar os algoritmos utilizados, suas vantagens e desvantagens. Apresentar os resultados para um pequeno vocábulario.

1.2.1 Objetivo geral

Estudar e analisar os algoritmos existentes para o reconhecimento de palavras ditas em um vocábulario pequeno e um ambiente não controlado.

1.2.2 Objetivo específico

Apontar a melhor solução para reconhecimento de palavras ditas em ambientes não controlados.

1.3 Metodologia

A metodologia adotada para a realização deste trabalho consiste nos seguintes passos:

- ❏ Pesquisa em livros, sites, artigos e notas de aula sobre o tema abordado e seus diversos aspectos;
- ❏ Estudo de algoritmos aplicados ao reconhecimento de fala;
- ❏ Implementação computacional de algoritmos aplicados ao reconhecimento de fala;
- ❏ Testes e validação dos algoritmos implementados;
- ❏ Análise e validação dos resultados obtidos com os métodos implementados;
- ❏ Documentação do trabalho.

No capítulo 2 é feita uma explicação do que é relevante para este trabalho com base na literatura.

2 FUNDAMENTAÇÃO TEÓRICA

De acordo com (RABINER; JUANG, 1993), os sistemas de reconhecimento de fala podem ser classificados em três grupos de acordo com a técnica utilizada. Estes grupos são :

- ❏ Reconhecedores por inteligência artificial;
- ❏ Reconhecedores por comparação de padrões;
- ❏ Reconhecedores baseados na análise acústico-fonética.

2.1 Sistemas de reconhecimento de fala

2.1.1 Reconhecedores baseados em inteligência artificial

Os sistemas de reconhecimento de fala que utilizam a inteligência artificial usa propriedades tanto dos reconhecedores por comparação de padrões quanto dos reconhecedores baseados na análise acústico-fonética. Sistemas com redes neurais são encaixados nesta classe. As redes Multilayer Perceptron usam uma matriz de ponderação que representa as conexões entre os nós da rede, e cada saída esta associada a uma unidade a ser reconhecida (MORGAN; SCOFIELD, 1991).

A abordagem de inteligência artificial se baseia no processo humano natural de ouvir, analisar e tomar uma decisão sobre as características acústicas medidas para reconher a fala. Faz parte do processo de reconheciemento de fala pela abordagem de inteligência artificial o processo de segmentação e rotulagem usado na análise acústico-fonética (RABINER; JUANG, 1993). Esta abordagem aplica o conceito de que o conhecimento é dinâmico e os modelos devem adaptar-se frequentemente.

2.1.2 Reconhecedores por comparação de padrões

Estes reconhecedores usam o príncipio de que o sistema foi treinado para reconhecer os padrões. Os sistemas por reconhecimento de padrões possuem duas fases diferentes :

- ❏ Treinamento;
- ❏ Reconhecimento.

Durante a fase de treinamneto são criados padrões de referência para o sitema. Na fase de reconhecimento compara-se os padrões obtidos com os padrões de referência criados na

fase anterior e calcula-se uma medida de similaridade entre os padões. O padrão mais similar ao desconheido é escolhido como reconhecido. Os sistemas que se baseiam nos Modelos Ocultos de Markov (HMM) se encaixam nesta categoria.

Dentre as diversas razões para usar a abordagem de comparação de padrões para reconheimento de fala podemos citar a simplicidade de uso, por ser um método de fácil entendimento que possui uma rica fundamentação matemática e é amplamente utilizado, e a robustez, trata-se de um método robusto e invariante para diferentes vocábularios, algoritmos de comparação de padrão e regras de decisão. Isto torna esta abordagem apropriada para uma vasta gama de unidades de fala, como fonemas, palavras isoladas ou frases (RABINER; JUANG, 1993).

2.1.3 Reconhecedores baseados na análise acústico-fonética

Os sistemas baseados na análise acústico-fonética decodificam o sinal de fala baseados nas características acústicas deste sinal e na relação entre elas (INCER, 1992). Os sistemas de análise desta classe devem considerar propriedades acústicas invariantes. Entre estas características estão a classificação entre sonoro e não sonoro, segmentção do sinal da fala, detecção das características que descrevem as unidades fonéticas e escolha do padrão que mais corresponde à sequencia de unidades fonéticas.

Os reconhecedores baseados na análise acústico-fonética trabalham em duas etapas. O primeiro passo na análise acústico fonética é chamado de fase de segmentação e rotulagem (RABINER; JUANG, 1993). Este passo envolve a segmentação do sinal da fala em regiões discretas, no tempo, onde as propriedades acústicas do sinal são representadas por um único fonema, ou estado. Em seguida uma ou mais etiqueta fonética é associada a cada região segmentada de acordo com as propriedades acústicas. O segundo passo para o reconhecimento tenta determinar uma palavra válida a partir da sequência de etiquetas fonéticas obtidas na fase anterior. As palavras são obtidas a partir de um determinado vocabulário, as palavras obtidas fazem sentido sintático e tem significado semântico.

2.2 Processamento digital de sinais

De acordo com (ORTIGUEIRA, 2005) um sinal é qualquer função associada a um fenômeno físico, econômico ou social e que transporta algum tipo de informação sobre ele.

Pode ser definido como uma descrição quantitativa de um dado fenômeno. A voz é um exemplo de sinal.

Os sinais podem ser classificados de diferentes formas de acordo com suas características e com o tipo de domínio e contradomínio. Segundo (ORTIGUEIRA, 2005) esta classificação pode ser feita de acordo com as seguintes características:

1. Variável independente: o sinal é contínuo se $t \in \mathbb{R}$ e discreto se $t \in \mathbb{Q}$. Os pontos $t_n, n \in \mathbb{Z}$ são chamados de instantes de amostragem. Sinal amostrado é o sinal discreto obtido por amostragem de um sinal contínuo.

2. Amplitude: os sinais podem ser classificados de acordo com a amplitude em :

   - ❏ Analógicos: sinal contínuo cuja amplitude pode assumir uma gama contínua de valores;
   - ❏ Quantificados: sinal cuja amplitude pode assumir, apenas, uma gama finita de valores;
   - ❏ Digitais: sinal resultante da codificação de um sinal amostrado e quantificado. A codificação consiste em atribuir a cada valor obtido por amostragem e quantificação um código.

3. Duração: os sinais cujo dominio é limitado dizem-se de duração finita, os restantes são de duração infnita.Os sinais de duração finita tambem são chamdos de janela.

4. Reprodutibilade: um sinal é dito determinístico se repetindo a mesma experiência obtém-se o mesmo resultado, caso isso não seja possível então trata-se de um sinal aleatório.

5. Periodicidade: os sinais determinísticos classificam-se ainda em aperiódicos e periódicos. Os sinais aperiódicos nao são repetitivos. Os sinais periódicos são repetitivos e possuem a relação $x(t) = x(t \mp T) \quad \forall \quad t$, onde T é o período. Quando $T < 2\pi$ a involvente final do sinal periódico $x(t)$ não coincide com a extensão periódica do sinal base $x_b(t)$ ocorre o fenômeno chamado *aliasing*. O fenômeno de aliasing é importante na conversão discreto-contínua e verifica-se no domínio da frequência.

6. Morfologia: formas simétricas a um eixo ou outro. Os sinais pares são simétricos ao eixo das ordenadas. Os sinais impares são simétricos ao eixo das abscissas.

7. Carater : outras medidas são consideradas. Um sinal pode ter carater escalar, vetorial. Por exemplo o sinal de saída de uma gregado de sensores é um sinal sensorial.

A análise frequencial moderna é um conjunto de técnicas matemáticas e ou físicas que permite obter o conteúdo frequêncial de qualquer sinal , a que se chama de espectro. O processo de obtenção de espectro chama-se análise espectral. O processo numérico usado para determinar o espectro é chamado de estimação espectral. A estimação espectral é feita em sinais de fonte fisica, como a voz, durante um intervalo de tempo finito. Na prática o conteúdo frequêncial de um dado sinal não é uniforme. Assume valores significativos em intervalos chamados bandas. A designação de filtro habitualmente usada em referência aos sistemas lineares, deriva da possibilidade de certos sistemas eliminarem ou atenuarem fortemente certas bandas. A Tabela 1 mostra alguns sinais e suas bandas.

Tabela 1: Bandas ocupadas por alguns sinais

Sinal	de	a
Eletrocardiograma	0 Hz	150 Hz
Eletroencefalograma	0 Hz	100 Hz
Voz	100 Hz	4000 Hz
Ruído do vento	100 Hz	1000 Hz
Ruído de tremor de terra	0.01 Hz	10 Hz
Radiodifusão	0.03 MHz	3 MHz
Onda curta	3 GHz	30 GHz
Radar, satélite, comun. espaciais	300 GHz	300 THz
Luz visível	370 THz	770 THz

3 CAPTURA DE ÁUDIO

A captura do sinal de áudio é uma parte fundamental para o desenvolvimento de um sistema reconhecedor de fala. Existem bases de dados disponíveis para testes em que a captura do sinal de áudio não é necessária, um vez que estas bases disponibilizam os arquivos de áudio. Um exemplo de base de dados de voz é a Aurora-1, esta base é construída por sinais de fala limpos e degradados através de oito tipos de ruídos (**??**). Neste trabalho optamos por realizar a captura do áudio pois este também faz parte do objetivo.

O som se propaga no ambiente por meio de ondas de forma contínua no tempo e no espaço a uma velocidade média de *340 metros/segundo* fazendo o ar vibrar. Esta onda sonora é capturada por meio de um microfone como uma onda analógia e é convertida para um sinal digital. A onda capturada é normalizada através de um filtro de passa-baixas. Circuitos que realizam esta conversão de onda são chamados de ADC (*analog digital converter*). O tamanho das amostras, expressa em bits, é um dos fatores que determina a precisão com que o som é representado em forma digital. Outro fator importante que afeta a qualidade de som é a taxa de amostragem. O teorema de Nyquist afirma que a frequência mais elevada que pode ser representado com precisão é, no máximo, metade da taxa de amostragem (PROAKIS; MANOLAKIS, 1996).

3.1 Bibliotecas para Captura de Áudio

Para o processo de reconhecimento de fala de qualquer tipo, primeiro é necessário capturar o sinal de áudio. A fase de captura de áudio é essencial para o bom desempenho do projeto. Existem diversas bibliotecas open-source que oferecem funções que realizam a captura e gravação de áudio, entre elas a Allegro e OpenGL, entretanto a aplicação dessas bibliotecas implica em um maior custo computacional, uma vez que estas trazem milhares de linhas de código junto com outras funções além das necessárias para a implementação deste projeto. Com base nisso, buscou-se uma alternativa que integrasse eficiência e baixo custo computacional para aplicações em áudio.

3.1.1 ALSA

ALSA (*advanced linux sound architeture*) consiste de um conjunto de drivers do kernel, uma biblioteca, uma API e programas utilitários para o suporte de som no linux. Jaroslav Kysela iniciou o projeto ALSA porque os drives de som do kernel Linux não estavam sendo devidamente mantidos e atualizados. Após a iniciativa mais desenvolvedores aderiram

ao projeto e a estrutura da API foi refinada. ALSA foi incorporada ao kernel oficial do Linux 2.5. A biblioteca fornecida pelo ALSA, libasound, fornece uma nomeação lógica dos dispositivos de hardware. Os nomes podem ser de dispositivos de hardware reais ou plugins (TRANTER, 2004). Os dispositivos de hardware usam o formato $HW : i, j$, onde i é o número do cartão e j do dispositivo do cartão. Uma placa de som tem um buffer de hardware que armazena amostras gravadas. Quando este buffer enche, ele gera uma interrupção. O driver de som do kernel, em seguida, utiliza o acesso direto à memória para transferir as amostras para um buffer de aplicativo na memória. O tamanho deste buffer pode ser programado por chamadas da biblioteca ALSA. Caso o buffer seja muito grande a tranferencia geraria uma latência excessiva. ALSA resolve isso dividindo o buffer em fragmentos e transfere os dados fragmentados. A Figura 1 ilustra a repartição do buffer em fragmentos, molduras e amostras.

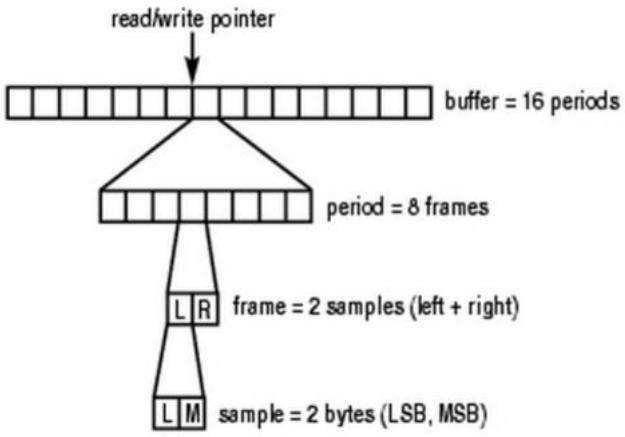

Figura 1: Buffer de aplicação. *fonte:(TRANTER, 2004)*

A API ALSA oferece seis principais interfaces. São elas a interface de controle, interface MIDI raw, interface de tempo, interface de sequência, interface mixer e interface de PCM. Esta última gerencia a captura e reprodução de áudio digital.

3.2 Arquivos WAVE

O formato de áudio adotado foi o WAVE. Neste tipo de formato o som é armazenado em sequências numéricas. O áudio é convertido em dados e armazenado bit a bit. O WAVE (.wav) foi criado pela IBM e pela Microsoft, nos anos oitenta e tem suporte a uma série de resoluções de bit, taxas de amostragens e canais de áudio. A taxa de amostragem em arquivo .wav refere-se ao número de amostras por segundo. O CD possui uma taxa de amsotragem de $44,100$, o que significa que cada segundo de áudio tem $44,100$ amostras. A quantidade

de bits usada determina quanta informação pode ser armazenada no arquivo. A quantidade de bits também interfere na amplitude do sinal. Em uma gravação de 8 bits estará disponível 256 níveis de amplitude, variando de 0 à 255. Em uma gravação de 16 bits a quantidade de níveis de amplitude disponíveis passa a $65,536$, variando entre $-32,768$ até 32767. A quantidade de 16 bits é suficiente para este projeto.

3.2.1 Cabeçalho WAVE

O cabeçalho de um arquivo .wav possui 44 bytes e é organizado como mostrado na Tabela 2.

Tabela 2: Formato de um cabeçalho de arquivo wave

Posição	Valor	Descrição
1 - 4	RIFF	Define como um arquivo RIFF
5 - 8	Tamanho do arquivo (int)	Tamanho máximo do arquivos em bytes
9 - 12	"WAVE"	Arquivo tipo cabeçalho wave
13 - 16	"fmt"	Marca formato chunk
17 - 20	16	Tamanho do formato dos dados
21 - 22	1	Formato tipo PCM
23 - 24	2	Quantidade de canais
25 - 28	44100	Taxa de amostragem
29 - 32	176400	(taxa de amostragem * bits por amostra * canais) / 8
33 - 34	4	limites
35 - 36	16	Quantidade de bits por amostra
37 - 40	data	Marca o início da seção de dados
41 - 44	Tamanho do arquivo (dados)	Tamanho da seção de dados

4 PRÉ-PROCESSAMENTO

O processo para reconhecimento de fala pode ser divido em várias etapas. O sinal de áudio é recebido do meio externo através de um transdutor e convertido para um sinal digital a partir deste momento devemos tratar este sinal. A Figura 2 ilustra as etapas do processo de extração de características MFCC.

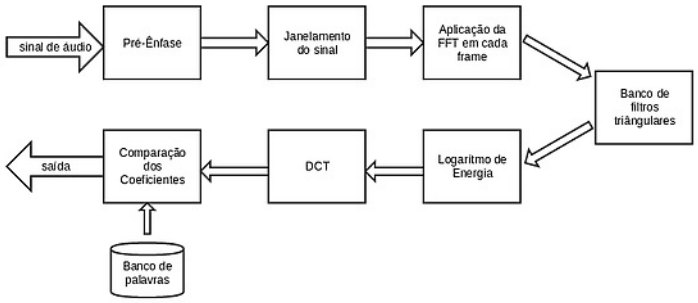

Figura 2: Etapas para extração de coeficientes MFCC. *fonte: Autoria própria*

O sinal recebido deve passar pelo pré-processamento para reduzir as interferências externas do sinal e ressaltar as informações úteis. Durante a etapa de pré-ênfase o sinal é normalizado. A normalização da amplitude do sinal garante que sons em diferentes alturas sejam processados igualmente. Os períodos de silêncio do sinal são retirados para que apenas dados importantes sejam armazenados.

Após a etapa de pré-ênfase é realizado o janelamento do sinal, ou seja, o sinal é dividido em frames. É aplicada uma janela de Hamming para atenuar as descontinuidades causadas no início e final de cada frame. A próxima etapa é a aplicação da Transformada Rápida de Fourier (FFT - do inglês *fast fourier transform*) no sinal. A equação 1 para obter a potência espectral.

$$S[k] = |X[k]|^2 = (real(X[k]))^2 + (imaginaria(X[k]))^2 \quad (1)$$

A FFT transforma um sinal do domínio do tempo para um do domínio da frequência. A Transformada Discreta de Fourier (DFT - do inglês *discret fourier transform*) possui complexidade $O(n^2)$ e a FFT possui complexidade $O(nlogn)$, por este motivo a FFT é usada em aplicações computacionais. A próxima etapa é a aplicação do banco de filtros triângulares, estes exigem uma explicação mais detalhada de como foram feitos. Esta explicação é feita em detalhes na seção **??**.

4.1 Filtros Triângulares

Para entendermos os filtros triângulares precisamos falar sobre a escala Mel.

4.1.1 Escala Mel

Em 1937 Stanley Smithy Stevens, John Volkman e Edwin Newmann propuseram o uso de uma variável psicoacústica chamada *pitch* para a criação de uma escala musical perceptual de tons em intervalos igualmente espaçados, chamada escala *mel*. A frequência ouvida pelo sistema auditivo humano é subjetiva e varia de acordo com cada indivíduo. Esta impressão subjetiva de frequência é a sensação subjetiva da intensidade ou a amplitude de um som. A escala *mel* é uma escala de pitches julgados pelos ouvintes como sendo igual em distância um do outro. O ponto de referência entre esta escala e a medição de freqüência normal é definida igualando um tom de 1000 Hz , 40 dB acima do limiar do ouvinte , com um pitch de 1000 *mels*. Abaixo de cerca de 500 Hz as escalas de *mel* e Hertz coincidem, acima disso intervalos cada vez maiores são julgados por ouvintes para produzir iteração igual aos pitches. A escala *mel* é baseada em um mapeamento entre a frequência real e o pitch aparentemente percebido do sistema auditivo humano. Para converter uma frequência em escala *mel* aplica-se a equação 2.

$$M(f) = 1125ln(1 + \frac{f}{700}) \tag{2}$$

A percepção humana de algumas frequências de sons complexos não podem ser individualmente dentro de certas bandas, quando uma dessas componentes cai fora da banda, chamada de banda crítica, ela pode ser identificada. Isto ocorre porque a percepção de uma frequência particular pelo sistema auditivo, por exemplo f_0, é influenciada pela energia da banda crítica das frequências em torno de f_0. O valor dessa banda varia nominalmente de 10% a 20% da frequência central do som, começando em torno de $100Hz$ para frequências abaixo de $1kHz$ e aumentando em escala logarítmica acima disso. Com base nestes fenômenos utiliza-se o logarítmo da energia total das bandas críticas em torno das frequências mel. A aproximação utilizada para este cálculo é a utilização de um banco de filtros espaçados uniformemente na escala mel, o banco de filtros triangulares. Os filtros *mel* são definidos de

acordo com a função 3.

$$H_m[k] = \begin{cases} 0 & k < k[m-1] \\ \dfrac{2(k-k[m-1])}{(k[m+1]-k[m-1])(k[m]-k[m-1])}, & k[m-1] \leq k \leq k[m] \\ \dfrac{2(k[m+1]-k)}{(k[m+1]-k[m-1])(k[m+1]-k[m])}, & k[m] \leq k \leq k[m+1] \\ 0 & k > k[m+1] \end{cases} \tag{3}$$

A Figura 3 mostra o banco de filtros usados na técnica MFCC. Cada filtro calcula a média do espectro em torno de um espectro central. Quanto maior a frequência, maior é a largura da banda.

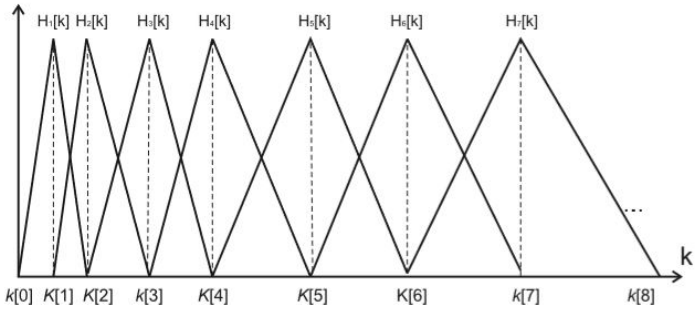

Figura 3: Banco de filtros triângulares MFCC. *fonte: (GORDILLO, 2013)*

Para determinar matematicamente os segmentos, parte- se da frequência extremas f_l e f_h que são as frequências de corte do banco de filtros em Hz. Esses valores são usados para dividir o intervalo em $B+1$ partes iguais. Para obter os valores em Hz, basta aplicar a função inversa 4.

$$k[m] = (\frac{N}{F_s})Mel^{-1}(Mel(f_l) + m\frac{Mel(f_h) - Mel(f_l)}{M+1}) \tag{4}$$

onde F_s é a frequência de amostragem em Hz, M é o número de filtros e N o número de amostras da FFT. $k[m]$ são as frequências digitais e Mel^{-1} determina a largura do banco de filtros e é dado por

$$Mel^{-1}(m) = 700(e^{\frac{m}{1125}} - 1) \tag{5}$$

Em seguida, obtém-se a log-energia da saída de cada um dos filtros *mel*. Por fim os coeficientes MFCC são obtidos aplicando a Transformada Discreta de Cosseno (DCT - do inglês *discret cosine transform*) ao logarítmo dos coeficientes de energia obtidos no passo anterior.

5 MODELOS OCULTOS DE MARKOV

Um modelo de Markov pode ser definido como um conjunto finito de estados ligados entre si por transições, formando uma máquina de estados. Estas transições estão ligadas por um processo estocástico . Há ainda um outro processo estocástico associado a um modelo de Markov, que envolve as observações de saída de cada estado. Se somente as observações de saída forem visíveis a um observador externo ao processo, diz-se então que os estados estão ocultos.

Um HMM é caracterizado por:

- ❏ Um conjunto de estados $S = \{S_1, S_2, \ldots, S_{n-1}, S_n\}$, onde n é o número de estados;
- ❏ Função de probabilidade de estado inicial $\pi = \{\pi_i\}$.

$$\pi_i = P[q_1 = S_i] \quad 1 \leq i \leq n \tag{6}$$

onde q_1 é o estado inicial $(t = 1)$.

- ❏ Função de probabilidade de transição A;
- ❏ Função de probabilidade de símbolos de saída B.

Considerando exclusivamente processos em que as probabilidades de transição não dependem do tempo e os HMMs são de primeira ordem, um HMM é considerado de primeira ordem quando a trasição do estado depende apenas da probabilidade do estado anterior mais recente. O conjunto de probabilidades de transição A é definido por:

$$A = \{a_{ij}\} \tag{7}$$

$$a_{ij} = P[q_{t-1} = S_i][q_t = S_j] \quad 1 \leq i, j \leq n \tag{8}$$

onde a_{ij} é a probabilidade de ocorrer uma transição do estado S_i para o estado S_j.
Os coeficientes a_{ij} devem obedecer às seguintes regras:

$$a_{ij} \geq 0 \quad 1 \leq i, j \leq n \tag{9}$$

$$\sum_{j=1}^{n} a_{ij} = 1 \quad 1 \leq i \leq n \tag{10}$$

A probabilidade de estar no estado S_j no instante de tempo t depende somente do instante de tempo t_1.

5.1 HMM e a função densidade de probabilidade

Um HMM também pode ser classificado de acordo com a função densidade de probabilidade.

5.1.1 Função densidade de probabilidade

Uma variável aleatória é uma função cujo valor é um número real determinado por cada elemento em um espaço amostral. Dada uma variável aleatória X, dizemos que $f(x)$ é uma função densidade de probabilidade de X, se e somente se $f(x)$ atender as seguintes condições:

$$f(x) \geq 0 \qquad a < x < b$$

$$\int_a^b f(x)dx = 1$$

5.1.2 HMM Discreto

O número de possíveis símbolos de saída é finito (RABINER; JUANG, 1993). A probabilidade de emitir o símbolo V_k no estado S_i é dada por $b_i(k)$. As propriedades da função de probabilidade B são:

$$b_i(k) \geq 0 \qquad 1 \leq i \leq n \qquad 1 \leq k \leq K$$

$$\sum_{k=1}^{K} b_i(k) = 1 \qquad 1 \leq i \leq n$$

As observações são discretas por natureza ou discretizadas através de uma técnica de quantização vetorial, gerando assim codebooks.

5.1.3 HMM Contínuo

A função densidade de probabilidade é contínua. Geralmente uma função densidade elipticamente simétrica, tal como a função densidade de probabilidade Gaussiana (RABINER;

JUANG, 1993). As observações são contínuas e a FDP contínua é usualmente modelada como uma mistura finita de matrizes gaussianas multidimensionais.

****

5.1.4 HMM Semicontínuo

O modelo é um caso intermediário entre contínuo e o discreto. O conjunto função densidade probabilidade é o mesmo usado para todos os estados e todos os modelos. A probabilidade de emissão dos simbolos de saída é dada por :

$$b_j(O_t) = \sum_{V_k \in \eta(O_t)}^{-} c_j(k) f(O_t|V_k) \qquad 1 \leq j \leq n$$

onde:

O_t é o vetor de entrada

$\eta(O_t)$ é o conjunto das funções densidade de probabilidade que apresentam os M maiores valores de $f(O_t|V_k)$, $1 \leq M \leq K$

K é o número de funções densidade de probabilidade, ou seja, os símbolos de saída

V_k é o k-ésimo símbolo de saída

$c_j(k)$ é a probabilidade de emissão do símbolo V_k no estado S_j

$f(O_t|V_k)$ é o valor da k-ésima função densidade de probabilidade.

5.2 Topologia

Uma maneira de classificar um HMM é de acordo com a estrutura de transição da matriz A da cadeia de markov. Existem vários modelos de HMM, tal como o ergódico totalmente conectado onde qualquer estado pode ser alcançado com um único passo, o modelo de caminhos paralelos e o modelo "left-right", também chamado de modelo Bakis. Para o reconhecimento de fala este último é o mais usado (RABINER; JUANG, 1993).

5.3 Os problemas a serem resolvidos

O HMM possui três problemas básicos a serem solucionados que são:

1. Problema de avaliação: Dada a sequência de observação $O = (o_1, o_2, o_3, ..., o_n)$ e o modelo $\lambda = (A, B, \pi)$, como calcular eficientemente $P(o|\lambda)$.

2. Problema da busca da melhor sequência de estados.

3. Problema de treinamento: como ajustar os parâmetros do modelo $\lambda(A,B,\pi)$ para maximizar $P(o|\lambda)$.

O problema 1, ou seja, o problema da avaliação pode ser solucionado através do procedimento *Forward-Backward*. O segundo problema é solucionado com a aplicação do algoritmo de *Viterbi* e o terceiro e último problema pode ser otimizado aplicando um procedimento iterativo como o método de *Baum-Welch*. Nas seções 5.3.1, 5.3.2 e 5.3.3 faz-se uma explicação sobre os procedimentos para a solução dos problemas 1, 2 e 3 respectivamente.

5.3.1 Foward-Backward

texto

5.3.2 Viterbi

O algoritmo de Viterbi é um algoritmo de programação dinâmica usado para encontrar a sequência de estados ocultos ótima. Dado uma sequência de estados ocultos de um HMM, o algoritmo de viterbi calcula a melhor sequência de estados baseados nas probabilidades de transição. Este algoritmo foi proposto em 1967 por Andrew Viterbi para a decodificação de códigos convolucionais em links de comunicação ruidosos. O algoritmo também possui aplicações em redes CDMA e GSM, modem dial-up, satélites, síntese de fala, linguística computacional e bioinformática. Em telecomunicação, um código convolucional é um tipo de código corretor de erro em que cada conjunto de *m* símbolos é transformado em um conjunto de *n* símbolos.

Algoritmo

- Inicialização:

  $\delta_1(i) = \pi_i b_i(O_1), \qquad 1 \leq i \leq N$

  $\Psi_1(i) = 0$

- Recursão:

  $\delta_t(j) = \max\limits_{1\leq i\leq N}\left[\delta_{t-1}(i)a_{ij}\right] b_j(O_t), \qquad 2 \leq t \leq T$

  $\Psi_t(j) = arg \max\limits_{1\leq i\leq N}\left[\delta_{t-1}(i)a_{ij}\right], \qquad 1 \leq j \leq N$

- Término:

  $P^* = \max\limits_{1\leq i\leq N}\left[\delta_{T(i)}\right]$

  $G_T^* = arg \max\limits_{1\leq i\leq N}\left[\delta_{T(i)}\right]$

Para ilustrar melhor o algoritmo de Viterbi vejamos um exemplo: um cientista nunca sai de casa e não tem nenhum contato com o mundo exterior, com excessão do seu gato. Todos os dias o gato sai para passear. O cientista observou o estado em que o gato voltava para casa durante quatro dias seguidos. No primeiro dia o gato voltou seco, no segundo e terceiro dia o gato retornou molhado e no quarto dia o gato retornou seco. Com base nessas observações o cientista gostaria de dizer se o dia estava ensolarado, nublado ou chuvoso. Temos então que a sequência de estados observáveis é $O = \{seco, molhado, molhado, seco\}$. E o vetor de probabilidade inicial é : $\pi = [0,5 \quad 0,3 \quad 0,2]$. A matriz de transição A e a matriz de probabilidade de emissão B são:

$$A = \begin{bmatrix} 0,5 & 0,3 & 0,2 \\ 0,2 & 0,3 & 0,5 \\ 0,1 & 0,2 & 0,7 \end{bmatrix}$$

$$B = \begin{bmatrix} 0,7 & 0,3 \\ 0,5 & 0,5 \\ 0,1 & 0,9 \end{bmatrix}$$

A inicialização do algoritmo:

$\delta_1(1) = 0,5 * 0,7 = 0,35$

$\delta_1(2) = 0,3 * 0,5 = 0,15$

$\delta_1(3) = 0,2 * 0,3 = 0,02$

$\Psi_1(1) = 0 \quad \Psi_1(2) = 0 \quad \Psi_1(3) = 0$

Passo de recursão:

Para $t = 2$ e $j = 1$:

$\delta_2(1) = \max_{1 \leq i \leq N} \left[\delta_1(i) a_{i1} \right] b_1(O_2)$

$\delta_1(1)_{a_{11}} = 0,35 * 0,5 = 0,175$

$\delta_1(2)_{a_{21}} = 0,15 * 0,2 = 0,03$

$\delta_1(3)_{a_{31}} = 0,02 * 0,1 = 0,002$

Substituindo o δ encontrado temos:

$\delta_2(1) = 0,175 * 0,3 = 0,0525$

$\Psi_2(1) = 0$

Para $t = 2$ e $j = 2$:

$\delta_2(2) = \max_{1 \leq i \leq N} \left[\delta_1(i) a_{i2} \right] b_2(O_2)$

$\delta_1(1)_{a_{12}} = 0,35 * 0,3 = 0,105$

$\delta_1(2)_{a_{22}} = 0,15 * 0,3 = 0,045$

$\delta_1(3)_{a_{32}} = 0,02 * 0,2 = 0,004$

Substituindo o δ encontrado temos:

$\delta_2(2) = 0,105 * 0,5 = 0,0525$

$\Psi_2(2) = 0$

Para $t = 2$ e $j = 3$:

$\delta_2(3) = \max_{1 \leq i \leq N} \left[\delta_1(i) a_{i3} \right] b_3(O_2)$

$\delta_1(1)_{a_{13}} = 0,35 * 0,2 = 0,07$

$\delta_1(2)_{a_{23}} = 0,15 * 0,5 = 0,075$

$\delta_1(3)_{a_{33}} = 0,02 * 0,7 = 0,014$

Substituindo o δ encontrado temos:

$\delta_2(3) = 0,075 * 0,3 = 0,0525$

$\Psi_2(3) = 1$

Para $t = 3$ e $j = 2$:

$\delta_3(1) = \max_{1 \leq i \leq N} \left[\delta_2(i) a_{i2} \right] b_2(O_3)$

$\delta_2(1)_{a_{11}} = 0,0525 * 0,3 = 0,01575$

$\delta_2(2)_{a_{21}} = 0,0525 * 0,3 = 0,01575$

$\delta_2(3)_{a_{31}} = 0,0675 * 0,2 = 0,0135$

Substituindo o δ encontrado temos:

$\delta_3(2) = 0,01575 * 0,5 = 0,007875$

$\Psi_3(2) = 0$

Para $t = 3$ e $j = 1$:

$\delta_3(1) = \max_{1 \leq i \leq N} \left[\delta_2(i) a_{i1} \right] b_1(O_3)$

$\delta_2(1)_{a_{13}} = 0,0525 * 0,5 = 0,02625$

$\delta_2(2)_{a_{23}} = 0,0525 * 0,2 = 0,0105$

$\delta_2(3)_{a_{33}} = 0,0675 * 0,1 = 0,007875$

Substituindo o δ encontrado temos:

$\delta_3(1) = 0,02625 * 0,3 = 0,007875$

$\Psi_3(1) = 1$

Para $t = 3$ e $j = 3$:

$\delta_2(2) = \max_{1 \leq i \leq N} \left[\delta_2(i) a_{i3} \right] b_3(O_3)$

$\delta_2(1)_{a_{11}} = 0,0525 * 0,2 = 0,0105$

$\delta_2(2)_{a_{21}} = 0,0525 * 0,5 = 0,02625$

$\delta_2(3)_{a_{31}} = 0,0675 * 0,7 = 0,04725$

Substituindo o δ encontrado temos:

$\delta_3(3) = 0,04725 * 0,9 = 0,042525$

$\Psi_3(3) = 2$

5.3.3 Baum-Welch

texto

REFERÊNCIAS BIBLIOGRÁFICAS

GORDILLO, C. D. A. **Reconhecimento de Voz Contínua Combinando os Atributos MFCC e PNCC com Métodos de Robustez SS, WD, MAP e FRN**. Dissertação (Mestrado) — PUC-RJ, 2013.

INCER, A. N. **Digital Speech Processing, Speech Coding, Syntesis and Recognition**. [S.l.]: Kluwer Academic Publishers, 1992.

MORGAN, D. P.; SCOFIELD, C. L. **Neural Networking and Speech Processing**. [S.l.]: Kluwer Academic Publishers, 1991.

ORTIGUEIRA, M. D. **Processamento Digital de Sinais**. [S.l.]: Fundação Calouste Gulbenkian: Lisboa, 2005.

PROAKIS, J. G.; MANOLAKIS, D. G. **Digital Signal Processing. Principles, Algorithms and Applications**. [S.l.]: Prentice-Hall: New Jersey, 1996.

RABINER, L. R.; JUANG, B. H. **Fundamentals of Speech Recongnition**. [S.l.]: Prentice-Hall, 1993.

TRANTER, J. Introduction to sound programming with alsa. **Linux Journal**, 2004. Disponível em: <http://www.linuxjournal.com/article/6735>. Acesso em: 10.4.2015.