Uso de Regras Fonológicas com Determinação de Vogal Tônica para Conversão Grafema-Fone em Português Brasileiro

Ana Carolina Siravenha, Nelson Neto, Valquíria Macedo e Aldebaro Klautau

***Abstract*—A conversão de uma seqüência de caracteres em seqüências de fones é um importante pré-requisito para serviços que envolvem reconhecimento e/ou síntese de voz. Contudo, a tarefa não é trivial e diversas técnicas de conversão vêm sendo adotadas ao longo da última década. Existe um número bem menor de estudos na área dedicados ao Português Brasileiro (PB) quando comparado ao Inglês, por exemplo. Este trabalho discute esforços para reduzir esta deficiência, enfatizando-se a conversão grafema-fone, e apresenta um sistema para reconhecimento de voz em PB usando o novo corpus West Point. Os seguintes recursos encontram-se disponíveis: HTK scripts, dicionário fonético, modelos acústico e de linguagem.**

***Index Terms*—Reconhecimento de voz, Grafema, Fone, Vogal Tônica, Dicionário Fonético.**

I. Introdução

O interesse por técnicas de conversão grafema-fone (G2P) vem de séculos, fato comprovado pelas descrições não sistemáticas feitas em estudos gramaticais da língua. A elaboração de um dicionário fonético de grande vocabulário, do qual são extraídas transcrições fonéticas de todas as palavras contidas no corpus usado para treino e teste, é um dos pontos fundamentais para desenvolvimento de sistemas conhecidos na literatura como *large vocabulary continuous speech recognition* (LVCSR). A construção de um dicionário de pronúncias para sistemas de reconhecimento automático de voz (ASR) é muito similar ao desenvolvimento de um módulo G2P para sistemas de síntese de voz [1].

Os sistemas G2P podem ser organizados em métodos baseados em regras e *data-driven*. Em [2], a combinação dos dois métodos foi implementada pela compilação de regras fonológicas e de determinação de vogais tônicas para o Português Europeu, usando a flexibilidade de integração de múltiplas fontes de informação dos transdutores de estados finitos. O conjunto de regras para conversão G2P no português é extremamente amplo. Em função disso, diversos estudos têm enfatizado o uso de algoritmos de aprendizagem para obtenção automática de regras G2P, como redes neurais de múltiplas camadas [3], algoritmos genéticos [4] e por indução TBL, ou *Transformation-Based Learning*, discutida em [5] e com aplicações em outras áreas, como correção ortográfica e codificação de diálogo.

O sistema de anotação fonética automático proposto por [6] submete o corpus EUROM_1 a quatro estágios. No primeiro, há uma conversão G2P "larga", que muitas vezes não corresponde ao que foi realmente falado. Em seguida, outras transcrições alternativas são realizadas, ficando a cargo de um decodificador de fones, que usa algoritmo de Viterbi, escolher a melhor transcrição na quarta etapa. De acordo com a aplicação desejada, é feita uma extração de características que serão parametrizadas usando *Mel-Frequency Cepstral Coefficients* (MFCC), no penúltimo estágio. Foi mostrado por [6] que os testes de alinhamento têm seu desempenho otimizado quando os modelos HMM são treinados especificamente para essa tarefa, enquanto que para a transcrição os melhores resultados são encontrados quando os modelos são treinados para decodificação ou reconhecimento. Além disso, [6] concluiu que a aplicação sucessiva de regras melhora significativamente a precisão da transcrição fonética.

A estratégia usada em [7] foi de induzir um aumento de contraste no cliticismo gramatical das palavras, fazendo com que as vogais à esquerda da vogal tônica se tornassem mais fracas. Por não ser direcionado à análise de palavras isoladas, o trabalho se baseia em análises gráficas para que seja feita a segmentação das palavras, definindo sílabas iniciais como orais e sílabas intermediárias como fracas. Trabalhando igualmente com palavras não-isoladas, [8] usa um algoritmo baseado em regras para determinação de vogais tônicas e para conversão G2P, estratégia que favorece a inserção de novas palavras à base sem abrir mão da eficiência.

O trabalho de [9] descreve um conversor G2P baseado em métodos de aprendizado de máquina. Foram investigados os métodos de aproximação *Memory Based Learning* e *Transformation Based Learning*, sendo que a aproximação híbrida retornou o melhor resultado, baseando-se nas taxas *Word Error Rate* (WER), *Phone Error Rate* (PER) e *Mean Normalized Levenshtein Distance* (MNDL). Tal pesquisa exalta a importância da informação silábica e do tamanho do corpus de treino.

Abordando a influência das variantes léxicas do PB, [10] esclarece os processos fonológicos, próprios das línguas naturais, como inserção ou apagamento de segmentos, assimilação segmental e estruturação silábica. Os resultados obtidos são comparados a um *software*, desenvolvido com base em regras que descrevem as variantes para as palavras do léxico de um sistema ASR para PB, com as transcrições geradas manualmente. Essa última pode ser, eficientemente, substituída pelas transcrições obtidas automaticamente.

A comunidade internacional beneficia-se de programas que promovem cooperações e até competições livres entre grupos

Laboratório de Processamento de Sinais (LaPS), DEEC, Universidade Federal do Pará (UFPA), Rua Augusto Correa, 1, CEP 660750-110, Belém, PA, Brasil. E-mails: {siravenha,nelsneto,valquiria,aldebaro}@ufpa.br.

de pesquisadores [11]. Porém, são tentativas isoladas e, atualmente, não existem *softwares recipes* de domínio público. Este trabalho, através da disponibilização de seus *scripts* [12], busca quebrar esse paradigma, permitindo a reprodução dos resultados em diferentes localidades. Nosso primeiro esforço foi a criação de um dicionário para PB com 11.827 palavras transcritas manualmente [13]. Esse conjunto de transcrições foi então utilizado como base de treino de um módulo de conversão G2P capaz de extrair automaticamente regras de transcrição a partir de um léxico através de técnicas de aprendizado de máquina por indução (árvore de decisão) e Naïve Bayes. O resultado foi a criação e disponibilização de um dicionário fonético de grande vocabulário para PB com aproximadamente 60.000 palavras.

Esta atual proposta contribui com um algoritmo baseado em regras para conversão G2P com determinação de vogal tônica para PB. Uma vantagem dos conversores baseados em regras é que o alinhamento lexical (dos grafemas) não se faz necessário, visto que o *software* não precisa ser treinado para gerar suas próprias regras. Ou seja, as propostas de conversão, baseadas em critérios fonológicos pré-estabelecidos, são fornecidas ao sistema de acordo com a língua a qual o aplicativo se destina. Em um segundo momento neste trabalho, o desempenho do dicionário fonético resultante é avaliado e comparado com outras metodologias para conversão G2P em um sistema ASR utilizando cadeias escondidas de Markov (HMMs) e a base de áudio West Point.

O presente trabalho encontra-se organizado da seguinte maneira. Na Seção II, são descritas as estratégias utilizadas para a conversão G2P. A Seção III apresenta o corpus West Point, assim como o *front end*, o modelo acústico baseado em HMM e o modelo de linguagem, identificando os esforços para o suporte ao PB. Já a Seção IV avalia os resultados obtidos e a Seção V conclui e sugere pesquisas futuras.

II. Sistemas Baseados no Conhecimento

Dentre as técnicas de conversão G2P encontradas na literatura, duas têm se destacado no âmbito daquelas baseadas no conhecimento: *data-driven* e baseadas em regras. A primeira técnica vale-se do aprendizado de máquina para a geração automática de regras a partir de um dicionário de treino, enquanto que a segunda, geralmente amparada por um lingüista, faz uso de regras pré-definidas para conversão grafema-fone.

A. Data-driven

O dicionário fonético *data-driven* utilizado neste trabalho foi construído por [13]. Inicialmente, foram geradas 11.827 transcrições ortográficas com 34 fones para o PB com base em modificações do alfabeto fonético SAMPA [14]. Em um segundo momento, essa versão foi usada para o treinamento de uma árvore de decisão J4.8 [15] através da ferramenta de aprendizado de máquina WEKA [16], seguindo o procedimento descrito em [13]. Assim, um novo dicionário foi obtido com as palavras mais freqüentes encontradas no *corpus* CETENFolha [17], aproximadamente 60.000 palavras. Aquelas palavras que não fazem parte das 11.827 transcrições originais não foram validadas individualmente, mas sim por meio de experimentos de reconhecimento de voz.

B. Baseado em Regras

A estrutura de regras aqui utilizada baseou-se nas regras G2P com determinação de vogal tônica descritas em [8]. Sua arquitetura é *self-contained*, ou seja, não carece de estágios intermediários, nem depende de outros algoritmos, para realizar análises específicas, como divisão silábica ou identificação de pluralidade. Existe uma ordem obrigatória para aplicação das regras. Primeiro são analisadas as regras consideradas mais específicas e, por último, a regra, ou caso geral, que finaliza a análise. Nenhuma análise co-articulatória entre palavras foi realizada, já que este processo de conversão G2P trabalhou apenas com palavras isoladas.

O algoritmo proposto foi desenvolvido em linguagem C# e usa 29 expressões regulares para determinação de vogais tônicas e 140 expressões regulares definidas para conversão G2P em 38 fones do alfabeto SAMPA [14]. Primeiramente, cada palavra do vocabulário deve ser expressa na forma: #abacaxi#, ou seja, delimitada pelo símbolo #, deixando claro para o analizador onde começa e onde termina cada palavra. Em seguida, é feita a análise para determinação da vogal tônica. O código abaixo exemplifica a estrutura da expressão regular usada na análise da palavra "abacaxi", que retorna a vogal *i* como tônica. Um objeto "Regex" é criado (a sintaxe é descrita em tutoriais sobre a linguagem C#) com o padrão a ser encontrado dentro da palavra analisada. Já o objeto "Match" recebe a resposta da comparação do padrão com a palavra apresentada. Sendo verdadeira, a vogal tônica é determinada, caso contrário, outras regras são testadas até que se esgotem as possibilidades e o caso geral seja aplicado. Um exemplo é mostrado a seguir.

```
Regex rule_8=new Regex("[^aeiou][iu][#]");
Match m8 = rule_8.Match(word);
if(m8.Success) {
    pos = m8.Index;
    strVw = word.Substring(pos+1,1);
    break;
}
```

De posse dessa informação, o próximo passo é a conversão grafema-fone, que segue a ordem seqüencial da palavra (*left-to-right*).

```
letter = word.Substring(index,1);
Regex idA = new Regex("a");
Match gA = idA.Match(letter);
if(gA.Success) {
    letter[index] = "a";
    index++;
}
```

É importante atentar que a ordem de precedência escolhida para o funcionamento do algoritmo não é casual. Para determinadas palavras, a presença de uma vogal tônica em determinada posição muda a interpretação da conversão, p.e.:

```
<e(V_ton)><l><C-h,Pont>  - [E]

<e(V_aton)><l><C-h,Pont> - [e]
```

A conversão é temporariamente armazenada em um arranjo

de *strings*, até que todos os grafemas da palavra sejam analisados. Por fim, a palavra e a sua respectiva conversão grafema-fone são escritas na forma abaixo:

```
abacaxi a b a k a S i sp
```

que é o formato sugerido pelo *software* HTK, onde o fone *sp* é o símbolo delimitador *short-pause*.

No decorrer dos testes realizados, constatou-se a necessidade de adaptação de algumas regras propostas por [8]. Algumas normas sugeridas que trabalham a nasalidade dos grafemas *a* e *u* não analisam se o grafema seguinte é uma vogal ou consoante. No entanto, verificou-se que essa distinção é importante no momento da conversão G2P. Por exemplo, segundo [8], a palavra "adotando", onde o grafema tônico nasal *a* é seguido da consoante *d*, seria convertida como abaixo:

```
adotando a d o t a~ n d u sp
```

Nota-se que a transcrição do fone *n* deveria ser desconsiderada, já que o fone $a\sim$ já representa a nasalidade do grafema *a*. Assim, após a adaptação descrita na primeira linha da Tabela I, a palavra usada como exemplo foi convertida da seguinte forma:

```
adotando a d o t a~ d u sp
```

Ainda sobre o grafema *u*, observou-se a inexistência de condições para verificar se o mesmo deve, ou não, ser classificado foneticamente como uma semivogal, representada aqui pelo fone *w*. Com isso, três novas regras foram elaboradas e incluídas antes da regra geral do grafema *u*.

Além da exceção feita à palavra "gratuito(a)", que não segue a primeira regra do grafema *i* fornecida por [8], outro ponto que se mostrou carente de análise foi o grafema *x*. O trabalho tomado como referência trata tal grafema apenas com exceções, ou seja, analisando palavra por palavra. Já esta pesquisa conserva as exceções, mas contribui com 17 regras fonológias para o grafema *x*. Por sua vez, todas as regras G2P adicionadas e/ou modificadas encontram-se listadas na Tabela I.

Com relação às regras para determinação de vogal tônica, fez-se necessário a inserção de apenas uma norma, no caso para o monossílabo "que", que assim como a palavra "quem", não se encaixa em nenhuma das regras propostas por [8].

TABLE I
ADAPTAÇÕES REALIZADAS ÀS REGRAS G2P PROPOSTAS POR [8] .

Grafema	Regra	Seqüência para o algoritmo	Fone
a	3	...(a(V_ton))(m,n)(V,h)...	[a∼]
u	2	...(u(m,n))(C-h)...	[u∼]
u	4	...(u)(m,n)(V,h)...	[u∼]
u	5	...(V-u)(u)...	[w]
u	6	...(q,g)(u)(a)...	[w]
u	7	...(g)(u)(o)...	[w]
x	1	...(V,C-f,m) (i) (x)...	[S]
x	2	...(f,m) (i) (x)...	[k s]
x	3	...((W_bgn)e,ê) (x) (V,C_v)...	[z]
x	4	...((W_bgn)ine) (x) (o,C_v)...	[k s]
x	5	...((W_bgn)ine) (x) (a,e,i)...	[z]
x	6	...((W_bgn)(e,ê,ine)) (x) (C_uv)...	[s]
x	7	...((W_bgn)e) (x) (Hf) (V,C_v)...	[z]
x	8	...((W_bgn)e) (x) (Hf) (C_uv)...	[s]
x	9	...(V-e) (x) (Hf) (Ltr)...	[k z]
x	10	...(V-e) (x) (V)...	[k s]
x	11	...(b,f,m,p,v) (e) (x) (V)...	[S]
x	12	...(V) (e) (x) (V)...	[z]
x	13	...(C-b,f,m,p,v) (e) (x) (V)...	[k s]
x	14	...((W_bgn)x)...	[S]
x	15	...(e,é,ê) (x) (C)...	[s]
x	16	...(x) (Pont)...	[k s]
x	17	...(x)...	[S]

III. ASR COM WEST POINT CORPUS

O *West Point Brazilian Portuguese Speech* é um corpus de gravações digitais para PB projetado e recolhido pelos funcionários e professores do Departamento de Línguas Estrangeiras (DFL) e do Centro de Tecnologia para Aprendizado de Língua Estrangeira (CTELL), instituições ligadas ao governo dos E.U.A, no intuito de desenvolver modelos acústicos para sistemas de reconhecimento de voz. O corpus vem sendo distribuído pela *Linguistic Data Consortium* com o catálogo LDC2008S04 [18] e consiste de sentenças lidas por 60 mulheres e 68 homens, nativos e não-nativos. As sentenças gravadas via microfone se resumem a 296 frases e expressões tipicamente usadas em aprendizado de línguas.

A. Preparação dos Dados

A preparação dos dados é um estágio essencial para qualquer projeto de desenvolvimento de reconhecedor de voz. Duas fontes de informação são necessárias: voz digitalizada e transcrita, ao nível de palavras e/ou ao nível de fonemas. Esta pesquisa se propõe a fornecer detalhes para a implementação de recursos que são específicos ao PB usando o *software* de domínio público HTK [19].

Vale a pena ressaltar que o corpus West Point possui algumas restrições, como ausência de transcrições fonéticas e ortográficas para algumas gravações, e nenhuma com alinhamento temporal. Outro aspecto problemático é a existência de arquivos de áudio com falhas, como ruídos, fala não clara, etc. Assim, uma etapa de pré-processamento foi realizada e 7.920 arquivos de voz com locutores nativos foram selecionados.

Então, esses arquivos foram divididos em dois conjuntos distintos: treino e teste. Nos experimentos realizados a base de treino foi composta por 6.334 arquivos, que corresponde a 384 minutos de gravação, e a base de teste com os restantes 1.586 arquivos de áudio, com um total de 96 minutos. Além das contribuições citadas anteriormente, este trabalho fornece um total de 68 transcrições ortográficas, até então inexistentes no corpus West Point, que podem ser livremente encontradas em [12].

B. Front end e Modelo Acústico

No processo de extração de parâmetros (*front end*) foram utilizados os consagrados MFCCs com 12 parâmetros estáticos, a energia e estimativas das duas primeiras derivadas, compondo um total de 39 parâmetros por quadro com duração de 20 milisegundos (ms) e deslocamento de 10 ms.

O modelo acústico foi iterativamente refinado [20]. Começando com modelos monofones compostos por uma única Gaussiana por mistura, as HMMs foram gradualmente

expandidas até a composição de um sistema com trifones vinculados e distribuições com múltiplas misturas. No modelo acústico inicial foram utilizados monofones e um modelo silêncio com 3 estados *left-to-right* para cada HMM. O modelo silêncio foi treinado e então copiado para criar o modelo vinculado *short pause* com apenas um estado acústico [19]. Em seguida, utilizou-se o método *flat-start* para inicialização dos parâmetros e a re-estimação de Baum-Welch para o treinamento dos monofones.

A seguir, foram construídos modelos trifones a partir dos modelos monofones. A técnica utilizada para construir os trifones, também conhecidos como dependentes do contexto, foi a *word-internal*, onde o contexto além das fronteiras das palavras não são considerados. Cada trifone foi clonado a partir do monofone que constitui seu fonema central. As matrizes de transição dos trifones que compartilham do mesmo fonema central foram vinculadas. Assim, os modelos trifones foram submetidos ao algoritmo de Baum-Welch.

Um problema clássico dos modelos dependentes do contexto é a ausência de material de treino suficiente para suportar a grande quantidade de trifones gerados [21]. Para contornar esse problema é essencial o compartilhamento de parâmetros. Dado um conjunto de características (também chamadas de questões [19]), uma árvore de decisão foi elaborada para vincular os trifones através de estados compartilhados (*tied-states*) por terem características fonéticas similares. Por exemplo, algumas questões usadas na construção da árvore de decisão são mostradas abaixo. Essa lista de questões também se encontra disponível.

```
QS "R_V-Fechada" { *+i,*+e,*+o,*+u }
QS "R_V-Front"   { *+i,*+E,*+e }
QS "R_Palatais"  { *+S,*+Z,*+L,*+J }
QS "L_V-Back"    { u-*,o-*,O-* }
QS "L_V-Aberta"  { a-*,E-*,O-* }
```

Nota-se que para um sistema trifone, é importante a inclusão de questões referentes aos contextos direito e esquerdo de cada fone. As questões devem avançar a partir de características gerais (como consoantes, vogais, ditongos, etc.) até particularidades mais específicas. Teoricamente, o conjunto de questões carregado através dos comandos *QS* deveria incluir todas as características lingüísticas ou fonéticas que possam influenciar acusticamente o contexto de um determinado fone.

Após o processo de compartilhamento (*tying*), os modelos trifones foram novamente re-estimados através do algoritmo de Baum-Welch.

C. Modelo de Linguagem

O CETENFolha [17] é um corpus de cerca de 24 milhões de palavras em PB, baseado em textos do jornal *Folha de S. Paulo* compilado pelo NILC/São Carlos, Brasil. O corpus original foi adaptado para que pudesse ser utilizado pelo pacote de ferramentas do *software* HTK. Alguns exemplos das operações de formatação são:

- Retirada de pontuação e *tags* ([ext], [t], [a] entre outras).
- Conversão para letras minúsculas.
- Expansão de números e acrônimos.
- Correção gramatical de palavras escritas incorretamente.

Um exemplo do resultado dessas operações é dado abaixo:

```
Antes: O Senado tem uma <<caixa preta>> de
       R\$ 1 milhão
Depois: o senado tem uma caixa preta de um
        milhão de reais
```

Esse experimento avalia a perplexidade (PP) do modelo de linguagem (ML) contra o número de sentenças usadas para treiná-lo. Utilizou-se a ferramenta *HBuild* para a construção do modelo de linguagem. O vocabulário usado foi mantidado constante durante todo o experimento, contendo 679 palavras presentes nas 296 frases do corpus West Point. O número de sentenças usadas para treinar o modelo de linguagem, composto por bigramas, variou entre 1.000 e 180.000 retiradas exclusivamente do corpus CETENFolha.

Através da ferramenta *HGen* as perplexidades das bigramas foram computadas usando 1.000 sentenças aleatoriamente escolhidas dentro da base de treino. Como era de se esperar, a perplexidade tende a diminuir com o aumento do número de sentenças usadas para treino [22]. Isso está relacionado com o fato de que as estatísticas dos modelos treinados melhoram a medida que ocorrências de pares de palavras são registadas no corpus de treino. A Tabela II mostra as perplexidades encontradas.

TABLE II
PERPLEXIDADE PARA DIFERENTES TAMANHOS DA BASE DE TREINO

	Número de sentenças usadas para treinar o ML							
	1k	10k	30k	60k	90k	120k	150k	180k
PP	44,9	33,3	26,2	21,7	19,2	16,4	15,7	14,8

IV. SIMULAÇÕES

Para avaliar a eficácia das regras G2P com determinação de vogal tônica, um dicionário fonético, composto pelas 679 palavras presentes no corpus West Point, foi construído através das regras e algoritmo apresentados na Seção II. Em seguida, esse dicionário foi usado para treinar um modelo acústico com 38 fones (HMMs) de acordo com os passos descritos na Seção III-B. Para efeito de comparação, o mesmo procedimento foi usado para elaborar outros dois modelos acústicos. O segundo modelo desenvolvido usou um dicionário fonético fiel às regras fonológicas descritas em [8], ou seja, sem considerar as alterações propostas neste trabalho. Já o último modelo acústico, composto por 34 HMMs, empregou o dicionário para grandes vocabulários construído em [13].

Um único modelo de linguagem bigrama foi montado usando apenas as frases do corpus West Point e perplexidade 28. O número de Gaussianas foi gradualmente aumentado até o limite de quatorze por mistura. A redução da WER para os três sistemas pode ser observada na Figura 1. Como era de se esperar, ocorreu uma redução exponencial da WER a medida que mais Gaussianas foram adicionadas, até a sua estabilização. Com taxa de erro de 1% no seu melhor desempenho, os dicionários baseados em regras apresentaram níveis de precisão não alcançados pelo dicionário elaborado através de técnicas de aprendizado de máquina.

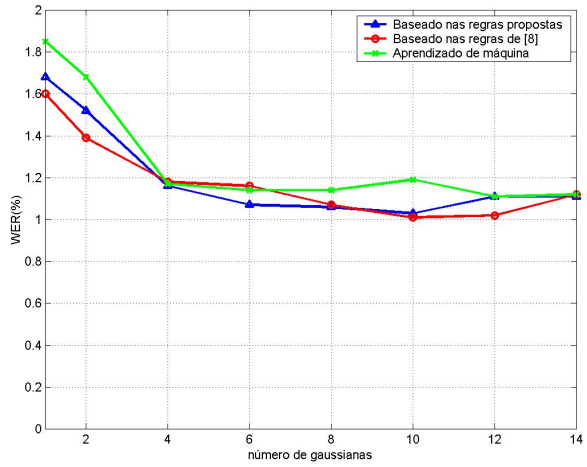

Fig. 1. Avaliação da WER(%) para um ML simplificado.

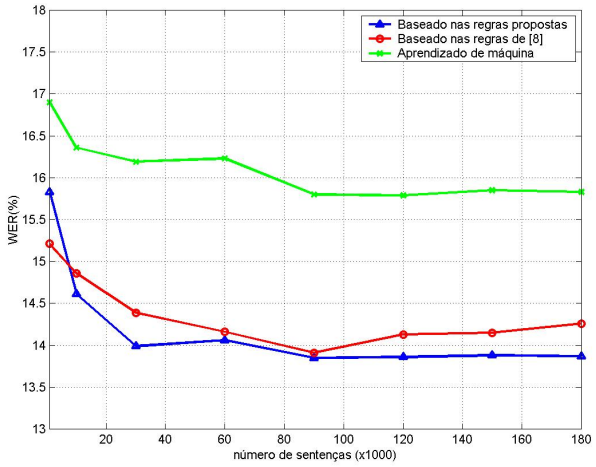

Fig. 2. Avaliação da WER para um ML independente.

Em um segundo momento, os modelos de linguagem elaborados na Seção III-C foram usados para testar o sistema, na intenção de avaliar a WER a partir de um modelo de linguagem desvinculado da base de texto usada para treinar e testar o modelo acústico. Simulações foram realizadas com os mesmos modelos acústicos criados anteriormente e com o número de Gaussianas igual a quatorze. Os resultados podem ser observados na Figura 2. Nota-se que a WER diminui à medida que o número de sentenças usadas para treinar o modelo de linguagem aumenta.

Sobre os experimentos realizados, os resultados permaneceram praticamente constantes, em alguns intervalos. A razão para isso pode estar relacionada com uma saturação do modelo de linguagem, onde quase todas as seqüências comuns de palavras já foram observadas, entretanto, as seqüências mais raras permanecem fora do âmbito do corpus de treino. Assim, a tarefa de estimar corretamente a probabilidade desses pares de palavras é bem mais complexa, já que adiciona-se uma quantidade reduzida de dados novos, sendo insuficiente para melhorar a performance do modelo de linguagem.

Novamente os melhores resultados foram obtidos com os sistemas que utilizaram um dicionário baseado em regras. Contudo, diante de um modelo de linguagem independente do contexto, ficou mais nítido que as mudanças sugeridas às regras de [8] melhoraram sensivelmente a performance do reconhecedor, estabilizando em 13.85% de WER após 90k frases de treino. É importante atentar que não existe interseção entre os corpora West Point e CETENFolha.

V. Conclusões

Este trabalho apresentou um estudo sobre o emprego de regras fonológicas com determinação de vogal tônica para reconhecimento de voz em PB. Verificou-se que tal processo não exige grandes recursos computacionais, e com poucas regras, é possível atingir taxas de acerto razoáveis. Os recursos encontram-se disponíveis ao público, permitindo a reprodução dos resultados em diferentes centros de pesquisa. É evidente que as bases de voz e texto aqui utilizadas são demasiadas pequenas para o desenvolvimento de grandes sistemas de ASR, especificamente em PB. No entanto, a estratégia é de salientar a criação de recursos necessários, mesmo que não sejam os ideais em termos de cobertura. Dessa forma a comunidade pode melhorar gradualmente aspectos como o dicionário de pronúncia, por exemplo. Os trabalhos futuros devem concentrar esforços em melhorar a taxa de palavras corretamente transcritas, principalmente na manipulação de nomes próprios e palavras estrangeiras, e avaliar o desempenho do sistema com o uso de técnicas mais refinadas, como trifones *cross-word* e modelos de linguagem *n*-gram.

References

[1] P. Barbosa, F. Violaro, E. Albano, F. Simões, P. Aquino, S. Madureira, and E. Françozo, "Aiuruetê: a high-quality concatenative text-to-speech system for brazilian portuguese with demisyllabic analysis-based units and hierarchical model of rhythm production," in *Proceedings of the Eurospeech'99, Budapest, Hungary*, 1999, pp. 2059–2062.

[2] Diamantino Caseiro, Isabel Trancoso, Luís C. Oliveira, and Maria do Céu Guerreiro Viana Ribeiro, "Grapheme-to-phone using finite-state transducers," in *IEEE Workshop on Speech Synthesis*, 2002.

[3] I. Trancoso, M. Viana, F. Silva, G. Marques, and L. Oliveira, "Rule-based vs. neural network based approaches to letter-to-phone conversion for portuguese common and proper names," in *ICSLP, Yokohama, Japan*, September 1994.

[4] E. Franzen, D. Augusto, and C.Barone, "Automatic discovery of brazilian portuguese letter to phoneme conversion rules through genetic programming," *PROPOR, pag.62-65, Faro, Portugal*, 2003.

[5] E. Brill, "Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging," *Computational Linguistics, vol.21, pp.543-566*, 1995.

[6] P. Carvalho, D. Caseiro, I. Trancoso, and L. Oliveira., "Anotação fonética automática de corpora de fala transcritos ortograficamente," *PROPOR'99 - IV Encontro para o Processamento Computacional da Língua Portuguesa Escrita e Falada*, 1999.

[7] E. C. Albano and A. A. Moreira., "Archisegment-based letter-to-phone conversion for concatenative speech synthesis in portuguese," *In Proc. of the Int. Conf. on Spoken Language Processing (ICSLP)*, 1996.

[8] D. Silva, A. de Lima, R. Maia, D. Braga, J. F. de Moraes, J. A. de Moraes, and F. Resende Jr., "A rule-based grapheme-phone converter and stress determination for brazilian portuguese natural language processing," *In VI International Telecommunications Symposium, Fortaleza, Brazil*, 2006.

[9] A. Teixeira, C. Oliveira, and L. Moutinho, "On the use of machine learning and syllable information in european portuguese grapheme-phone conversion," in *7th Workshop on Computational Processing of Written and Spoken Portuguese, Itatiaia, Brazil*, 2006.

[10] I. Seara et al, "Geração automática de variantes de léxicos do português brasileiro para sistemas de reconhecimento de fala," in *XX Simpósio Brasileiro de Telecomunicações*, 2003, pp. v.1. p.1–6.
[11] "http://www.pascal-network.org/challenges/pronalsyl/," Visited in April, 2006.
[12] "http://www.laps.ufpa.br/falabrasil," Visited in April, 2008.
[13] Chadia Hosn, Luiz Alberto Novaes Baptista, Tales Imbiriba, and Aldebaro Klautau, "New resources for brazilian portuguese: Results for grapheme-to-phoneme and phone classification," *In VI International Telecommunications Symposium, Fortaleza, Brazil*, 2006.
[14] "http://www.phon.ucl.ac.uk/home/sampa/home.htm," Visited in May, 2008.
[15] I. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*, Morgan Kaufmann, 1999.
[16] "http://www.cs.waikato.ac.nz/ml/weka/," Visited in March, 2008.
[17] "http://acdc.linguateca.pt/cetenfolha/," Visited in January, 2008.
[18] "http://www.ldc.upenn.edu.," Visited in March, 2008.
[19] S. Young, D. Ollason, V. Valtchev, and P. Woodland, *The HTK Book (for HTK Version 3.4)*, Cambridge University Engineering Department, 2006.
[20] P. Woodland and S. Young, "The htk tied-state continuous speech recognizer," *In: Proc. Eurospeech'93, Berlim*, 1993.
[21] P. Woodland, J.J. Odell, V. Valtchev, and S. Young, "Large vocabulary continuous speech recognition using htk," *IEEE Int'l Conf. Acoustics, Speech, and Signal Processing, vol. 2, pp. 125–128, Adelaide*, 1994.
[22] R. Teruszkin and F.G. Vianna, "Implementation of a large vocabulary continuous speech recognition system for brazilian portuguese," *Journal of Communication and Information Systems, vol. 21, no. 3, pages 204-218*, 2006.