Redes Neurais Convolucionais no Reconhecimento de Fala em Português para jogos em plataformas móveis

Elaynne da S. Torres[1]* Leonardo de J. Silva[1]† Rafael M. Santos [1]‡ Hendrik T. Macedo [1]§
Leonardo N. Matos [1]¶

[1]Universidade Federal de Sergipe, Departamento de Computação, Brasil

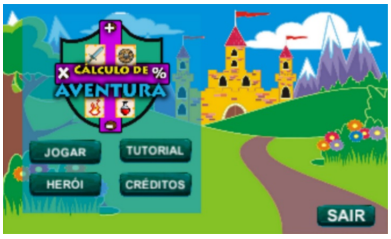

Figura 1: Cálculo de Aventura, Menu.

RESUMO

Uma das principais dificuldades no desenvolvimento de aplicações com reconhecimento de fala é a distorção de áudio causado pelo ruído ambiente. Portanto, não é comum encontrar jogos com reconhecimento automático de fala (ASR), especialmente para plataformas móveis. Este artigo descreve o desenvolvimento de dois jogos com interface 2D, que utilizam um sistema de reconhecimento online de fala. A arquitetura híbrida de reconhecimento usa uma Rede Neural Convolucional (CNN) com Modelo Escondido de Markov (HMM) para reduzir o problema do ruído ambiente. O primeiro jogo, intitulado de "Cálculo de Adventura"é um RPG no estilo de turnos e é focado em crianças de 7 a 10 anos, cujos desafios são propostas matemáticas. O outro, intitulado "Break Aracajú", é baseado no clássico Breaker Ball. Foi comparada a acurácia de reconhecimento em variados ambientes com ruídos aditivos. Resultados de experimentos com voluntários mostram que o ruído exerce pouca influência sobre o reconhecimento. O reconhecimento com o modelo CNN-HMM se mostrou factível para uso em jogos em dispositivos móveis.

Palavras-chave: Jogos para plataformas móveis, Reconhecimento Automático de Fala, Redes Neurais Convolucionais.

1 INTRODUÇÃO

A qualidade de sistemas de reconhecimento automático de fala têm evoluído sistematicamente nos últimos anos. Variados tipos de aplicações têm sido desenvolvidos com interfaces adaptadas para uso de reconhecedores de fala. Particularmente notáveis nesta linha, são aplicações voltadas para Tecnologia Assistiva, Controle de Robôs Móveis e Automação Residencial [2][3]

O uso de reconhecimento de fala em jogos não costuma ser foco de grande atenção, especialmente se considerarmos as versões para plataformas móveis. Mesmo para uso em consoles ou PCs, as opções são restritas e apresentam problemas. O jogo Binary Domain (Microsoft) possui reconhecimento de comandos em inglês. O reconhecimento de fala que se utiliza da tecnologia embarcada do sensor Kinect possui baixa qualidade de reconhecimento [1] e parece ser consideravelmente afetado por ambientes ruidosos [2]. O Kinect também é utilizado para processar comandos de voz em inglês no jogo Mass Effect 3 [3]. A solução de reconhecimento CMUSphinx é utilizada para reconhecimento contínuo em inglês no jogo de xadrez JaberChess [4].

Possivelmente, o pouco investimento no uso de reconhecimento de fala para interação com jogos reside na baixa tolerância por parte das soluções de reconhecimento mais tradicionais à ruidos do ambiente, o que forçaria o uso de headsets.Uma das soluções de reconhecimento mais tradicionais que apresenta pouca tolerância à ruídos se dá pela junção dos Modelos de Mistura Gaussiana (GMM), responsável pela representação das características do sinal da fala, com os Modelos Escondidos de Markov (HMM), responsável pela modelagem sequencial do sinal [7]. Recentemente, uma arquitetura de Aprendizado Profundo, as Redes Neurais Convolucionais (CNN) foram usadas com sucesso para substituir essa modelagem tradicional. A combinação da CNN com HMM no reconhecimento de fala (CNN-HMM) mostrou-se eficiente, mesmo em ambienteis ruidosos [8], [1]. CNNs são variações da Rede Perceptron Multicamadas (MLP) compostas de sucessivas convoluções e camadas de sub-amostragem que realizam um pré-processamento dos dados de entrada e uma camada MLP que é responsável pela saída das CNNs. A camada convolucional é composta de conjuntos de filtros chamados mapas de características responsável pela robustez das mudanças e distorções dos dados de entrada da mesma classe [6].

Considerando o potencial demonstrado por jogos digitais no processo de aprendizagem de crianças [9] e considerando a robustez recentemente demonstrada pelas arquiteturas profundas de aprendizado para reconhecimento de fala, este artigo mostra como a CNN pode ser utilizada para confecção de jogos para uso em plataformas móveis com serviço de reconhecimento online, ou seja, efetuado

*e-mail: elaynne.ufs@gmail.com
†e-mail: leonardo2013ufs@gmail.com
‡e-mail: rafaelmsse@gmail.com
§e-mail:hendrik@dcomp.ufs.br
¶e-mail: lnmatos@ufs.brr

[1]http://goo.gl/5b5m7W
[2]http://support.xbox.com/pt-BR/xbox-360/accessories/speech-recognition
[3]http://masseffect.bioware.com/about/kinect/

por um serviço remoto.

Neste artigo apresentamos o desenvolvimento e avaliação de dois diferentes jogos para plataformas móveis como forma de demonstrar a aplicabilidade do modelo CNN-HMM como ferramenta de reconhecimento de fala bem tolerante à ruidos do ambiente. O jogo denominado "Cálculo de Aventura"permite que crianças dos anos iniciais do Ensino Fundamental respondam vocalmente a desafios de matemática básica. O jogo denominado "Break Aracaju"permite o controle vocal de uma plataforma móvel cujo objetivo é rebater um objeto na direção de blocos suspensos.

2 TREINAMENTO DA REDE CNN-HMM

Este trabalho utiliza a arquitetura da CNN-HMM apresentada em [8] como serviço remoto de reconhecimento de fala para aplicações móveis. Nesta seção, descrevemos resumidamente o processo de treinamento do modelo para uso nos jogos.

Foram realizados dois treinamentos para a arquitetura de reconhecimento de palavras isoladas. O primeiro treinamento foi realizado com a base de fala composta pelos comandos “avance”, “direita”, “esquerda”, “recue” e “pare", cujo áudio foi gravado em desktops (treinamento 1). O segundo treinamento foi realizado com amostras coletadas apenas em dispositivos móveis referente aos numerais de "zero"à "nove"(treinamento 2).

Na primeira base, cada palavra foi repetida 10 vezes por 8 voluntários (6 homens e 2 mulheres), com as gravações tendo sido feitas em ambientes não controlados. Ambos os áudios foram gravados na seguinte configuração: taxa de amostragem de 8000 Hz, mono, 16 bits e arquivos wav. Cada áudio gravado das duas bases passou por um processo de anotação fonética de forma semi-automática com o plug-in EasyAlign [5] além do PRAAT, software livre para análise acústica. A anotação fonética foi utilizada em cada uma das palavras capturadas nos respectivos audios.

A segunda base teve a participação de 13 voluntários (8 mulheres e 5 homens), tendo cada palavra (de zero à nove) sido repetida 5 vezes. Os áudios foram gravados em ambientes não controlados e com interferência de ruídos de rua: som de trânsito, chuva e conversa entre pessoas.

3 DESCRIÇÃO DOS JOGOS

Os jogos foram desenvolvidos na versão gratuita do motor de jogos Unity3D, com uso da linguagem C#. O Unity3D permite processo de *build* para diferentes plataformas móveis como Android, IOS e Windows Phone. Um dos jogos baseia-se em sistema de turnos de RPG e outro requer resposta em tempo real. Esses jogos foram desenvolvidos para analisar o impacto desse tipo de reconhecedor em ambas as situações. Devido ao tempo de reconhecimento depender quase completamente da Internet, essa arquitetura é aplicável em qualquer tipo de jogo por turno (ex: RPGs como Final Fantasy, jogos de tabuleiro como xadrez, jogos educativos do tipo Quiz). Para executar comandos independentes, é possível enviar audio sequencialmente e o servidor irá fazendo reconhecimento conforme o audio chega, tornando o recebimento de palavra reconhecida mais rápido.

3.1 Cálculo de Aventura

O jogo Cálculo de Aventura é um jogo RPG single player de aventura com temática medieval e possui música ambientada. O público-alvo são crianças entre 7 a 9 anos. O jogo possui 6 fases (figura 2). Nesse jogo têm-se disponível 4 comandos de voz mapeados para as 4 opções de respostas das operações aritméticas que surgem. Apenas uma resposta é correta. Cada opção de resposta gera uma ação no personagem: “ataque com espada” ao pronunciar "avance", “defesa com escudo” ao pronunciar "recue", “usar poção de cura” ao pronunciar "direita"e “usar magia de fogo” ao pronunciar "esquerda". Cada operação matemática tem 30 segundos para ser respondida antes que nova pergunta seja lançada. O tempo estipulado foi definido levando-se em conta o tempo de 5 segundos para reconhecimento da palavra e o tempo observado que uma criança de 7 anos usualmente leva para fazer as contas usando os dedos das mãos.

Figura 2: Fases 1, 2, 3 e 4 do jogo Cálculo de Aventura.

O *game design* foi elaborado pensando numa proposta lúdica e educativa, sendo a segunda baseada no comportamento operante do psicólogo Skinner que conceitualiza o reforço, punição e extinção de comportamentos para aprendizado de uma tarefa [10]. Esse jogo requer que o jogador já tenha conhecimentos de tabuada e serve para a prática da mesma de forma interativa.

3.2 Break Aracaju

O jogo Break Aracaju é baseado em um clássico dos jogos, o Break Ball. Objetivo do jogo é não deixar o caju cair. A plataforma móvel em forma de folha deve ser movimentada através dos comandos de voz. Ela deve mover para direita, esquerda ou parar de modo a evitar a queda do caju, além de redirecionar o mesmo contra um conjunto de blocos na parte superior da tela (figura 3 à esquerda). Estes blocos de madeira devem ser destruídos para se obter maior pontuação (figura 3 à direita).

Figura 3: Tela de jogo e tela de pontuação do jogo Break Aracaju.

A tabela 1 resume os comandos vocais permitidos e seus respectivos efeitos em cada um dos dois jogos.

3.3 Arquitetura de reconhecimento

O reconhecimento de fala é um processo onde a voz é transcrita para texto com significado mais próximo ao sinal de entrada. Esse texto é aplicado para executar comandos no jogo. Sua arquitetura consiste de um *front-end* e um *back-end*, respectivamente responsáveis pela extração de características do áudio e classificação da palavra.

Na implementação dos 2 jogos, ambas as etapas são realizadas no servidor de reconhecimento. Dizemos que trata-se de um NSR (Network Speech Recognition). A vantagem de se ter o reconhecimento num servidor é que se possui maior poder de processamento e maiores vocabulários, entretanto dependerá de boa conexão e estará sujeito a atrasos na rede. O reconhecedor CNN-HMM usado nos jogos foi completamente implementado em python com a *framework* theano.

Tabela 1: Comandos vocais permitidos nos jogos.

COMANDOS	CÁLCULO DE AVENTURA	BREAK ARACAJU
Avance	executa ataque	não disponível
Recue	executa defesa	não disponível
Direita	executa magia	executa movimentação da plataforma para direita
Esquerda	executa usar cura	executa movimentação da plataforma para esquerda
Pare	não disponível	para a plataforma

Tabela 2: Acurácia para gramática do treinamento 1

Dispositivo	Sem ruído	Conversa	Chuva	Trânsito
Desktop	74,2%	59,9%	67,4%	62,9%
Mobile	73,0%	56,0%	65,0%	58,0%

Tabela 3: Acurácia para gramática do treinamento 2

Dispositivo	Sem ruído	Conversa	Chuva	Trânsito
Desktop	76,0%	70,0%	87,4%	61,8%
Mobile	81,7%	76,0%	55,0%	70,0%

Para abranger o maior número de dispositivos capazes de executar o jogo, optou-se por desenvolver um servidor que execute as duas etapas do reconhecimento online. Seria possível realizar a etapa de extração de características do áudio offline e em seguida realizar o envio desses vetores de características para o servidor. Porém, é passível de perda de qualidade do reconhecimento final e neste artigo apresentaremos os resultados do reconhecimento do áudio sem essa perda de dados.

A comunicação se dá na conexão entre cliente e servidor via socket TCP/IP. Na máquina servidora, o socket foi implementado em python 3.4 em uma máquina executando S.O. Linux. A máquina foi configurada para rodar o reconhecedor CNN-HMM com theano 0.6. Nos clientes, o socket foi implementado em C#, como o restante do jogo. O servidor aceita conexão com vários clientes criando uma nova *thread* para cada conexão e retorna a palavra reconhecida para o cliente. Cada cliente solicita conexão após gravar o áudio e fica esperando o resultado transcrito do mesmo. Se o mesmo cliente gravar outro áudio antes de receber uma resposta, este áudio é ignorado pelo servidor. A figura 4 ilustra o esquema geral do reconhecedor.

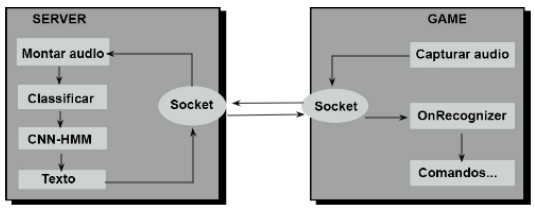

Figura 4: Esquema geral

4 EXPERIMENTOS E RESULTADOS

Com o objetivo de analizar a acurácia do reconhecedor, foram realizados experimentos com 8 voluntários. Cada voluntário utilizou seu próprio dispositivo em 4 tipos de ambientes controlados durante a execução dos jogos: um ambiente sem ruído, um ambiente com ruído de chuva, um ambiente com ruído de trânsito e um ambiente com ruído de conversa. Esses ruídos utilizados foram coletados no youtube e manteve-se em 46dB (decibeis) para chuva e transito, 39dB para ruído de conversa e 20dB em som ambiente. O pico máximo do audio nos testes com adição de ruído foi de 78dB, e 70dB em ruído ambiente.

No primeiro experimento, 8 voluntários repetiram cada um dos 5 comandos permitidos 5 vezes, para a gramática do treinamento 1 (comandos avance, recue, direita, esquerda e pare) em cada um dos 4 tipos de ambiente. No segundo experimento, 4 voluntários repetiram 5 vezes as 10 palavras da gramática do treinamento 2 (numerais de zero à nove). Ou seja, a classificação da palavra ocorreu 800 vezes na primeira base de palavras e mais 800 vezes na segunda base.

O tempo médio de reconhecimento para o cliente desktop foi de 3,28s e para o cliente em plataformas móveis foi de 5,7s. O tempo de reconhecimento é independente do ruído e é calculado baseado na soma do tempo de envio dos dados, processamento e retorno da resposta. O tempo só é computado a partir de quando se inicia a conexão com o servidor, que é apenas após os 2 segundos de captura do áudio. Pode-se observar que os dispositivos móveis gastam mais tempo de transmissão dos dados na rede, que é causado pelo tipo de arquitetura de rede wifi e 3g, que possui maior perda de dados do que uma conexão cabeada.

Para calcular o tempo de reconhecimento em PC foi feita a média de 150 execuções de comandos realizados divididos igualmente entre 3 voluntários, dois homens e uma mulher. Já o tempo de reconhecimento em dispositivos móveis foi calculado na média de 250 execuções divididas entre 5 voluntários, quatro homens e uma mulher. Cada voluntário tinha de 20 à 30 anos, exceto um voluntário de 57 anos, e repetiu as palavras de "zero"a "nove"5 vezes com ruído ambiente de uma casa.

Os dispositivos móveis usados para testes foram: Quantum Go 3G, Moto X Play 4g, Samsung Galaxy Star Trios 3g, Moto G1 3g, Tablet HP8 1401 wifi. Para teste em desktop foram usados 3 microfones: Logitech Webcam c270, microfone do headset Tritton 720+ 7.1 e microfone do headset Genius HS-G500V.

As tabelas 2 e 3 mostram resultados das taxas de reconhecimento dos clientes em *desktop* e em plataforma móvel para cada um dos experimentos descritos anteriormente.

As tabelas mostram que o reconhecimento em desktop é superior ao de plataformas móveis. A razão é possivelmente o fato de sua base de dados ter sido completamente gravada com microfones de PCs. Importante ainda ressaltar que os testes para esse treinamento requeria uma boa e pausada dicção. Com o uso de numerais como comandos, as taxas de reconhecimento subiram. Os motivos são: (i) o treinamento foi feito com um maior número de voluntários, (ii) a gramática é maior que a anterior e (iii) as palavras possuem menos fonemas. Além disso, o reconhecimento em plataformas móveis demonstrou superioridade em 3 (três) ambientes em relação ao cliente desktop. As tabelas 4 e 5 de co-ocorrência para ambientes sem ruído corroboram a precisão no reconhecimento (a primeira coluna representa o que foi dito e as linhas o que foi reconhecido).

Em virtude da dificuldade de boa qualidade do áudio na captura para situações de chuva para esses dispositivos, os áudios do desktop demonstraram melhor taxa total de reconhecimento: 73,8% contra 70,6%. Sendo o resultado do reconhecimento em dispositivos móveis na chuva bem menor que em desktop, o que rendeu taxa de reconheicmento total menor. Acredita-se que o chiado intenso da chuva se confunda com o fonema "sh"e "ss"que ocorrem nas palavras "dois", "tres", "seis", "sete"e "cinco", com esta confusão

Tabela 4: Sem ruído para gramática do treinamento 2

Ocorrência	zero	um	dois	três	quatro
zero	93,5	0	0	0	6,5
um	0	50	0	10	0
dois	0	0	100	0	0
tres	0	0	0	100	0
quatro	0	0	0	0	100
cinco	0	0	0	0	0
seis	0	0	0	73,3	0
sete	0	0	0	0	0
oito	0	0	0	0	0
nove	0	0	6,4	0	26,7

Tabela 5: Sem ruído para gramática do treinamento 2

Ocorrência	cinco	seis	sete	oito	nove
zero	0	0	0	0	0
um	10	0	0	30	0
dois	0	0	0	0	0
tres	0	0	0	0	0
quatro	0	0	0	0	0
cinco	100	0	0	0	0
seis	0	6.7	20	0	0
sete	0	0	100	0	0
oito	0	0	0	100	0
nove	0	0	0	0	66,7

tendo maior intensidade em aparelhos móveis, como é possível observar nas tabelas de co-ocorrência com ruído para a gramática do treinamento 2 (tabelas 6 e 7). Já os ruídos de trânsito e conversa não acentuam nenhum fonema em relação aos usados na gramática.

5 Conclusão

Este artigo apresentou o uso de reconhecimento automático de fala para jogos em plataforma móveis. Em particular, foi proposta e implementada uma arquitetura composta por uma rede neural convolucional (CNN) combinada com HMM para efetuar o reconhecimento online de 5 diferentes comandos vocais. Estes comandos vocais foram utilizados em dois jogos distintos implementados para estas plataformas.

Um aspecto importante no uso de reconhecimento de fala para jogos é a possível interferência do ruído ambiente na qualidade do reconhecimento; o próprio som do jogo de fato pode interferir grandemente na acurácia. A rede CNN-HMM utilizada se mostrou robusta a diferentes tipos de ruído, demonstrando que o investimento no desenvolvimento de jogos com interfaces vocais pode e deve ser estimulado, considerando que a linguagem vocal é o modo de comunicação mais natural entre humanos.

Tabela 6: Com ruído para gramática do treinamento 2

Ocorrência	zero	um	dois	tres	quatro
zero	60	0	0	0	0
um	0	63,3	3,3	6,7	0
dois	0	0	70	26,7	0
tres	0	0	3,3	76,7	0
quatro	0	0	3,3	3,3	70
cinco	0	10	30	0	0
seis	0	0	13,3	36,7	0
sete	0	0	6,7	3,3	0
oito	0	0	23,3	13,3	0
nove	0	0	0	6,7	0

Tabela 7: Com ruído para gramática do treinamento 2

Ocorrência	cinco	seis	sete	oito	nove
zero	16,7	3,3	20	0	0
um	16,7	0	3,3	6,7	0
dois	0	3,3	0	0	0
tres	3,3	16,7	0	0	0
quatro	16,7	0	3,3	0	3,3
cinco	55	0	0	5	0
seis	13,3	36,7	0	0	0
sete	0	3,3	86,7	0	0
oito	0	0	3,3	60,0	0
nove	10	3,3	10	0	70

Ao tempo que o reconhecimento online implementado nos jogos em questão apresentou boas taxas de reconhecimento, uma limitação importante de se executar em um servidor tanto a extração de caracteristicas do sinal de fala quanto o reconhecimento do comando vocal propriamente dito é a alta dependência de uma boa taxa de transferência de dados. Nos jogos em questão, o tempo de espera para reconhecimento chegou a 5 segundos no total.

Agradecimento

Agradecimentos ao CNPq pelo apoio financeiro [Universal 14/2012, Proc. 483437/2012-3] e bolsa PIBITI concedida à Leonardo Silva.

Referências

[1] O. Abdel-Hamid, A. Mohamed, H. Jiang, and G. Penn. Applying convolutional neural networks concepts to hybrid nn-hmm model for speech recognition. *2012 IEEE Int. Conf. on. IEEE*, pages 4277–4280, 1969.

[2] G. Araujo and H. T. Macedo. Context-sensitive asr for controlling the navigation of mobile robots. *21th Brazilian Symp. on Art. Intell.*, 7589:192–201, 2012.

[3] G. Araujo, H. T. Macedo, L. N. Matos, A. Oliveira, E. Silva, W. Sampaio, A. Silva, T. Mendonça, and M. Soto. Semantic validation of uttered commands in voice-activated home automation. *14th Int. Conf. on Art. Intell., 2012*, 12:643–649, 2012.

[4] J. Chen. A mobile chess app speech recognizer for visual and motion impaired users. *http://cmusphinx.sourceforge.net/page/2/*, 2016.

[5] J. Goldman. Easyalign: an automatic phonetic alignment tool under praatm. *12th Annual Conf. of the Int. Speech Commu. Association*, ser. Interspeech'11, 2011.

[6] T. Huang, J. Li, and Y. Gong. An analysis of convolutional neural networks for speech recognition. *12th Annual Conf. of the Int. Speech Commu. Association*, ISSN :1520-6149(Nº 15361822):pp. 4989 – 4993, 2015.

[7] F. Jelinek. Statistical methods for speech recognition. The MIT Press #1 Notes, 1976.

[8] R. M. Santos, L. N. Matos, H. T. Macedo, and J. Montalvao. Speech recognition in noisy environments with convolutional neural networks. *In: 2015 Brazilian Conf. on Intell. Systems (BRACIS), Natal.*, pages 175–179, 2015.

[9] E. Siqueira, D. Monteiro, D. Souza, and L. Marques. *Jogo Digital no Auxílio de Crianças com Déficit em Leitura e Escrita*, 2014.

[10] B. Skinner. Skinner, b. f. (1980). contingências do reforço: Uma análise teórica (r. moreno, trad.). *São Paulo: Abril Cultural.*, 2012.