UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

IDENTIFICAÇÃO DE IDIOMA E LOCUTOR EM SISTEMAS VOIP

LUIZ EDUARDO MARINHO GUSMÃO

ORIENTADOR: ANDERSON CLAYTON ALVES NASCIMENTO
CO-ORIENTADOR: ALEXANDRE ROMARIZ

DISSERTAÇÃO DE MESTRADO EM ENGENHARIA ELÉTRICA
ÁREA DE CONCENTRAÇÃO INFORMÁTICA FORENSE E SEGURANÇA DA INFORMAÇÃO

PUBLICAÇÃO: PPGENE.DM - 097/12

BRASÍLIA / DF: FEVEREIRO/2012

UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

IDENTIFICAÇÃO DE IDIOMA E LOCUTOR EM SISTEMAS VOIP

LUIZ EDUARDO MARINHO GUSMÃO

DISSERTAÇÃO DE MESTRADO SUBMETIDA AO DEPARTAMENTO DE ENGENHARIA ELÉTRICA DA FACULDADE DE TECNOLOGIA DA UNIVERSIDADE DE BRASÍLIA, COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE PROFISSIONAL EM INFORMÁTICA FORENSE E SEGURANÇA DA INFORMAÇÃO.

APROVADA POR:

ANDERSON CLAYTON ALVES NASCIMENTO, Doutor, UnB
(ORIENTADOR)

FLAVIO ELIAS DE DEUS, Doutor, UnB
(EXAMINADOR INTERNO)

GEORGES NZE, Doutor, UnB
(EXAMINADOR EXTERNO)

DATA: BRASÍLIA/DF, 27 DE FEVEREIRO DE 2012.

FICHA CATALOGRÁFICA

GUSMÃO, LUIZ EDUARDO MARINHO
Identificação de Idioma e Locutor em Sistemas VoIP [Distrito Federal] 2012.
xxii, 54 p., 297 mm (ENE/FT/UnB, Mestre, Engenharia Elétrica, 2012).

Dissertação de Mestrado – Universidade de Brasília, Faculdade de Tecnologia. Departamento de Engenharia Elétrica.

1. Sistemas VoIP 2. Identificação de idioma
3. Identificação de locutor 4. Redes neurais

I. ENE/FT/UnB. II. Título (Série)

REFERÊNCIA BIBLIOGRÁFICA

GUSMÃO, L. E. M. (2012). Identificação de Idioma e Locutor em Sistemas VoIP. Dissertação de Mestrado, Publicação PPGENE.DM - 097/12, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF, 54p.

CESSÃO DE DIREITOS

NOME DO AUTOR: Luiz Eduardo Marinho Gusmão
TÍTULO DA DISSERTAÇÃO: Identificação de Idioma e Locutores em Sistemas VoIP.
GRAU/ANO: Mestre/2012.

Luiz Eduardo Marinho Gusmão
Universidade de Brasília
Campus Universitário Darcy Ribeiro – Asa Norte
CEP 70910-900
Brasília/DF - Brasil

À minha amada esposa, Isabel, e aos meus queridos pais, Luiz e Hilma.

AGRADECIMENTOS

Ao Prof. Dr. Anderson Nascimento, meu orientador, por sua constante disponibilidade e paciência para prestar auxílio sempre que requisitado.

Ao Prof. Dr. Flavio Elias Gomes de Deus, em nome do qual agradeço aos demais professores do curso, por sua incansável doação e pelo conhecimento compartilhado.

Ao Perito Criminal Federal e Prof. Dr. Hélvio Peixoto por sua enorme dedicação, sem a qual, este curso não teria acontecido.

Aos Peritos Criminais Federais e Mestres André Morum e Levi Roberto Costa, por terem dividido as alegrias e ajudado a superar as tristezas no decorrer do curso.

O presente trabalho foi realizado com o apoio do Departamento Polícia Federal – DPF, com recursos do Programa Nacional de Segurança Pública com Cidadania – PRONASCI, do Ministério da Justiça.

RESUMO

IDENTIFICAÇÃO DE IDIOMA E LOCUTOR EM SISTEMAS VOIP

Autor: Luiz Eduardo Marinho Gusmão
Orientador: Anderson Clayton Alves Nascimento
Co-orientador: Alexandre Romariz
Programa de Pós-graduação em Engenharia Elétrica
Brasília, fevereiro de 2012

O presente trabalho explora o uso de redes neurais artificiais na identificação de informações sobre o conteúdo de transmissões de voz sobre IP criptografas. O estudo aborda especificamente duas informações: o idioma da chamada e o seu locutor. Para demonstrar na prática os métodos empregados, foram realizados experimentos em laboratório, nos quais o tráfego de rede gerado pelo *software* Skype foi capturado e analisado.

ABSTRACT

LANGUAGE AND SPEAKER DETECTION IN VOIP SYSTEMS

Author: Luiz Eduardo Marinho Gusmão
Supervisor: Anderson Clayton Alves Nascimento
Co-supervisor: Alexandre Romariz
Programa de Pós-graduação em Engenharia Elétrica
Brasília, february of 2012

The current academic work explores the use of artificial neural networks on the content's identification of encrypted voice over IP systems. This study specifically approaches two kinds of information: the language and the caller's identity. Laboratory experiments were developed to demonstrate the employed methods, in which Skype's network traffic was collected and analyzed.

SUMÁRIO

1. INTRODUÇÃO 1

1.1. TRABALHOS RELACIONADOS 2

1.2. OBJETIVO 4

1.3. ORGANIZAÇÃO DO TRABALHO 5

2. SISTEMAS VOIP 6

2.1. FUNCIONAMENTO 7

2.2. PROTOCOLO DE SINALIZAÇÃO 8

2.3. PROTOCOLO DE TRANSPORTE 8

2.4. CODECS VOIP 9

3. SKYPE 13

3.1. FUNCIONAMENTO 13

3.2. SILK 16

3.2.1. Taxa de Amostragem 16

3.2.2. Taxa de Criação de Pacotes 17

3.2.3. Taxa de *Bits* 17

3.2.4. Taxa de Perda de Pacotes 18

3.2.5. FEC (*Foward Error Correction*) 18

3.2.6. Complexidade 18

3.2.7. DTX (*Discontinuous Transmission*) 18

3.3. SEGURANÇA 19

4. REDES NEURAIS ARTIFICIAIS 20

4.1. ARQUITETURA 23

4.1.1. Progressiva de camada única (*Single-Layer Feedforward*) 23

4.1.2. Progressiva multicamadas (*Multilayer Feedforward*) 24

4.1.3. Recorrentes (*Feedback*) 25

4.2. TREINAMENTO 25

4.2.1. Treinamento Supervisionado 26

4.2.2. Treinamento Não Supervisionado 27

5. EXPERIMENTO 28
5.1. PREPARAÇÃO DO CORPUS 28
5.2. COLETA DO TRÁFEGO DE REDE 29
5.3. ANÁLISE E PREPARAÇÃO DOS DADOS 32
5.3.1. Identificação de Idioma 34
5.3.2. Identificação de Locutor 35
5.4. TREINAMENTO DA REDE NEURAL ARTIFICIAL 36
5.4.1. Identificação de Idioma 38
5.4.2. Identificação de Locutor 43
5.5. RESULTADOS 48

6. CONCLUSÃO 51
6.1. LIMITAÇÕES 51
6.2. TRABALHOS FUTUROS 52

REFERÊNCIAS BIBLIOGRÁFICAS 53

LISTA DE TABELAS

Tabela 2.1 - Comparação entre tarifas de operadora convencional e VoIP 6
Tabela 2.2 - Taxas de amostragens utilizadas por tecnologias diversas 10
Tabela 3.1 - Sistemas operacionais que possuem versões do Skype 13
Tabela 3.2 - Frequência máxima de amostragem utilizada por cada modo de operação 16
Tabela 3.3 - Taxa de criação de pacotes 17
Tabela 3.4 - Taxa de *bits* para cada modo de operação 18
Tabela 3.5 - Localização dos arquivos contendo informações sobre a utilização do Skype 19
Tabela 5.1 - Estatísticas sobre os pacotes capturados durante a identificação do idioma 34
Tabela 5.2 - Estatísticas sobre os pacotes capturados durante a identificação do locutor 35
Tabela 5.3 - Algoritmo SCG 39

LISTA DE FIGURAS

Figura 2.1 - Funcionamento de um sistema VoIP 7
Figura 2.2 - Exemplos de taxas de amostragem de um sinal analógico 9
Figura 2.3 - Comparação entre sons codificados através de VBR e CBR 11
Figura 2.4 - Arquivo de áudio contendo períodos de silêncio no início e no fim 12
Figura 2.5 - Redução do quantidade de *bytes* transmitidos 12
Figura 3.1 - Componentes da arquitetura do Skype 15
Figura 4.1 - Representação do neurônio humano (Buckland, 2002) 20
Figura 4.2 - Representação do neurônio artificial 21
Figura 4.3 - Representação do *perceptron* 21
Figura 4.4 - Função degrau 22
Figura 4.5 - Função linear 23
Figura 4.6 - Função sigmoide 23
Figura 4.7 - RNA progressiva de camada única 24
Figura 4.8 - RNA progressiva multicamadas 24
Figura 4.9 - RNA recorrente 25
Figura 5.1 - Fases dos experimentos 28
Figura 5.2 - Topologia do laboratório utilizado durante os experimentos 29
Figura 5.3 - Etapas da chamada 30
Figura 5.4 - Janela contendo as informações técnicas sobre a chamada 30
Figura 5.5 - Conexão ponto a ponto entre os comutadores 31
Figura 5.6 - Janela de configuração do Skype para Windows 32
Figura 5.7 - Janela de configuração do Skype para Mac OS X 32
Figura 5.8 - Gravação original 33
Figura 5.9 - Tráfego de rede (Mac OS X) 33
Figura 5.10 - Tráfego de rede (Windows) 33
Figura 5.11 - Distribuição dos tamanhos dos pacotes por idioma 34
Figura 5.12 - Matriz de confusão 36
Figura 5.13 - Arquitetura da RNA utilizada para a identificação de idioma 38
Figura 5.14 - Matriz de confusão do treinamento da RNA utilizada na identificação do idioma 40
Figura 5.15 - Matriz de confusão da validação da RNA utilizada na identificação do idioma 41

Figura 5.16 - Matriz de confusão do teste da RNA utilizada na identificação do idioma 41
Figura 5.17 - Matriz de confusão consolidada das subfases da RNA utilizada na identificação do idioma 42
Figura 5.18 - Performance do treinamento da RNA utilizada na identificação do idioma 42
Figura 5.19 - Curva ROC do treinamento da RNA utilizada na identificação do idioma 43
Figura 5.20 - Arquitetura da rede utilizada para a identificação do locutor 44
Figura 5.21 - Matriz de confusão do treinamento da RNA utilizada na identificação do locutor 45
Figura 5.22 - Matriz de confusão da validação da RNA utilizada na identificação do locutor 45
Figura 5.23 - Matriz de confusão do teste da RNA utilizada na identificação do locutor 46
Figura 5.24 - Matriz de confusão consolidada das subfases da RNA utilizada na identificação do locutor 46
Figura 5.25 - Performance do treinamento da RNA utilizada na identificação do locutor 47
Figura 5.26 - Curva ROC do treinamento da RNA utilizada na identificação do locutor 47
Figura 5.27 - Matriz de confusão do resultado da detecção de idioma 48
Figura 5.28 - Matriz de confusão do resultado da detecção de locutor 49
Figura 5.29 - Matriz de confusão do resultado da detecção de locutor (5:5) 49
Figura 5.30 - Matrix de confusão do resultado da detecção de locutor (10:0) 50

LISTA DE SÍMBOLOS, NOMENCLATURA E ABREVIAÇÕES

AES – Advanced Encryption Standard

BPS – Bits por segundo

CBR – Constant Bit Rate

CODEC – Codificador/Decodificador

CPAN – Comprehensive Perl Archive Network

DTW – Dinamic Time Warping

DTX – Discontinuous Transmission

FEC – Foward Error Correction

IAX – Inter-Asterisk Exchange Protocol

IETF – Internet Engineering Task Force

IP – Internet Protocol

IRMA – Information Resources Management Association

ITU-T – International Telecommunication Union - Telecommunication Standardization Sector

KBPS – Kilobits por segundo

MCU – Multipoint Control Unit

MSE – Mean-Squared Error

NAT – Network Address Translation

P2P – Peer-to-peer

PBX – Private Branch Exchange

PSP – PlayStation Portable

RFC – Request for Comments

RNA – Redes Neurais Artificiais

ROC – Receiver Operating Characteristic

RSA – Ron Rivest, Adi Shamir and Leonard Adleman

RTCP – Real-time Transport Control Protocol

RTP – Real-time Transport Protocol

SCG – Scaled Conjugate Gradient

SIP – Session Initiation Protocol

SRTP – Secure Real-time Transport Protocol

TCP – Transmission Control Protocol

UDP – User Datagram Protocol

VAD – Voice Activity Detection

VBR – Variable Bit Rate

VoIP – Voice Over Internet Protocol

XML – Extensible Markup Language

1. INTRODUÇÃO

A interceptação telefônica é um procedimento comum e bem consolidado no Brasil, sendo bastante utilizado pela polícia judiciária para a obtenção de provas ou indícios do cometimento de crimes. As técnicas para a realização deste tipo de interceptação são amplamente conhecidas pelos órgãos policiais e já existem soluções que automatizam todo o processo, como o sistema Guardião[1].

Sabendo disso, os criminosos vêm buscando alternativas para a manutenção do sigilo de suas comunicações, como, por exemplo, o uso de mensagens de correio eletrônico e de mensagens instantâneas, através de *softwares* como o MSN Live Messenger[2]. Essa mudança no comportamento das organizações criminosas resultou no emprego de um novo método de investigação, conhecido como interceptação telemática.

Apesar de existir previsão legal para esse procedimento investigativo na legislação brasileira, sua utilização na prática não é tão usual quanto sua contrapartida telefônica, pois as técnicas para o seu emprego ainda não estão padronizadas e amplamente divulgadas no Brasil.

Uma outra opção que pode ser utilizada pelos criminosos é o uso dos sistemas de voz sobre IP (*Voice Over Internet Protocol* - VoIP), como o Skype[3] e o Google Voice[4]. Além das chamadas de voz entre terminais que utilizam os referidos sistemas, estes oferecem ainda uma ampla gama de recursos extras, como chamadas em conferência, chamadas em vídeo e até a realização ligações para telefones fixos e celulares.

Adicionalmente à quantidade de recursos e à facilidade de utilização, outro fator para a popularização desses sistemas é o custo de sua utilização, uma vez que muitos dos serviços oferecidos não são tarifados.

Entretanto, a característica dos sistemas VoIP que mais interessa aos órgãos de segurança pública vem a ser a privacidade das chamadas, obtida com a utilização de criptografia. Logo, o novo desafio dos referidos órgãos é encontrar mecanismos de interceptação telemática que possam lidar com informações cifradas. Os sistemas VoIP geralmente utilizam algoritmos criptográficos robustos e reconhecidos pelo mercado, como o AES, usado pelo Skype.

[1] http://www.escoladaajuris.org.br/cam/2011/abril/Nedson/Apresentacao_Ajuris.pdf
[2] http://explore.live.com/messenger
[3] http://www.skype.com/
[4] http://www.google.com/voice

O *Advanced Encryption Standard (AES)*, é o algoritmo padrão utilizado pelo governo americano e consiste em uma cifra de bloco simétrica, que utiliza blocos de 128 *bits* e chaves de 128, 192 ou 256 *bits*. Considerando a utilização de uma chave de tamanho de 128 *bits*, e a realização de uma decifração por microssegundo, o tempo necessário para quebrar esta chave por meio de força bruta é de $5,4 \times 10^{24}$ anos (Stallings, 2010).

Portanto, a tarefa de decifrar uma comunicação VoIP cifrada por meio de técnicas de força bruta mostra-se uma tarefa impraticável e a busca de técnicas alternativas é uma questão crucial. Um dos métodos propostos é o ataque conhecido como *side-channel*, que tira proveito do "vazamento" de certas informações sobre uma transmissão, como o tamanho e o intervalo de tempo dos pacotes gerados.

Este trabalho propõe, então, a utilização deste tipo de ataque para a obtenção de informações sobre o conteúdo de chamadas VoIP, visando detectar o idioma utilizado na conversação e o seu locutor. O *software* escolhido para ser submetido a esta investida foi o Skype, devido à sua grande popularidade.

Para atingir o objetivo proposto, foi necessário capturar o tráfego de rede de várias chamadas, nas quais foram transmitidos os áudios de gravações realizadas por diversos locutores diferentes e em 4 idiomas distintos.

A classificação das informações obtidas nas categorias pretendidas foi realizada com uso de redes neurais artificiais (RNA), devido sua habilidade em resolver problemas complexos, como a identificação de padrões.

A abordagem apresentada resultou na classificação correta do idioma português, dentre outros três, em 80,6% dos casos. Também resultou na identificação de um locutor previamente monitorado com uma taxa de acerto de até 90% dos casos.

De acordo com o conhecimento do autor, este é o primeiro trabalho que aborda a detecção de idiomas e locutores em chamadas criptografadas realizadas a partir do *software* Skype e do *codec* SILK (Vos; Jensen; Soerensen, 2010), e com a utilização de redes neurais artificiais como classificador.

1.1. TRABALHOS RELACIONADOS

Além da facilidade e do baixo custo de utilização, outro fator decisivo para a popularização dos sistemas VoIP é a segurança. Atualmente, como a grande maioria dos aplicativos realiza o

ciframento da comunicação, descobrir o conteúdo das mesmas utilizando técnicas como a força-bruta é uma tarefa impraticável, conforme demonstrado na seção anterior.

Entretanto, de acordo com diversos estudos previamente publicados, é possível utilizar as informações obtidas a partir da captura do tráfego de rede, como o tamanho dos pacotes e o intervalo de tempo entre os mesmos, na tarefa de identificação do conteúdo de uma chamada. Dentre as informações passíveis de identificação, podem ser citadas o idioma, o locutor e as frases pronunciadas durante uma chamada.

T. Lella e R. Bettati (2007) demonstraram os efeitos do recurso de supressão de silêncio na detecção do conteúdo de chamadas realizadas por sistemas VoIP. Mediante a análise de três sistemas VoIP diferentes (Google Talk, Skype e Speaker Freely), observaram que, durante a transmissão de voz, os *softwares* produzem pacotes de tamanhos e em intervalos de tempo distintos dos produzidos durante períodos de silêncio. Utilizaram uma rima infantil e calcularam o intervalo entre os pacotes gerados por cada palavra. Por meio da análise deste intervalo, puderam determinar a sequência com que palavras foram articuladas durante um discurso capturado. Utilizaram classificação Bayesiana e cadeias ocultas de Markov como classificadores.

O ataque proposto por Lu (2007) também explora o recurso de supressão de silêncio e o tamanho dos pacotes gerados durante a transmissão para obter informações sobre seu conteúdo. A quantidade de *bytes* transmitidos em um certo período de tempo foi utilizada no treinamento de um modelo oculto de Markov, com o objetivo de identificar se parte de uma transmissão previamente monitorada estava sendo reproduzida ou se os locutores envolvidos na transmissão estavam se comunicando novamente. Os resultados obtidos com a utilização do Skype atingiram 33% e 44% de acertos na identificação do discurso e locutor, respectivamente.

O objetivo de Charles V. Wright et al. (2007) foi a identificação do idioma utilizado em uma chamada cifrada, por meio da utilização de uma variação da distribuição χ^2. Os pesquisadores criaram modelos para 21 idiomas, utilizando os tamanhos dos pacotes gerados pelo *software* VoIP Linphone e codificação utilizando taxa de *bits* variável. O resultado alcançado foi uma taxa de acerto de 66%. Porém, ao reduzir a quantidade de idiomas para 14, a taxa de acerto foi superior a 90%

No ano seguinte, Charles V. Wright et al. (2008) demonstraram que é possível identificar frases pronunciadas em uma chamada criptografada, a partir da suposição de que uma palavra

sempre resulta em tamanhos de pacotes ordenados da mesma forma. Foram criados modelos para as palavras desejadas, por meio da decomposição das mesmas em fonemas. Estes, então, foram correlacionados às quantidades de *bytes* de cada pacote produzido a partir de uma chamada. Utilizou-se o modelo oculto de Markov e a precisão média atingida foi de 51%, passando de 90%, em certos casos. Este estudo utilizou o *codec* chamado Speex e a codificação do áudio foi realizada com taxa de *bits* variável, uma vez que neste tipo de codificação os pacotes são produzidos com tamanhos diferentes para cada tipo de som.

Finalmente, na pesquisa elaborada por Benoît Dupasquier et al. (2010) foi proposto um método para a identificação de sentenças pronunciadas durante uma conversação realizada via Skype. Durante seus experimentos, os autores demonstraram que, considerando o tamanho dos pacotes gerados durante a transmissão, um determinado discurso sempre produzirá os mesmos resultados. Seu método analisava os tamanhos dos pacotes resultantes da transmissão de fonemas contidos em frases como "*I put the bomb in the plane*". Em seguida, com a utilização de sintetizadores de voz, foram gerados modelos para cada fonema isoladamente, de modo que foi possível identificar a ocorrência dos mesmos em frases transmitidas de forma cifrada, com uma taxa de acerto de até 83%. Os testes foram realizados no idioma inglês e adotaram o algoritmo DTW (*Dynamic Time Warping*), que foi utilizado como classificador das sentenças.

1.2. OBJETIVO

O principal objetivo deste trabalho foi demonstrar que é possível identificar informações sobre o conteúdo de uma chamada realizada através do Skype, com a utilização do *codec* SILK, apesar do ciframento empregado na transmissão e do desconhecimento prévio de qualquer informação sobre o áudio original.

Foi apresentado, ainda, um método para que as situações supracitadas pudessem ser classificadas, independentemente da quantidade de pacotes produzidos durante uma chamada, aproximando o experimento de uma situação real, na qual a duração da chamada é indeterminada.

O trabalho também se propôs a mostrar o funcionamento dos sistemas VoIP, especialmente do Skype, com sua arquitetura ponto-a-ponto, e do *codec* atualmente utilizado pelo mesmo, o SILK.

Além disso, o trabalho apresentou alguns conceitos referentes às redes neurais artificiais e sua utilização na classificação das informações desejadas.

Finalmente, o trabalho também tinha como objetivo a realização de experimentos que pudessem demonstrar, na prática, como é possível a detecção das informações desejadas de em uma chamada VoIP. Para tal, diversos algoritmos e configurações da RNA foram testados e os que tiveram o melhor desempenho foram apresentados, bem como os resultados dos experimentos.

1.3. ORGANIZAÇÃO DO TRABALHO

Os próximos capítulos do presente trabalho foram divididos da seguinte maneira:

- No capítulo 2 são descritas os principais conceitos e características dos sistemas VoIP, incluindo o funcionamento de um de seus principais componentes, o *codec*;
- No capítulo 3 o Skype é apresentado, já que é o mais popular dos aplicativos de voz sobre IP e que, por este motivo, foi o *software* escolhido para a realização dos experimentos;
- No capítulo 4 os conceitos gerais sobre as redes neurais artificiais são descritos;
- No capítulo 5 são apresentados os métodos utilizados durante a realização dos experimentos e os resultados alcançados pelos mesmos; e
- O capítulo 6, por fim, traz as conclusões deste trabalho, incluindo suas limitações e as sugestões para futuros trabalhos.

2. SISTEMAS VOIP

Os sistemas de voz sobre IP (VoIP) são um conjunto de tecnologias que tem como objetivo primário oferecer o serviço de transmissão de sinais voz por meio de redes de comunicação que utilizem o protocolo IP. Apesar de ser seu objetivo principal, os sistemas VoIP atuais não se limitam à comunicação de voz e oferecem outros tipos de serviços, como o envio de mensagens de texto, chamadas em vídeo e até a troca de arquivos.

Estes sistemas se tornaram populares entre os usuários domésticos a partir da popularização dos serviços de banda larga, que proporcionaram uma melhora na qualidade do áudio, chegando a níveis próximos ao da telefonia convencional. Segundo Simionovich (2008), um marco importante na história desta tecnologia foi o lançamento, em 1995, do *software* batizado de Internet Phone, desenvolvido pela empresa israelense Vocaltec. Este foi o primeiro aplicativo a oferecer comunicação de voz entre usuários utilizando uma rede IP.

Desde então, os sistemas VoIP evoluíram bastante e não ficaram restritos apenas ao ambiente doméstico, popularizando-se também entre os usuários corporativos. No mercado atual de VoIP existem soluções implementadas, tanto via *software*, que são os chamados *softphones*, quanto através de *hardware*. Considerando as diversas soluções existentes, o Skype, o Google Voice, o Vonage[5] e o Cisco IP Communicator[6] merecem ser destacadas.

Dentre as principais vantagens deste tipo de telefonia estão a implementação nativa de serviços inteligentes, como a identificação de chamadas, o ciframento da comunicação e, principalmente, o custo reduzido de sua utilização. Vale destacar que em muitos casos, as chamadas não chegam nem a ser tarifadas. Consequentemente, os sistemas VoIP se tornaram fortes concorrentes das companhias telefônicas. Um comparativo entre os custos de chamadas de longa distância realizadas para o estado do Rio de Janeiro por uma companhia de telefonia fixa e pelo Skype podem ser vistos na tabela 2.1:

Tabela 2.1 - Comparação entre tarifas de operadora convencional e VoIP

Operadora	Telefone Fixo	Telefone Móvel
Oi	0,46	1,72
Skype	0,10	0,62

[5] http://www.vonage.com/
[6] http://www.cisco.com/en/US/products/sw/voicesw/ps5475/index.html

2.1. FUNCIONAMENTO

De uma forma geral, o funcionamento de um sistema de voz sobre IP consiste nas etapas de localização dos usuários, de gerenciamento da chamada, também denominada sessão, na conversão do sinal analógico de voz em digital e, finalmente, na transmissão do sinal convertido.

Na primeira etapa, o sistema deve empregar técnicas que permitam a localização dos clientes do serviço, independentemente da forma na qual estejam conectados.

Na próxima etapa, o sistema deve fornecer aos usuários os meios necessários para que uma chamada VoIP, ou sessão, possa ser estabelecida e encerrada. Adicionalmente, também pode oferecer outros recursos, como o estabelecimento de uma conferência, com vários interlocutores.

Na etapa seguinte, o sistema deve aplicar técnicas visando a conversão do sinal de voz analógico em digital, de modo que este possa ser transmitido pela rede. Estas técnicas também servem para atingir o equilíbrio ideal entre a qualidade do áudio e a largura de banda utilizada.

Finalmente, a última etapa consiste no transporte do sinal convertido do locutor até o usuário ouvinte, independentemente da maneira como estes se conectam. A figura 2.1 ilustra o funcionamento básico de um sistema VoIP:

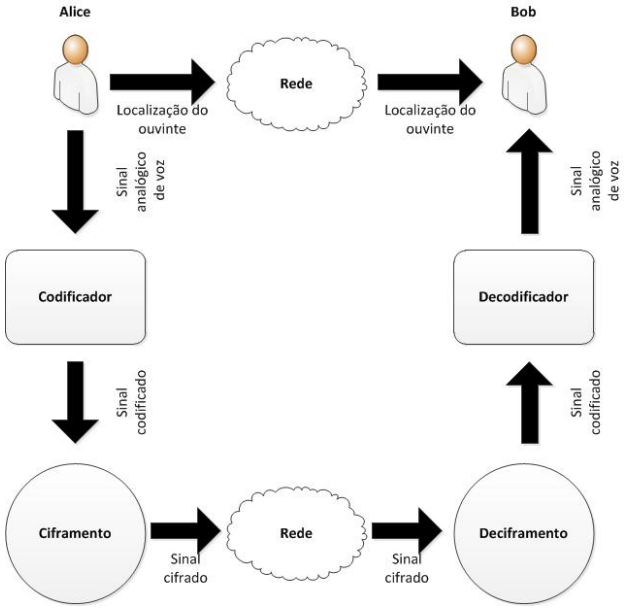

Figura 2.1 - Funcionamento de um sistema VoIP

2.2. PROTOCOLO DE SINALIZAÇÃO

Os objetivos principais do protocolo de sinalização são a localização do destinatário de uma chamada e o gerenciamento da mesma, incluindo o seu estabelecimento e o seu encerramento. O SIP, o H.323 e o IAX podem ser citados como protocolos desta categoria (T. Abbasi et al., 2005).

O primeiro, o SIP (*Session Initiation Protocol* ou, em português, Protocolo de Inicialização de Sessão), foi definido pela RFC 3261 (J. Rosenberg et al., 2002), do IETF (*Internet Engineering Task Force*), e é o protocolo de sinalização mais utilizado em sistemas VoIP (IRMA, 2010). Apesar disso, não foi desenvolvido especificamente para este fim, podendo ser utilizado em outras aplicações multimídia.

O H.323, por sua vez, não corresponde a um único protocolo de comunicação, mas sim a uma recomendação do ITU-T (*International Telecommunication Union - Telecommunication Standardization Sector*) para a transmissão de conteúdo multimídia através das mais diferentes topologias de rede (ITU-T, 2009). Sua arquitetura prevê a utilização de vários componentes de rede, como terminais, unidades de controle multiponto (MCU) e *gateways*, o que torna sua implementação mais complexa que as demais.

Finalmente, o IAX (*Inter-Asterisk Exchange Protocol*) é um protocolo de sinalização produzido pela empresa estadunidense Digium[7], para ser utilizado por seu sistema Asterisk (Simionovich, 2008), um popular sistema de PBX (*Private Branch Exchange*) de código aberto e que implementa comunicações por meio de redes IP. Apesar de não ser um padrão do IETF, seu funcionamento é definido através da RFC 5456 (M. Spencer et al., 2010).

Diferentemente do SIP e do H.323, o IAX foi desenvolvido especificamente para o uso em sistemas VoIP, característica que o torna mais econômico em termos de consumo de banda da rede.

2.3. PROTOCOLO DE TRANSPORTE

O protocolo de transporte é o responsável pela entrega do conteúdo da conversação gerada pelo sistema VoIP ao destinatário, sendo que o principal protocolo utilizado para este fim é o RTP (*Real-time Transport Protocol*, ou Protocolo de Transporte em Tempo Real, em português). Foi publicado em 1996 pelo IETF e é atualmente definido pela RFC 3550 (H.

[7] http://www.digium.com/

Schulzrinne et al., 2003), além de fazer parte das recomendações da especificação H.323, do ITU-T. Seu objetivo é proporcionar a transmissão em tempo real de conteúdo multimídia por meio de redes IP, podendo atuar tanto sobre o protocolo TCP (*Transmission Control Protocol*) quanto sobre o UDP (*User Datagram Protocol*), apesar deste ser o mais comum.

O RTP possui um protocolo auxiliar, chamado RTCP (*RTP Control Protocol*), que atua na execução das tarefas de monitoração e controle da qualidade de serviço da transmissão.

Outro protocolo que trabalha em conjunto com o RTP, e foi concebido com o objetivo de proporcionar segurança ao mesmo (Perkins, 2003), chama-se SRTP (*Secure RTP*). Definido pela RFC 3711 (M. Baugheret et al., 2004), o SRTP utiliza o algoritmo AES para realizar o ciframento da informação que será transmitida.

2.4. CODECS VOIP

Além dos protocolos de comunicação, os sistemas VoIP utilizam outro tipo de *software*, responsável pela transformação do sinal analógico de voz em conteúdo digital (*bits*). O responsável por esta tarefa é chamado de *codec*, abreviação em inglês para codificador e decodificador.

A conversão é realizada mediante a coleta de amostras de um sinal analógico ao longo do tempo, resultando em uma sequência de *bits*, ou *bitstream*. A frequência com que as amostras são coletadas é denominada de taxa de amostragem. A quantidade de amostras em um segundo é medida em hertz (Hz). O gráfico demonstrando a taxa de amostragem de um sinal analógico pode ser visto na figura 2.2.

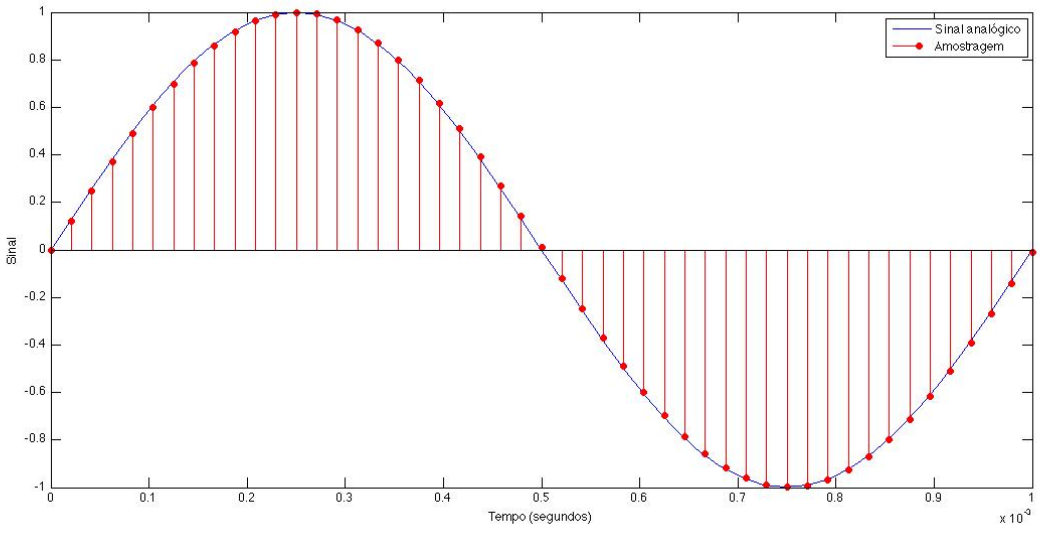

Figura 2.2 - Exemplos de taxas de amostragem de um sinal analógico

A taxa de amostragem depende da qualidade de áudio pretendido. Exemplos de taxas de amostragem podem ser vistas na tabela 2.2.

Tabela 2.2 - Taxas de amostragens utilizadas por tecnologias diversas

Exemplo	Taxa de Amostragem
Chamada telefônica	8000 Hz
Rádio AM	22050 Hz
Rádio FM	32000 Hz
CD de música	44100 Hz

Outro parâmetro que deve ser considerado pelo *codec* durante o processo de codificação é a taxa de *bits* (*bitrate*) com a qual o áudio será convertido. Medido em *kilobits* por segundo (kbps), este valor geralmente é escolhido de acordo com o meio no qual a transmissão será realizada, influenciando diretamente na qualidade do áudio. A taxa de *bits* pode ser classificada entre variável e constante.

No primeiro tipo, conhecido como CBR (*Constant Bit Rate*), a quantidade de bits necessários para a codificação de um intervalo de áudio é sempre constante, independentemente de sua complexidade. É indicado para aplicações nas quais a taxa de transmissão necessite ser constante, como no caso de *streaming*.

Diferentemente do primeiro tipo, no VBR (*Variable Bit Rate*), a quantidade de *bits* necessários para codificar um intervalo de áudio é diretamente proporcional à complexidade dos sons reproduzidos (B. Fries; M. Fries, 2005). Se por um lado ocorre, então, uma economia de banda, pois os sons mais simples necessitam de uma quantidade menor de *bits*, por outro lado, esta característica torna os *codecs* deste tipo vulneráveis a ataques que buscam identificar informações sobre o áudio transmitido baseando-se no tráfego de rede. A figura 2.3 mostra a comparação dos resultados da codificação de um mesmo intervalo de áudio utilizando *codecs* CBR e VBR.

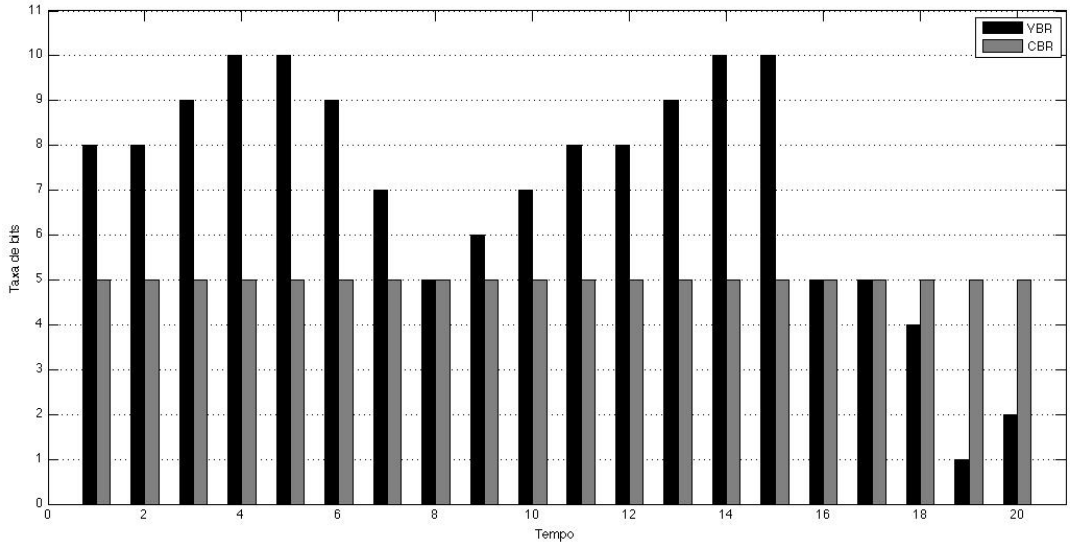

Figura 2.3 - Comparação entre sons codificados através de VBR e CBR

Visando a economia da banda utilizada durante uma transmissão, o *codec* também realiza a compressão do sinal de voz. Existem dois tipos básicos de compressão de áudio. O primeiro tipo é a compressão com perda de dados, chamada em inglês de *lossless*. Como o próprio nome deixa a entender, neste tipo de compressão, as informações não são perdidas, pois o *codec* apenas substitui os sons mais comuns e as sequências repetidas por um mesmo valor.

O outro tipo é chamado de compressão com perda de dados, ou *lossy*, em inglês, e consistem em eliminar os sons não essenciais para o entendimento da conversa, como, por exemplo, os sons inaudíveis ao ouvido humano. Assim sendo, por haver perda de informação durante a codificação, o *codec* que utiliza este tipo de compressão tende a possuir um resultado melhor que o primeiro tipo.

Além da compressão, outra técnica que pode ser utilizada pelo *codec* visando preservar a largura de banda do meio é chamada de supressão do silêncio. Este recurso, também é conhecido como VAD (*Voice Activity Detection* ou, em português, detecção da atividade de voz), e permite ao *codec* reduzir o envio de pacotes ao detectar períodos de silêncio durante uma chamada. Para evitar que o interlocutor suspeite que a chamada foi encerrada por conta do silêncio, é gerado um ruído chamado de *comfort noise* (J. Alexander et al., 2005). As figuras 2.4 e 2.5 representam, respectivamente, o gráfico de um arquivo de áudio e a quantidade de *bytes* gerados pela transmissão do mesmo.

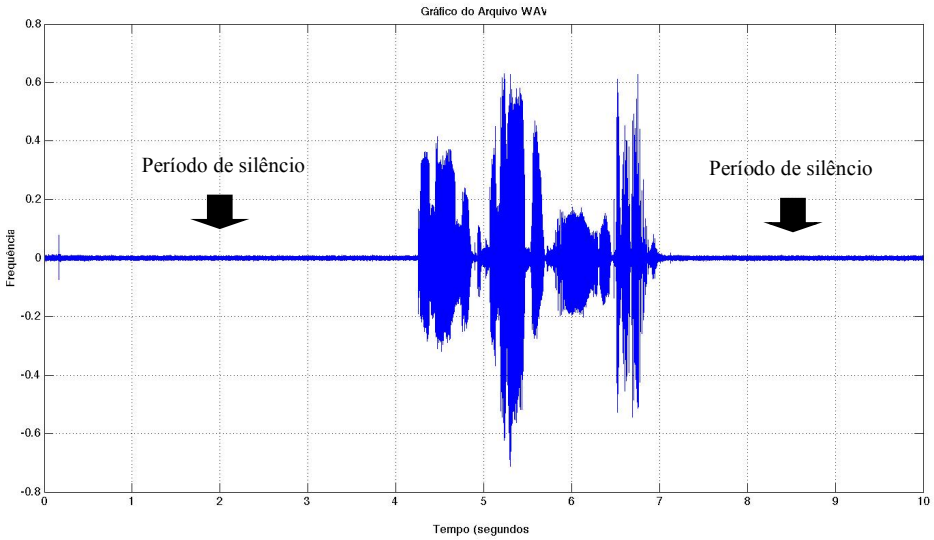

Figura 2.4 - Arquivo de áudio contendo períodos de silêncio no início e no fim

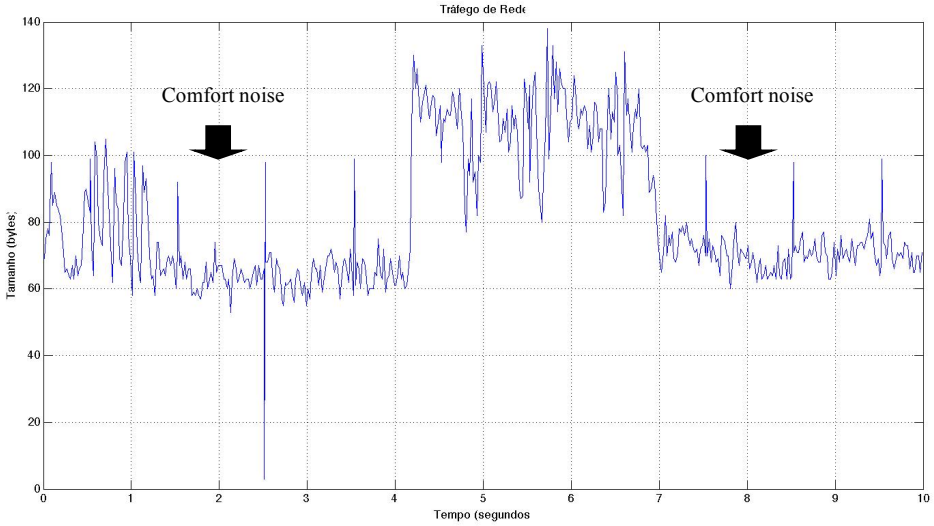

Figura 2.5 - Redução do quantidade de *bytes* transmitidos

Nas figuras 2.4 e 2.5 foi possível identificar visualmente que, durante os períodos de silêncio, no qual o *comfort noise* é utilizado, os dados continuam a ser transmitidos, porém em quantidade menor. Esta característica foi explorada por alguns autores de estudos anteriores, conforme mencionado na seção 1.1.

3. SKYPE

O Skype é um serviço de comunicação criado em 2003, em Luxemburgo, pelos mesmos criadores do *software* Kazaa[8]. Seu principal produto é o serviço de comunicação de voz entre seus assinantes. Adicionalmente, o Skype também oferece os serviços de mensagens instantâneas, conferência, vídeo-chamadas, compartilhamento de arquivos e até a integração de chamadas com a rede telefônica. Atualmente, existem versões do Skype para os sistemas operacionais apresentados na tabela 3.1.

Tabela 3.1 - Sistemas operacionais que possuem versões do Skype

Sistema Operacional	Tipo de Plataforma
Android	Móvel
BlackBerry OS	Móvel
iOS	Móvel
Mac OS X	Microcomputador
Linux	Microcomputador
Symbian	Móvel
Windows	Microcomputador
Windows Mobile	Móvel

O Skype também está presente no console de videogame portátil Sony PSP e até nos chamados televisores inteligentes. Existem ainda empresas que fabricam aparelhos telefônicos próprios para rodar o Skype ou mesmo adaptadores que podem ser usados em aparelhos convencionais.

O suporte a uma grande variedade de plataformas, aliado a uma ampla gama de recursos, torna o Skype um dos serviços de voz sobre IP mais utilizados atualmente. No final de 2010, o Skype possuía cerca de 663 milhões de usuários registrados e uma média de 145 milhões de usuários conectados por mês[9].

3.1. FUNCIONAMENTO

Como o Skype é um programa proprietário e de código fechado, seu funcionamento é documentado mediante informações disponibilizadas publicamente pelo próprio Skype e pela

[8] http://www.kazaa.com/
[9] http://www.sec.gov/Archives/edgar/data/1498209/000119312511056174/ds1a.htm#rom83085_3a

utilização de técnicas de engenharia reversa, realizadas a partir da análise do tráfego de rede gerado pelo mesmo.

O Skype utiliza uma estrutura de rede distribuída e uma arquitetura P2P (*peer-to-peer*), formada por 3 tipos de componentes:

- Nós-clientes;
- Super-nós; e
- Servidores dedicados.

Os nós-cliente são todos os equipamentos nos quais a versão cliente do *software* Skype estiver instalada, incluindo os dispositivos móveis. Sua função se limita às atividades realizadas pelo usuário por meio da *interface* gráfica do programa, como a realização de chamadas e a troca de mensagens e de arquivos.

Os super-nós também possuem a versão cliente do Skype instalada, porém, além das funções citadas anteriormente, também são responsáveis por auxiliar na localização de outros usuários do serviço. Por este motivo, para que um dispositivo possa se candidatar a ser um super-nó, o mesmo precisa atender aos requisitos relacionados a seguir (Skype, 2010):

- Poder computacional suficiente;
- Largura de banda suficiente;
- Conexão direta à *internet*, sem restrições impostas por *firewall*, *proxy* ou NAT (*Network Address Translation*); e
- Não possuir a função de super-nó desabilitada manualmente nas configurações do dispositivo.

Finalmente, os servidores dedicados são equipamentos do próprio Skype e são os responsáveis pela autenticação dos usuários, pela atualização das novas versões do *software* e pela manutenção da lista de contatos. Os servidores dedicados também podem fornecer a localização de usuários que não tenham sido localizados pelos super-nós. O resumo das atribuições de cada tipo de nó podem ser vistas na figura 3.1.

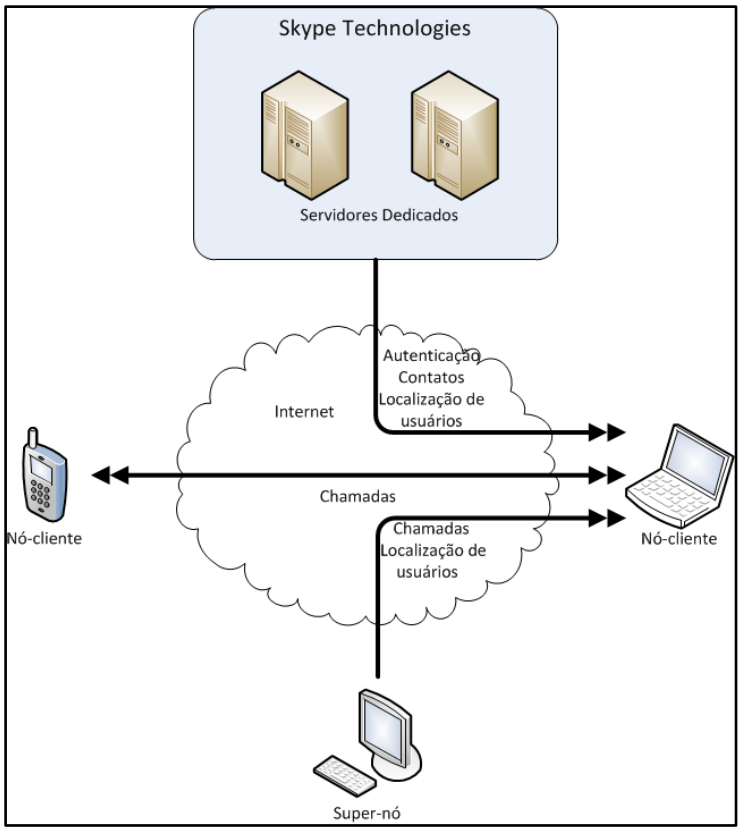

Figura 3.1 - Componentes da arquitetura do Skype

Apesar da adoção deste tipo de arquitetura otimizar a tarefa de localização de usuários, por outro lado, permite que o serviço fique suscetível a diversos tipos de óbices, como os ocorridos em dezembro de 2010, quando vários usuários enfrentaram problemas ao tentar utilizar o serviço. A causa do mesmo foi a falha em cerca de 30% dos super-nós da rede. Mas devido ao uso da arquitetura P2P, a interrupção no serviço se propagou para uma quantidade bem maior de usuários[10].

Devido às suas características, que o tornam a escolha ideal para o uso em aplicações multimídia em tempo real, o protocolo UDP é a escolha preferencial do Skype para o transporte dos sinais de voz e vídeo. O protocolo TCP é utilizado durante a fase de autenticação, porém também pode ser utilizado para o transporte do conteúdo multimídia, nas situações em que não seja possível usar o UDP, como, por exemplo, devido às restrições impostas por *firewalls*.

[10] http://blogs.skype.com/en/2010/12/cio_update.html

Durante sua instalação, o Skype escolhe arbitrariamente o número de uma porta para que passe a ser utilizada para receber conexões externas. Este número pode ser manualmente alterado pelo próprio usuário, se desejado.

3.2. SILK

O *codec* de voz atualmente adotado pelo Skype chama-se SILK, e vem sendo utilizado desde as versões 4.0 Beta para Windows, 2.8.0.438 Beta para Mac OS X e 2.1 Beta para Linux.

O primeiro esboço do SILK foi submetido à IETF em julho de 2009, o segundo em março de 2010 e o último em setembro de 2010, não estando, ainda, pronto para se tornar uma recomendação. As informações disponibilizadas nesta seção foram extraídas desta última versão (K. Vos et al., 2010), por se tratar da descrição mais detalhada encontrada do referido *codec*.

O sinal analógico de voz submetido ao SILK é codificado em intervalos de 20 milissegundos, chamados de quadros. O SILK faz ainda o uso de alguns parâmetros para ajustar o modo como o processo de codificação é realizado. Este parâmetros podem ser modificados em tempo para permitir que o *codec* se adapte aos diferentes tipos de *hardware* e às diversas condições de rede que pode encontrar. Os parâmetros em questão serão descritos nas próximas subseções.

3.2.1. Taxa de Amostragem

As amostragens realizadas pelo SILK podem ser feitas em quatro frequências diferentes: 8, 12, 16 e 24 kHz. Este valor é negociado, no início de cada sessão, pelos dispositivos envolvidos, devendo o receptor informar qual a taxa máxima que está apto a receber. Foram descritos quatro modos de operação, classificados de acordo com a frequência máxima, conforme mostrado na tabela 3.2.

Tabela 3.2 - Frequência máxima de amostragem utilizada por cada modo de operação

Modo de Operação	Taxa de Amostragem (Hz)
Narrowband (NB)	8000
Mediumband (MB)	12000
Wideband (WB)	16000
Super Wideband (SWB)	24000

O modo *narrowband* é utilizado somente durante a realização de chamadas envolvendo a rede telefônica convencional, uma vez que esta é taxa utilizada neste tipo de rede.

O valor da taxa de amostragem não é fixo e pode ser alterado pelo *codec*, se necessário.

3.2.2. Taxa de Criação de Pacotes

Durante o processo de codificação, os quadros são agrupados em conjuntos de 1 a 5 por pacote, originando 1 pacote a cada 20, 40, 60, 80 ou 100 milissegundos. As taxas de pacotes criados por segundo podem ser vistas na tabela 3.3.

Tabela 3.3 - Taxa de criação de pacotes

Tempo de criação individual de um pacote (milissegundos)	Quadros por pacote	Pacotes por segundo
20	1	50
40	2	25
60	3	16,67
80	4	12,5
100	5	10

Devido à sobrecarga causada pelos cabeçalhos utilizados durante a transmissão, quanto maior o número de pacotes transmitidos, maior será o consumo da largura de banda. Desta forma, o *codec* poderá optar por diminuir o consumo de banda por meio da redução da quantidade de pacotes. Entretanto, esta decisão poderá ocasionar na perda de uma quantidade maior de pacotes e, consequentemente, de conteúdo de voz, caso ocorram erros durante a transmissão.

3.2.3. Taxa de *Bits*

Durante uma sessão, o SILK define qual a taxa média de *bits* a ser utilizada durante a transmissão, que varia de acordo com o modo de operação estabelecido no início. Este valor pode ser alterado no decorrer da sessão e pode ser ajustado quadro a quadro, a critério do *codec*. Os valores recomendados para cada modo de operação são mostrado na tabela 3.4 (Spittka; Astrom; Vos, 2010).

Tabela 3.4 - Taxa de *bits* para cada modo de operação

Modo de Operação	Taxa de Bits
Narrowband	5 - 20 kbps
Mediumband	7 - 25 kbps
Wideband	8 - 30 kbps
Super Wideband	20 - 40 kbps

Quanto maior for a taxa de *bits*, melhor será a qualidade do áudio, porém é possível atingir uma boa qualidade com a utilização de 1 a 1,5 *bits* por amostra.

3.2.4. Taxa de Perda de Pacotes

Este parâmetro pode ser ajustado durante a sessão e permite ao *codec* se adaptar à perda de pacotes. Caso um pacote seja perdido durante a transmissão, o *codec* decidirá a quantidade de pacotes que o receptor deverá receber corretamente antes de decodificar a próxima sequência de *bits*.

3.2.5. FEC (*Foward Error Correction*)

É um recurso opcional utilizado pelo SILK visando a correção de erros durante a transmissão. Seu uso deve ser acordado no início da sessão e resulta na produção de uma sequência adicional de *bits,* que permite ao receptor reconstruir o quadro perdido sem a necessidade de retransmissão. Este excesso é adicionado aos quadros subsequentes.

A decisão pela utilização desta correção de erros é baseada em algumas estimativas, como a taxa de perda de pacotes e a capacidade do canal de comunicação.

3.2.6. Complexidade

O *codec* implementa algumas técnicas visando otimizar os recursos computacionais do dispositivo no qual estiver em execução. Por exemplo, no modo *super wideband*, o Skype pode utilizar até 80 MHz de um processador x86.

3.2.7. DTX (*Discontinuous Transmission*)

É a implementação do recurso de detecção da atividade de voz (VAD) no Skype, responsável pela redução da taxa de *bits* durante a transmissão, ao detectar períodos de silêncio ou sons em segundo plano. Esta técnica resulta na codificação de um quadro a cada 400 milissegundos, ao invés dos usuais 20, e, consequentemente, em pacotes menores.

3.3. SEGURANÇA

Todas as comunicações realizadas entre os nós do Skype são criptografadas, utilizando o algoritmo AES. Também é utilizado um esquema de chaves públicas que usa certificados RSA. A chave pública dos servidores dedicados é distribuída juntamente com o próprio *software*.

Apesar do forte esquema de criptografia, é possível identificar correlações entre o período de silêncio do arquivo de áudio original e os tamanhos dos pacotes, conforme mencionado anteriormente nas seções 1.1 e 2.4. Este fato demonstra que, por ser determinístico, o Skype não é tão seguro como se imagina a princípio.

O Skype também deixa alguns rastros sobre sua utilização, armazenados localmente, em diretórios do disco rígido do usuário, conforme mostrado na tabela 3.5.

Tabela 3.5 - Localização dos arquivos contendo informações sobre a utilização do Skype

Sistema Operacional	Diretório
Linux	%appdata%\skype
Mac OS X	~/Library/Application Support/Skype
Windows	~/.Skype

Dentre os arquivos quem contêm informações relevantes sobre a utilização do Skype, merecem destaque os arquivos “config.xml”, “shared.xml” e “main.db”. O primeiro está no formato XML (Extensible Markup Language) e armazena informações sobre as configurações do programa, como os nomes dos dispositivos de áudio utilizados durante uma chamada.

Já o arquivo “shared.xml”, que também está no formato XML, contém a relação de endereços IP e números de portas relativos aos super-nós utilizados pelo o usuário do computador em questão. Os 200 endereços relacionados estão no formato hexadecimal e são constantemente atualizados.

Por sua vez, o arquivo “main.db” está localizado em um nível abaixo dos anteriores, em um subdiretório cujo nome é igual ao nome do usuário no serviço Skype. Diferentemente dos demais, este arquivo é um banco de dados no formato SQL Lite. Dentre as informações que estão disponíveis no mesmo, podem ser citadas: a lista de contatos, o histórico de chamadas, as mensagens trocadas e os arquivos enviados e recebidos pelo usuário.

4. REDES NEURAIS ARTIFICIAIS

A rede neural artificial (RNA) é uma abordagem computacional aplicada na resolução de problemas complexos, como o controle do tráfego aéreo, o reconhecimento da fala e o diagnóstico de doenças, como o câncer (Beale, M. H.; Hagan, M. T.; Demuth, H. B., 2010). O seu funcionamento foi inspirado no comportamento do cérebro humano, que é capaz de responder à situações novas, utilizando a experiência adquirida anteriormente, a partir do processamento de uma série de eventos externos.

O cérebro humano contém bilhões de células especialistas, chamadas de neurônios, que são responsáveis processamento das informações. A figura 4.1 mostra o modelo de um neurônio humano e seus componentes:

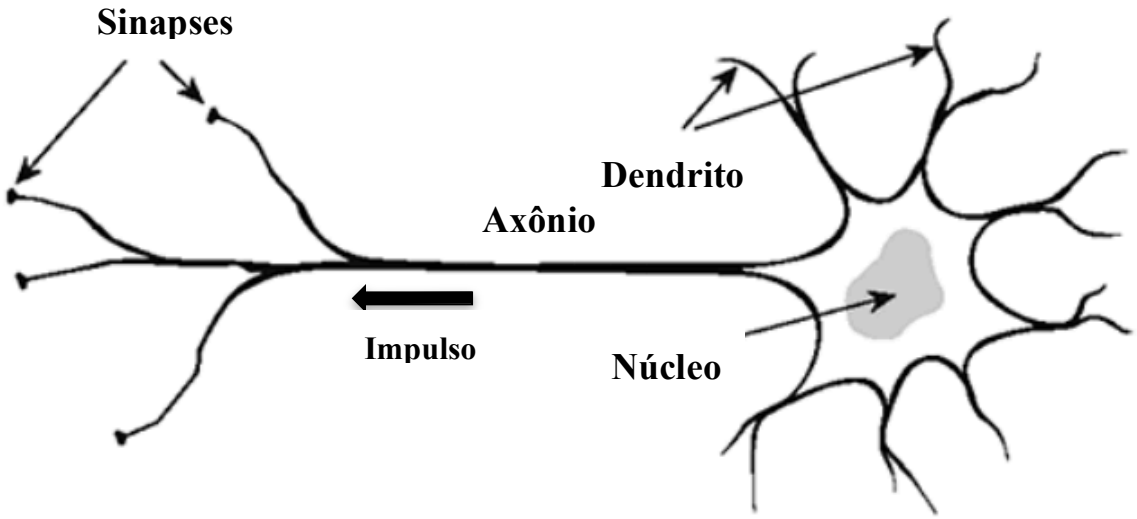

Figura 4.1 - Representação do neurônio humano (Buckland, 2002)

Um neurônio se conecta com diversos outros neurônios, formando uma rede neural natural. Essa conexão é realizada por meio das sinapses, que interligam o axônio de um neurônio aos dendritos dos próximos, transmitindo a informação através de impulsos cerebrais.

A representação do neurônio artificial, mostrada na figura 4.2, foi influenciada por sua contraparte natural. O neurônio artificial é uma unidade de processamento capaz de realizar cálculos simples, a partir de valores de entrada, e apresentar o resultado na forma de saídas.

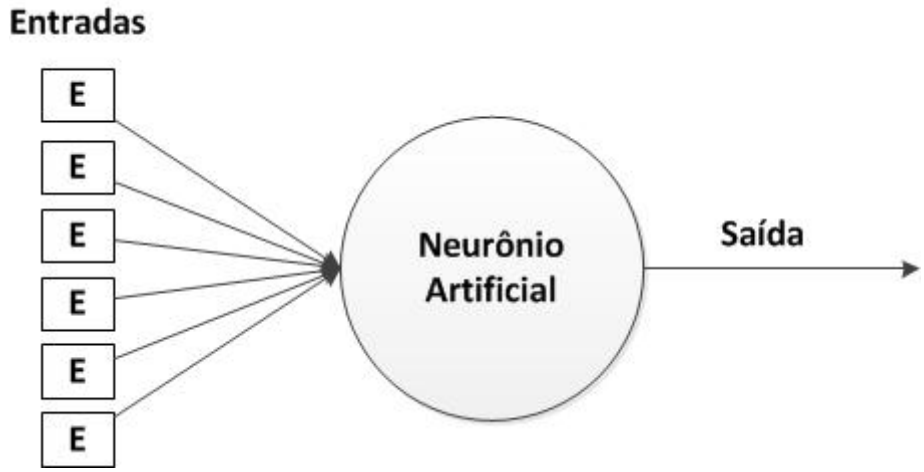

Figura 4.2 - Representação do neurônio artificial

Um avanço ao modelo de neurônio artificial foi a introdução do conceito de *perceptron*, a partir do qual se passou a adotar o uso de pesos às conexões (Rojas, 1996), e de um valor constante denominado *bias*, que não é afetado pelo valor das entradas. O *perceptron* é considerado o tipo mais simples de rede neural artificial e é utilizado em problemas linearmente separáveis. A representação do *perceptron*, com seus pesos e *bias*, pode ser vista na figura 4.3.

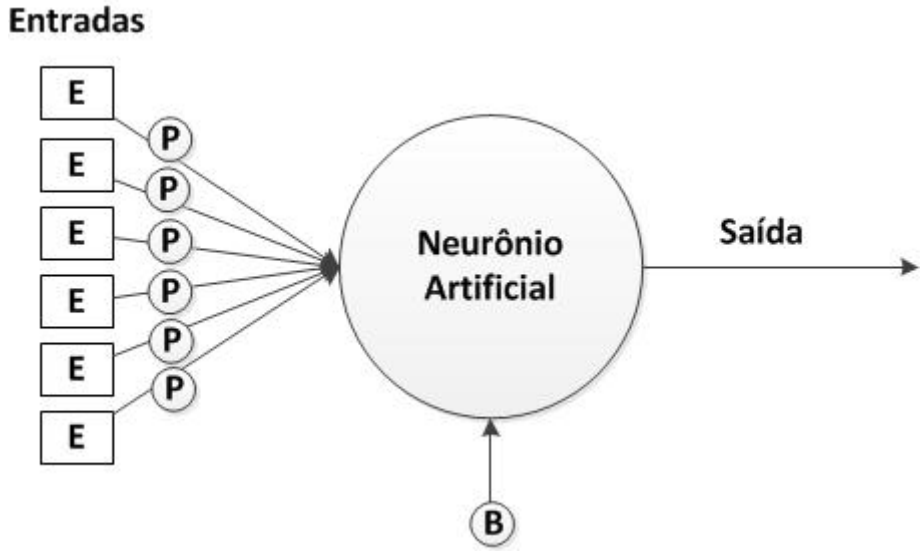

Figura 4.3 - Representação do *perceptron*

A saída "*y*" de um neurônio é calculada por meio de uma função de ativação, apresentada na equação 4.1, que recebe como entrada o resultado "*r*" do produto do valor de entrada "*x*" pelo peso "*w*", atribuído à conexão. O produto é, então, somado a um valor constante "*b*", chamado de *bias*.

$$y = f(r) = f(x.w + b) \tag{4.1}$$

Por sua vez, a saída de uma rede neural poderia ser representada pela equação 4.2.

$$y = f(r) = f(\sum_{i=0}^{n} x_i \,.\, w_i + b) \tag{4.2}$$

As funções de ativação mais comuns chamam-se degrau (limiar), linear e sigmoide (Beale, M. H.; Hagan, M. T.; Demuth, H. B., 2010), e estão apresentadas, respectivamente, nas equações 4.3, 4.4, e 4.5, e nas figuras 4.4, 4.5 e 4.6.

$$y = \begin{cases} 0, se\ r < 0 \\ 1, se\ r \geq 0 \end{cases} \tag{4.3}$$

$$f(r) = r \tag{4.4}$$

$$f(r) = \frac{1}{1 + e^{-r}} \tag{4.5}$$

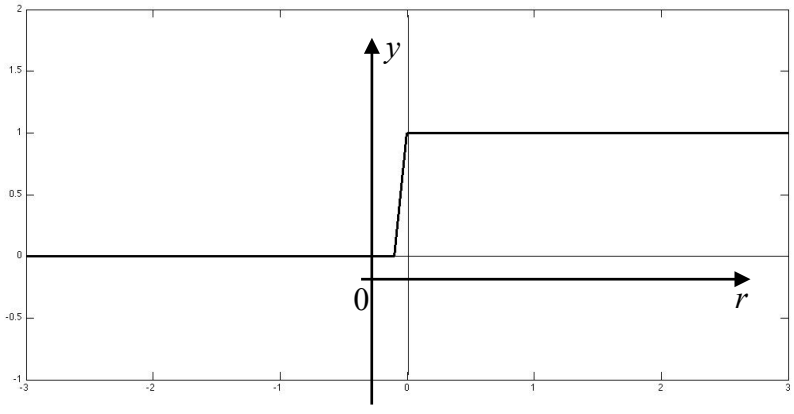

Figura 4.4 - Função degrau

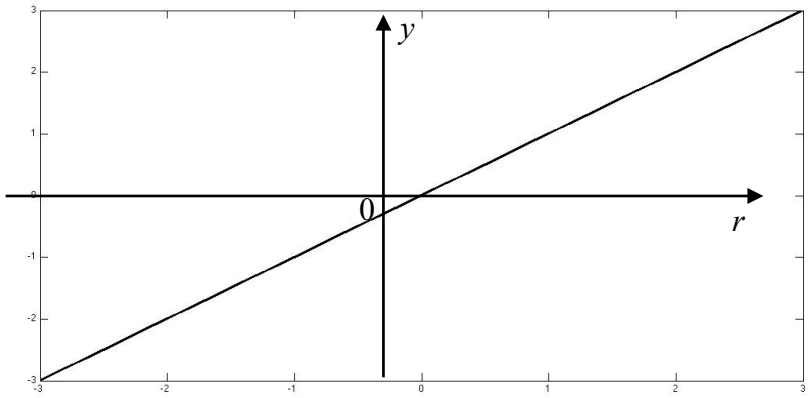

Figura 4.5 - Função linear

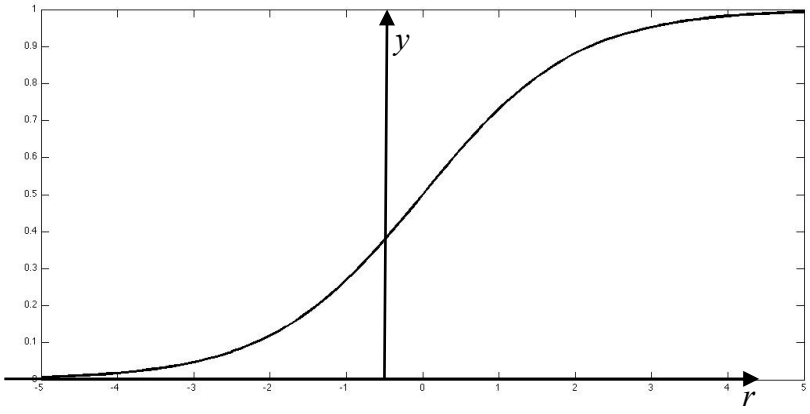

Figura 4.6 - Função sigmoide

4.1. ARQUITETURA

Considerando a organização das redes neurais artificiais em camadas, as RNA podem ser classificadas em: progressiva de camada única, progressiva multicamadas e recorrente multicamadas (Haykin, 1994).

4.1.1. Progressiva de camada única (*Single-Layer Feedforward*)

As redes neurais projetadas com esta arquitetura possuem apenas uma camada capaz de realizar o processamento, que é a camada de saída. A informação flui de maneira progressiva, apenas no sentido da camada de entrada para a de saída. Os *perceptrons* são exemplos de tipos de rede desta arquitetura, apresentada na figura 4.7.

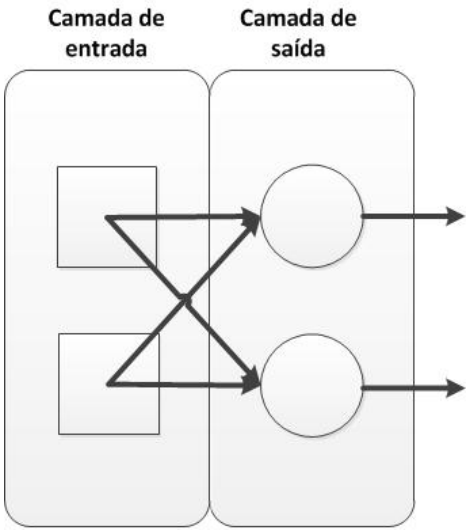

Figura 4.7 - RNA progressiva de camada única

4.1.2. Progressiva multicamadas (*Multilayer Feedforward*)

Diferentemente do modelo anterior, nesta arquitetura são introduzidas uma ou mais camadas intermediárias de neurônios, ou seja, capazes de realizar o processamentos das informações. A informação, porém, continua trafegando em um sentido apenas. A saída de uma camada serve como entrada para a camada seguinte. Este tipo de arquitetura é mais complexa e é a recomendada quando a quantidade de entradas é grande (Haykin, 1994). *Perceptrons* multicamadas são exemplos de redes deste tipo de arquitetura, mostrada na figura 4.8.

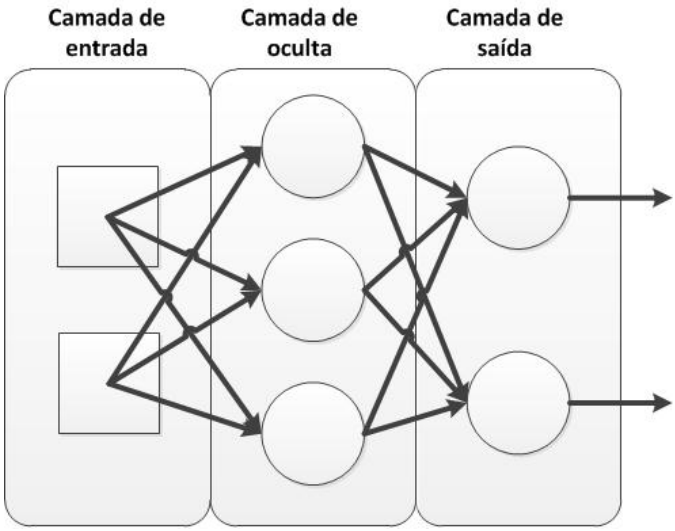

Figura 4.8 - RNA progressiva multicamadas

4.1.3. Recorrentes (*Feedback*)

Também conhecida como interativa ou cíclica, a principal diferença entre as redes recorrentes e as progressivas é a informação que pode seguir em mais de um sentido, realimentando outros neurônios como entrada. Redes deste tipo, ilustrado na figura 4.9, podem ou não conter camadas ocultas, sendo que a utilização destas tem influência direta no desempenho da rede (Haykin, 1994).

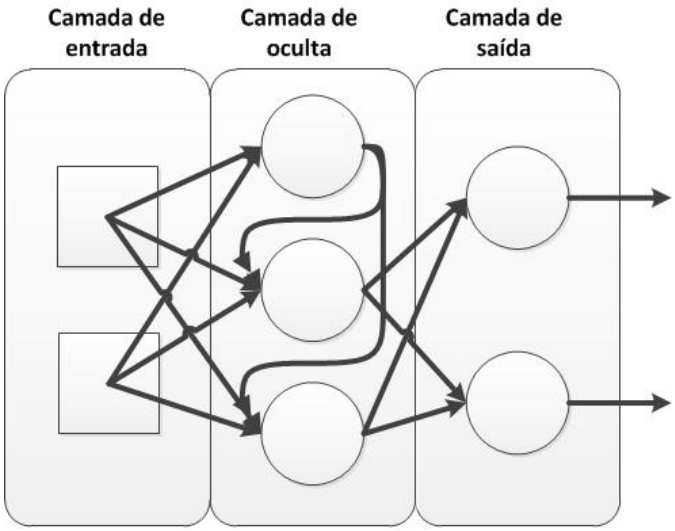

Figura 4.9 - RNA recorrente

4.2. TREINAMENTO

O treinamento ou aprendizado de uma RNA consiste na utilização de um conjunto de regras para ajustar os pesos e o *bias* das conexão da rede, para que o resultado da mesma atinja os valores esperados. O objetivo do treinamento é permitir que a RNA seja capaz de generalizar a solução de um problema a partir de um conjunto limitado de entradas e não a partir de todos as entradas possíveis. Para isso, a utilização de um conjunto de entradas representativas e a escolha correta dos parâmetros da rede são importantes para evitar o *overfitting*. Este problema ocorre quando a rede apresenta resultados satisfatórios apenas para as entradas já conhecidas, não sendo capaz de generalizar a solução para novas entradas.

O tipo de treinamento de uma RNA pode ser classificado em supervisionado e não supervisionado.

4.2.1. Treinamento Supervisionado

No treinamento supervisionado, a rede tem acesso aos valores esperados que sejam gerados pelas saídas. A rede, então, compara os valores esperados aos valores efetivamente obtidos, sendo o resultado chamado de erro. Essa comparação é realizada em cada momento do treinamento, resultando no ajuste dos pesos das conexões, de acordo com o algoritmo escolhido, com o objetivo de aproximar as saídas obtidas das saídas desejadas. O algoritmo utilizado no treinamento supervisionado de *perceptrons* é chamado de regra delta, mostrado na equação 4.6, na qual δ representa o erro calculado, ou seja, a diferença entre o valor esperado e o valor obtido.

$$\Delta w_i = x_i . \delta \tag{4.6}$$

Outro algoritmo bastante utilizado em redes deste tipo chama-se retropropagação (*backpropagation*), que é uma generalização da regra delta, utilizado em redes multicamadas. Neste caso, o erro é calculado, primeiramente, para a camada de saída, para, em seguida, ser propagado para a camada oculta predecessora. O algoritmo de retropropagação é mostrado na equação 4.7.

$$\Delta w_j = \sum_{i=1}^{n+1} w_{ij} . \delta_i \tag{4.7}$$

O cálculo do erro para a camada de saída pode ser visto na equação 4.8.

$$\delta_j = y_j(1 - y_j).(d_j - y_j) \tag{4.8}$$

Por sua vez, o cálculo do erro para as camadas ocultas pode ser visto na equação 4.9.

$$\delta_j = y_j(1 - y_j) \sum_{q=1}^{m} w_{jq} \delta_q \tag{4.9}$$

Outro método comumente utilizado no treinamento supervisionado de rede neurais é o cálculo do erro quadrático médio (*mean-squared error*) da diferença entre o valor produzido pela saída e o valor esperado, mostrado na equação 4.10.

$$MSE = \frac{1}{n}\sum_{i=1}^{n} \delta^2 \tag{4.10}$$

4.2.2. Treinamento Não Supervisionado

Diferentemente do tipo anterior, durante o treinamento não supervisionado, a rede não toma conhecimento dos valores desejados para a saída. Desta forma, o ajuste dos pesos não é feito por meio do cálculo do erro, mas sim em função das semelhanças entre os valores de entrada apresentados. Dentre os exemplos de redes neurais, que utilizam o treinamento não supervisionado, podem ser citadas os mapas auto organizáveis (*self organizing maps* - *SOM*) de Kohonen.

5. EXPERIMENTO

Os experimentos realizados durante o desenvolvimento deste trabalho demonstraram que é possível identificar certas informações sobre o conteúdo de conversas realizadas através do Skype, apesar do esquema de criptografia utilizada pelo mesmo. O primeiro teve como objetivo a identificação do idioma utilizado durante a transmissão, enquanto que o segundo teve como objetivo identificar a presença de um determinado locutor na chamada.

Os dois experimentos foram divididos em cinco fases, mostradas na figura 5.1.

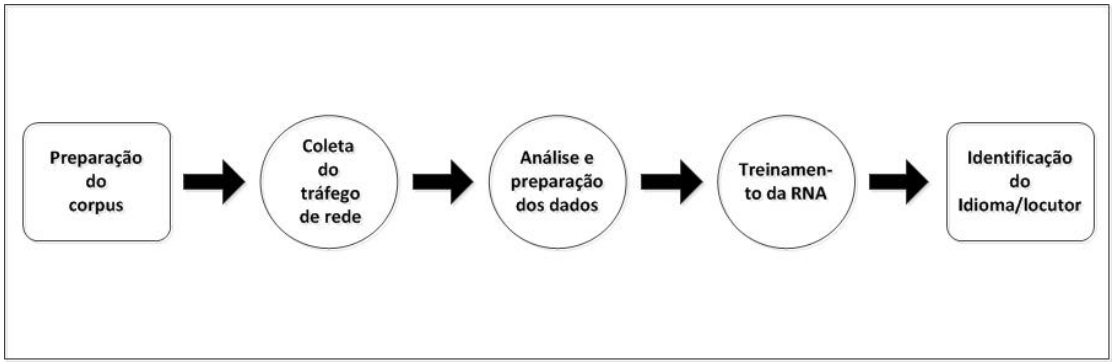

Figura 5.1 - Fases dos experimentos

5.1. PREPARAÇÃO DO CORPUS

Para a realização da primeira parte do experimento, a identificação do idioma, foi utilizado o conjunto de arquivos de áudio denominado "22 *Language Corpus*"[11], disponibilizado pelo *Center for Spoken Language Understanding*, da Universidade de Saúde & Ciência de Oregon, localizado na cidade norte-americana de Portland. Apesar do nome, a versão 1.2 deste *corpus*, que foi a utilizada durante o experimento, era composta por um conjunto de arquivos contendo discursos em 21 idiomas, pois o francês não estava disponível.

A coleta das conversações foi realizada através de ligações telefônicas, nas quais os locutores dos diferentes idiomas respondiam as mesmas perguntas, como: "qual o seu endereço" e "qual a sua última refeição".

No primeiro experimento foram utilizados os arquivos referentes aos discursos de 396 locutores, divididos em 99 locutores de cada um dos seguintes idiomas: alemão, espanhol,

[11] http://www.cslu.ogi.edu/corpora/22lang/

inglês e português. Para cada idioma foram utilizados 90 locutores para o treinamento da rede e 9 para a validação da mesma. A duração dos discursos variava entre 0’35” e 3’53”.

Já para a segunda parte do experimento, a identificação do locutor, além dos arquivos referentes a 90 locutores de língua portuguesa do *corpus* mencionado anteriormente, foram incluídos 10 arquivos do programa “Café com a Presidenta”, contendo gravações da Presidente da República, Dilma Rousseff. Os arquivos deste programa tiveram de ser editados para excluir a participação do apresentador. Do total de 100 arquivos, 90 foram reservados para o treinamento e 10 para a validação da RNA. Os arquivos possuíam duração mínima de 0’50” e máxima de 02’17”.

5.2. COLETA DO TRÁFEGO DE REDE

Para a realização dos experimentos foram utilizados dois microcomputadores contendo a versão 10.6, do sistema operacional Mac OS X, de codinome *Snow Leopard*. Ambos foram conectados, através de portas *gigabit ethernet*, a um aparelho Apple AirPort Extreme que, por sua vez, foi conectado à internet por meio de um *cable modem*. A conexão à *internet* foi compartilhada através da técnica NAT e possuía largura de banda de 500 kbps. A topologia do laboratório pode ser vista na figura 5.2.

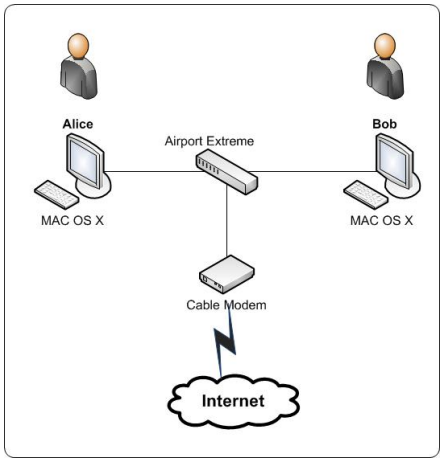

Figura 5.2 - Topologia do laboratório utilizado durante os experimentos

Os arquivos do *corpus* estavam armazenados no computador de Alice e cada chamada consistia nos passos mostrados na figura 5.3.

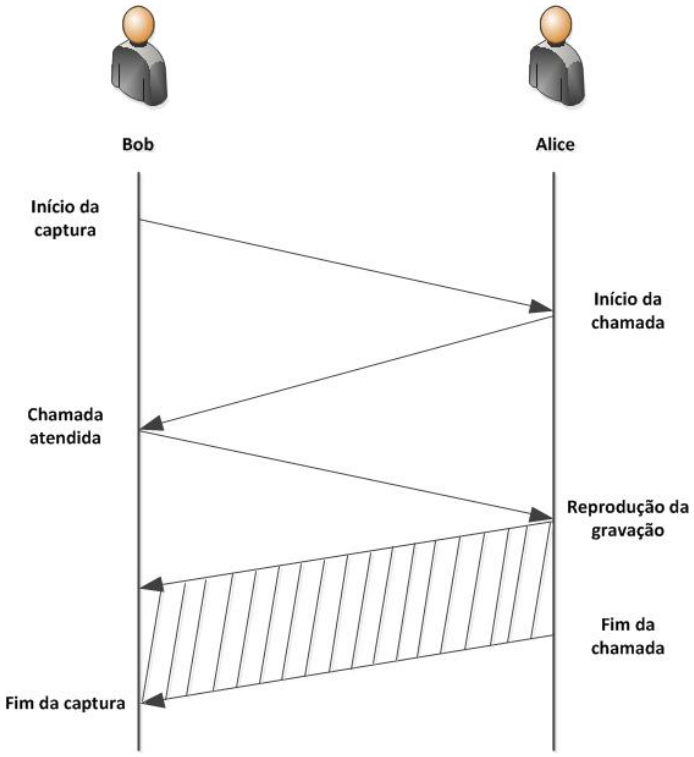

Figura 5.3 - Etapas da chamada

O processo de captura do tráfego de rede foi realizado com a utilização do *software* TCP Dump e os resultados foram gravados em arquivos individuais, um para cada chamada. Cada uma destas foi realizada com a utilização do protocolo UDP, da versão 5.1.0.935 do Skype para Mac OS X e do *codec* SILK, conforme foi apontado na figura 5.4.

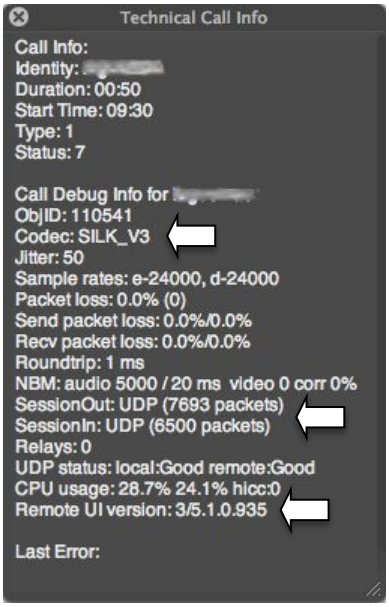

Figura 5.4 - Janela contendo as informações técnicas sobre a chamada

Todas as chamadas realizadas duramente os experimentos ocorreram por meio de conexões ponto a ponto, entre os computadores de Alice e Bob, e com o uso do protocolo UDP, conforme ilustrado na figura 5.5.

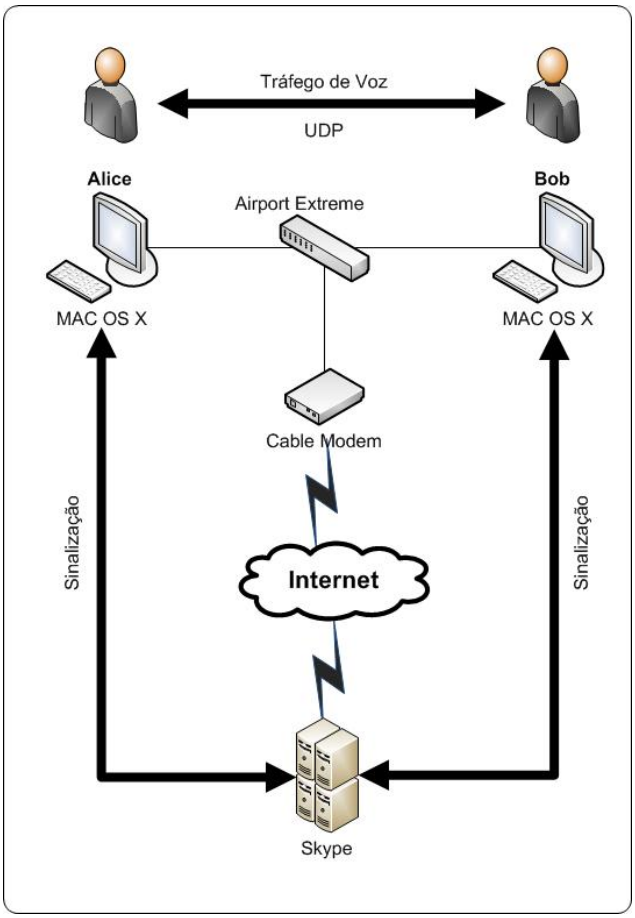

Figura 5.5 - Conexão ponto a ponto entre os comutadores

Apesar do laboratório ter utilizado apenas a versão do Skype para MAC OS X, também foram realizados alguns testes com a versão 5.5.0.124 do Skype para Windows. Os resultados destes testes permitiram concluir que, a partir de uma mesma gravação, os pacotes gerados pelos dois sistemas operacionais é semelhante, conforme demonstrado na próxima seção.

A identificação correta do tráfego de rede gerado pelo Skype pode requerer esforços consideráveis, pois seu protocolo é proprietário e não possui uma assinatura específica. Entretanto, esta tarefa não foi abordada no presente trabalho, uma vez que é o objetivo específico de outros estudos (Molnár, S.; Perényi, M., 2011). Durante a realização dos experimentos, uma vez que os números das portas de comunicação eram conhecidos, foi

possível identificar corretamente o tráfego VoIP. Os números em questão podem ser identificados por intermédio das configurações do próprio aplicativo, conforme mostrado nas figuras 5.6 e 5.7.

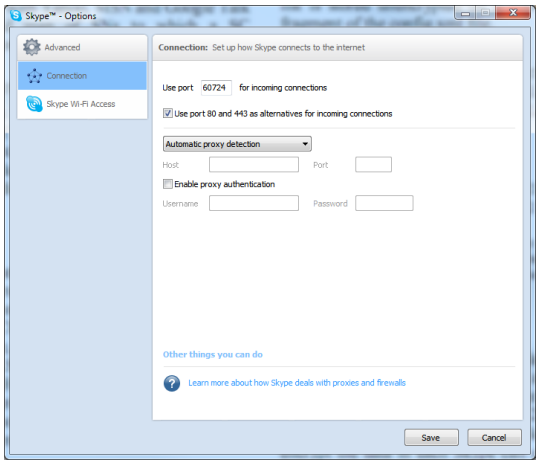

Figura 5.6 - Janela de configuração do Skype para Windows

Figura 5.7 - Janela de configuração do Skype para Mac OS X

5.3. ANÁLISE E PREPARAÇÃO DOS DADOS

A partir dos arquivos em estado bruto contendo o tráfego de rede capturado, o passo seguinte foi a realização da extração das informações contidas nos mesmos. Neste processo foi utilizado a biblioteca Net::TcpDumpLog[12], disponível no repositório CPAN (*Comprehensive Perl Archive Network*), para a linguagem interpretada Perl. Os dados sobre cada pacote da camada de aplicação foram exportados para arquivos em formato de texto, para posterior treinamento da RNA.

[12] http://search.cpan.org/~bdgregg/Net-TcpDumpLog-0.11/TcpDumpLog.pm

Durante a análise dos dados exportados, e ao compará-los aos arquivos contendo as gravações originais, foi possível perceber uma expressiva redução nos tamanhos dos pacotes capturados. Esta diminuição coincidia com os períodos de silêncio existentes nos arquivos do *corpus*, conforme mostrado nas figuras 5.8, 5.9 e 5.10.

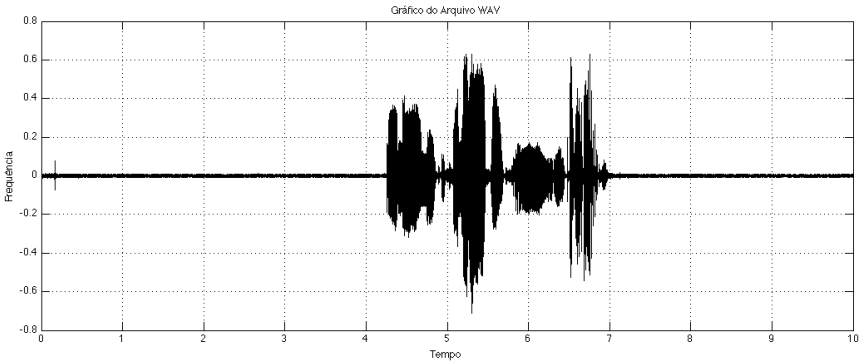

Figura 5.8 - Gravação original

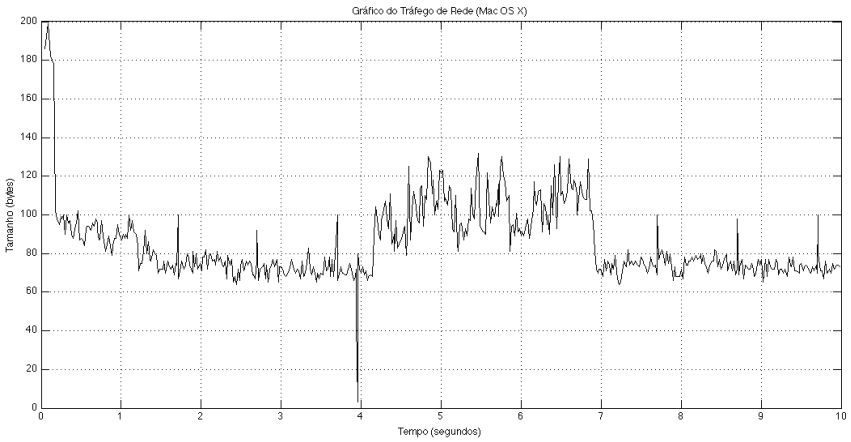

Figura 5.9 - Tráfego de rede (Mac OS X)

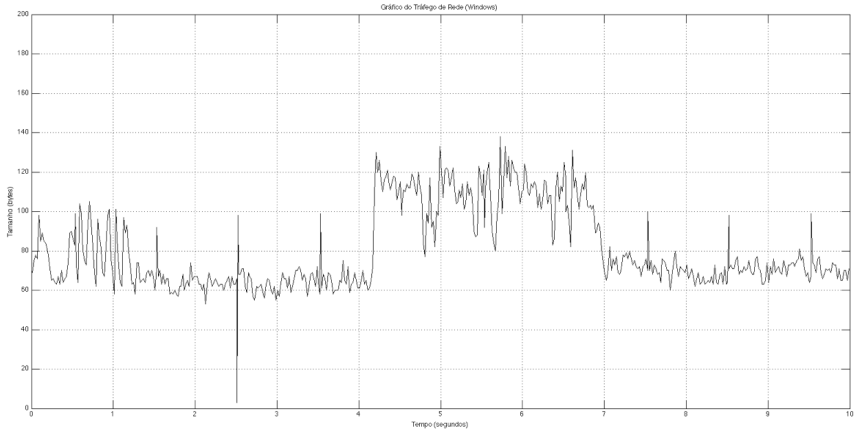

Figura 5.10 - Tráfego de rede (Windows)

Então, foi possível perceber que, salvo algumas exceções, os tamanhos dos pacotes variavam entre 100 e 160 *bytes* durante os momentos de fala. Como o Skype utiliza o recurso DTX para reduzir a taxa de *bits* com que realiza a codificação do áudio, ao detectar períodos de silêncio ou sons em segundo plano, é possível concluir que os valores fora deste intervalo não representam os momentos de fala de um locutor.

Por este motivo, este intervalo foi utilizado para filtrar os valores utilizados no treinamento da RNA responsável pela identificação de idioma. Porém, como os períodos de silêncio também podem representar pausas em um discurso, os valores menores ou iguais a 160 *bytes* foram mantidos no treinamento da RNA utilizada na identificação do locutor.

5.3.1. Identificação de Idioma

Descontando o cabeçalho UDP, foram capturados aproximadamente 238,62 *megabytes*, distribuídos em 2656553 pacotes. As quantidades e médias de pacotes e *bytes*, divididos por idioma, podem ser vistas na tabela 5.1.

Tabela 5.1 - Estatísticas sobre os pacotes capturados durante a identificação do idioma

Idioma	Total de Pacotes	Pacotes/s	Bytes	Bytes/s
Alemão	663999	49,33	62362911	4615,91
Espanhol	598661	49,64	56682865	4671,64
Inglês	808775	50,14	76071265	4723,53
Português	585118	49,63	55098882	4672,59

Também foi verificado que a distribuição dos tamanhos dos pacotes em cada idioma é similar, com a maioria dos pacotes possuindo entre 100 e 110 *bytes*. A figura 5.11 mostra essa distribuição.

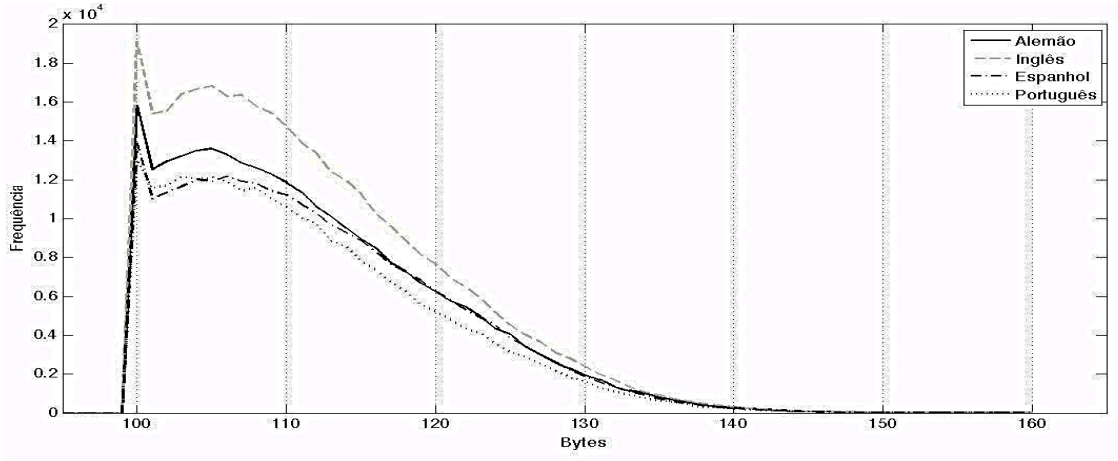

Figura 5.11 - Distribuição dos tamanhos dos pacotes por idioma

Os dados utilizados como entrada para a RNA foram os valores obtidos da distribuição da quantidade de *bytes*, compreendida entre 100 e 160, para cada pacote. Porém, assim como em uma situação real, os arquivos de áudio utilizados não continham a mesma duração. Consequentemente, os discursos mais longos produziram uma quantidade maior de pacotes. Este fato poderia causar uma distorção nos resultados, uma vez que a frequência das ocorrências aumentava de acordo com o acréscimo de pacotes. Para contornar este problema, o cálculo para se chegar ao valor de entrada foi a quantidade de pacotes distribuídos no intervalo, em razão da quantidade de pacotes de um dado locutor, conforme a equação 5.1.

$$Se\ 100 \leq bytes \leq 160: x = \frac{distribuição\ do\ pacote}{total\ de\ pacotes} \tag{5.1}$$

5.3.2. Identificação de Locutor

As estatísticas dos pacotes capturados podem ser vistas na tabela 5.2:

Tabela 5.2 - Estatísticas sobre os pacotes capturados durante a identificação do locutor

Estatística	Valores
Total de pacotes	560278
Pacotes/segundo	49,31
Bytes	52911997
Bytes/segundo	4674,74
Menor pacote	3 bytes
Maior pacote	749 bytes

Os dados utilizados como entrada para a rede foram os valores obtidos da distribuição da quantidade de *bytes*, compreendida entre 1 e 160, para cada pacote. A mesma preocupação com as diferentes durações dos arquivos de áudio mencionadas na seção anterior foram observadas no processo de identificação do locutor. Deste modo, o cálculo do valor das entradas da RNA foi o mesmo, como observa-se na equação 5.2.

$$Se\ bytes \leq 160: x = \frac{distribuição\ do\ pacote}{total\ de\ pacotes} \tag{5.2}$$

5.4. TREINAMENTO DA REDE NEURAL ARTIFICIAL

O *software* utilizado para o treinamento da RNA foi a versão 7.9 do MATLAB[13], da empresa Math Works, escolhido devido à quantidade de recursos disponíveis no "*Neural Network Toolbox*". Este pacote permite o rápido ajuste dos parâmetros da rede, bem como disponibiliza uma grande quantidade de algoritmos de treinamento.

O *corpus* utilizado durante esta fase foi dividido em três subconjuntos, empregados nas seguintes subfases (Zhang, 2010): o treinamento propriamente dito, a validação e o teste da rede. O treinamento tem como objetivo o ajuste dos pesos das conexões da RNA. Por sua vez, a validação visa verificar a capacidade de generalização da rede e evitar a memorização das entradas (*overfitting*). A validação é realizada por meio do cálculo do erro quadrático médio e resulta na performance do treinamento da rede neural artificial. Finalmente, o teste consiste em avaliar o comportamento da rede ao ser submetida à um novo conjunto de entradas.

Os resultados de cada subfase foram disponibilizados por meio de uma matriz de confusão, que é uma ferramenta utilizada para representar a precisão de um determinado classificador. As células posicionadas na diagonal principal representam as situações de verdadeiro positivo e negativo, ou seja, as situações nas quais os resultados foram classificados corretamente (Marques, 2011). Deste modo, para que um classificador possua um bom desempenho, a maior parte das ocorrências deve estar presente na referida diagonal. A figura 5.12 ilustra a matriz de confusão (Oweiss, 2010).

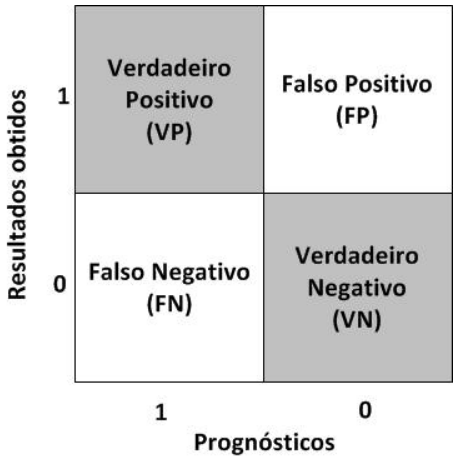

Figura 5.12 - Matriz de confusão

[13] http://www.mathworks.com/products/matlab/

A partir dos dados disponibilizados na matriz de confusão, é possível chegar aos valores da acurácia (A) da RNA, calculados através da equação 5.3 (D. Zhang et al., 2009).

$$A = \frac{VP + VN}{VP + FP + FN + VN} \tag{5.3}$$

Além da acurácia, outros valores também podem ser úteis na medição do desempenho da RNA (D. Zhang et al, 2009). O primeiro é a taxa de ocorrências positivas classificadas corretamente, chamada de taxa de verdadeiros positivos (TVP), cujo cálculo é mostrado na equação 5.4.

$$TVP = \frac{VP}{VP + FN} \tag{5.4}$$

O segundo valor é a taxa de falsos positivos (TFP), que representa a razão de ocorrências negativas classificadas incorretamente, calculada por meio da equação 5.5.

$$TFP = \frac{FP}{FP + VN} \tag{5.5}$$

Por outro lado, a taxa de verdadeiros negativos (TVN) representa as ocorrências negativas classificadas de maneira correta. Seu cálculo é realizado pela equação 5.6.

$$TVN = \frac{VN}{VN + FP} \tag{5.6}$$

Por último, a taxa de falsos negativos é a proporção de ocorrência negativas erroneamente classificadas, que é computada pela equação 5.7.

$$TFN = \frac{FN}{FN + VP} \tag{5.7}$$

5.4.1. Identificação de Idioma

Durante esta fase, do *corpus* de 396 locutores, 360 foram reservados para treinamento e 36 para o teste posterior da rede. A matriz de entrada possuía dimensões de 61 x 360. Uma segunda matriz, de dimensões 1 x 360, continha os valores desejados e foi utilizada para auxiliar no treinamento.

A normalização das entradas foi realizada por meio da submissão de seus valores à uma função de ativação, responsável por convertê-los para números contidos no intervalo [-1,1]. O referido algoritmo pode ser visto na equação 5.8, sendo que x_{maior} e x_{menor} são os maiores e menores valores da entrada em uma determinada linha da matriz, respectivamente:

$$f(x) = \frac{2.(x - x_{menor})}{x_{maior} - x_{menor}} - 1 \tag{5.8}$$

A RNA foi configurada para apresentar respostas binárias, retornando 1 para o idioma português e 0 para os demais. A arquitetura utilizada pela rede, ilustrada na figura 5.13, foi a progressiva, com *perceptrons* multicamadas, composta por uma camada de entrada, formada por 61 entradas, por uma camada oculta, composta por 7 neurônios, e por uma camada de saída, contendo 1 neurônio. A função de transferência utilizada em cada camada foi a sigmoide.

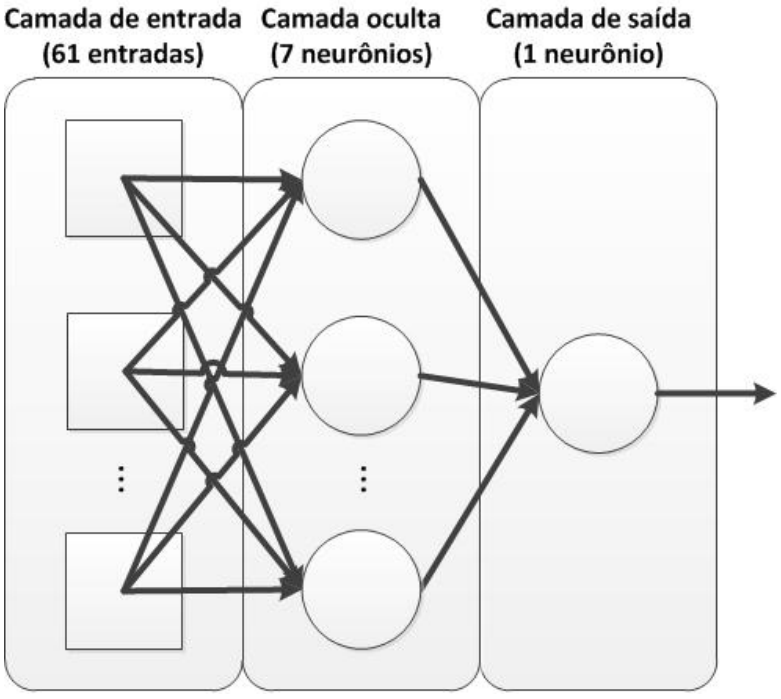

Figura 5.13 - Arquitetura da RNA utilizada para a identificação de idioma

Outras configurações de rede foram testadas durante a fase de treinamento, como a utilização de *perceptrons* de camada única e de treinamento não supervisionado, empregando-se o algoritmo de mapas auto-organizáveis de Kohonen. Porém, os índices de identificação correta do idioma ficaram na faixa de 75%.

A configuração que apresentou melhores resultados foi através do treinamento supervisionado, com a utilização do algoritmo SCG (*scaled conjugate gradient backpropagation*), uma variação do *backpropagation* tradicional, mostrado na tabela 5.3 (Møller, 1990).

Tabela 5.3 - Algoritmo SCG

1.	$escolher\ w_1\ e\ \sigma > 0, \lambda_1 > 0\ e\ \overline{\lambda_1} = 0$ $p_1 = r_1 = -E'^{(w_1)}$ $k = 1$ $sucesso = verdadeiro$
2.	se sucesso = verdadeiro $\sigma_k = \frac{\sigma}{\lvert p_k \rvert}$ $s_k = \frac{E'(w_k + \sigma_k p_k) - E'(w_k)}{\sigma_k}$ $\delta_k = p_k^T s_k$
3.	$s_k = s_k + (\lambda_k - \bar{\lambda}_k) p_k$ $\delta_k = \delta_k + (\lambda_k - \bar{\lambda}_k) \lvert p_k \rvert^2$
4.	$se\ \delta_k \leq 0$ $s_k = s_k + (\lambda_k - 2\frac{\delta_k}{\lvert p_k \rvert^2}) p_k$ $\bar{\lambda}_k = 2(\lambda_k - \frac{\delta_k}{\lvert p_k \rvert^2})$ $\delta_k = -\delta_k + \lambda_k \lvert p_k \rvert^2,\ \lambda_k = \overline{\lambda_k}$
5.	$\mu_k = p_k^T r_k, \qquad \alpha_k = \frac{\mu_k}{\delta_k}$
6.	$\Delta_k = \frac{2\delta_k [E(w_k) - E(w_k + \alpha_k p_k)]}{\mu_k^2}$

7.	$se\ \Delta_k \geq 0$ $w_{k+1} = w_k + \alpha_k p_k$ $r_{k+1} = -E'(w_{k+1})$ $sucesso = verdadeiro: \bar{\lambda}_k = 0,$ $se\ k\ mod\ N = 0, então\ ir\ para\ passo\ 1: p_{k+1} = r_{k+1}$ $senão$ $\beta_k = \frac{\lvert r_{k+1}\rvert^2 - r_{k+1} r_k}{\mu_k}$ $p_{k+1} = r_{k+1} + \beta_k p_k$ $se\ \Delta_k \geq 0.75, então\ \lambda_k = \frac{1}{2}\lambda_k$ $senão, sucesso = falso: \bar{\lambda}_k = \lambda_k$
8.	$se\ \Delta_k \leq 0.25, então\ \lambda_k = 4\lambda_k$
9.	$se\ r_k \neq 0, então\ k = k + 1\ e\ ir\ para\ passo\ 2$ $senão, ir\ para\ fim\ e\ retornar\ w_{k+1}$

As 360 gravações utilizadas durante esta fase foram divididas aleatoriamente nos seguintes subconjuntos: treinamento (80%), validação (10%) e teste (10%). As matrizes de confusão de cada subfase, nas quais "1" representa o idioma português e "0" os demais, podem ser vistas nas figuras 5.14, 5.15 e 5.16.

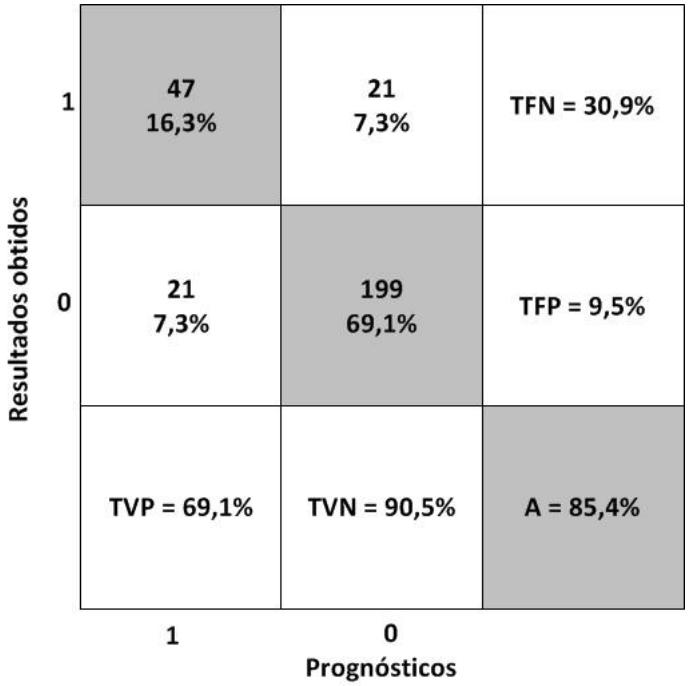

Figura 5.14 - Matriz de confusão do treinamento da RNA utilizada na identificação do idioma

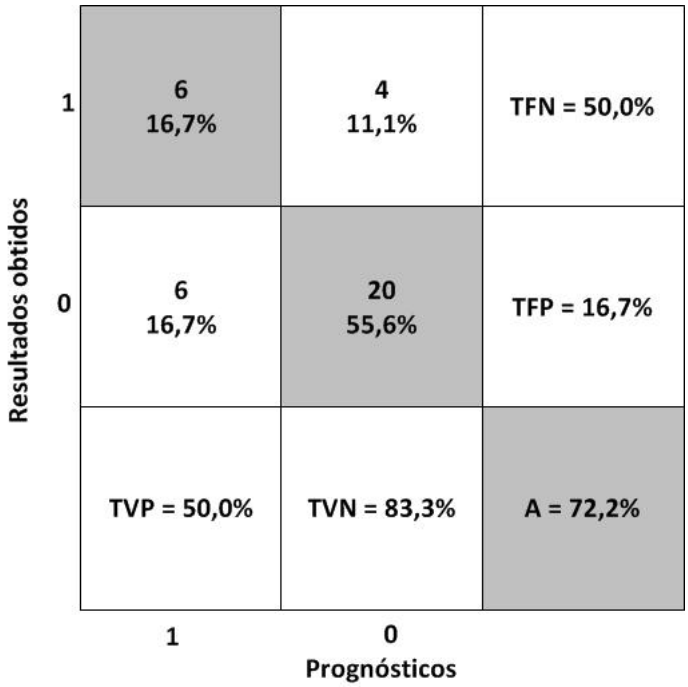

Figura 5.15 - Matriz de confusão da validação da RNA utilizada na identificação do idioma

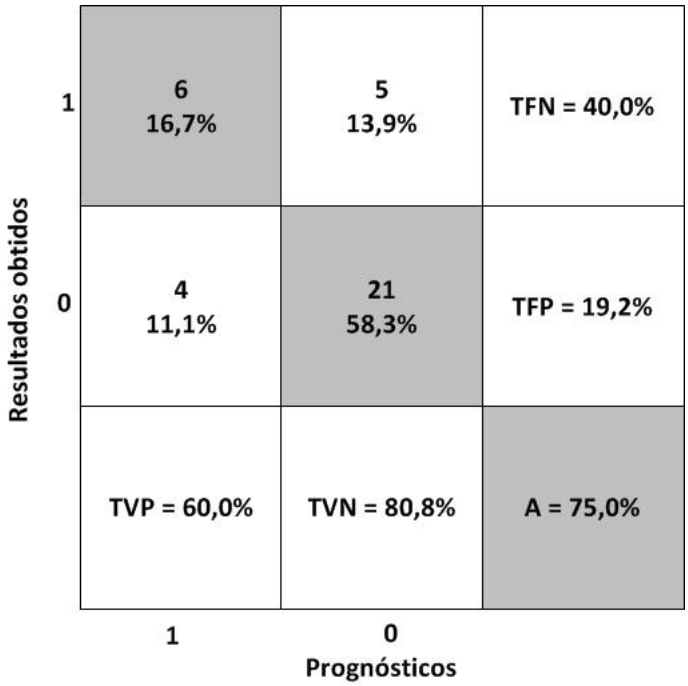

Figura 5.16 - Matriz de confusão do teste da RNA utilizada na identificação do idioma

A matriz de confusão das subfases consolidadas pode ser vista na figura 5.17.

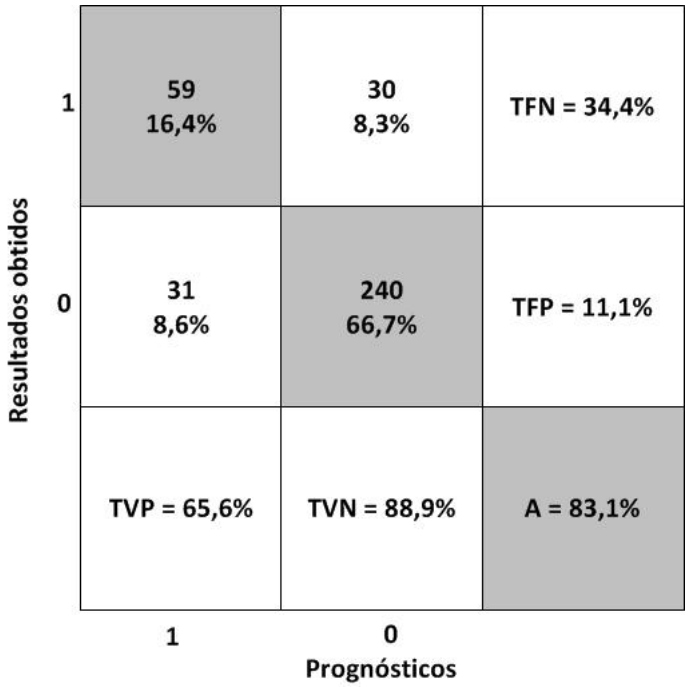

Figura 5.17 - Matriz de confusão consolidada das subfases da RNA utilizada na identificação do idioma

De acordo com a medição de performance da rede, realizada durante a subfase de validação, na 24ª interação a RNA obteve o menor erro quadrático médio, conforme apontado, por meio de círculo, na figura 5.18.

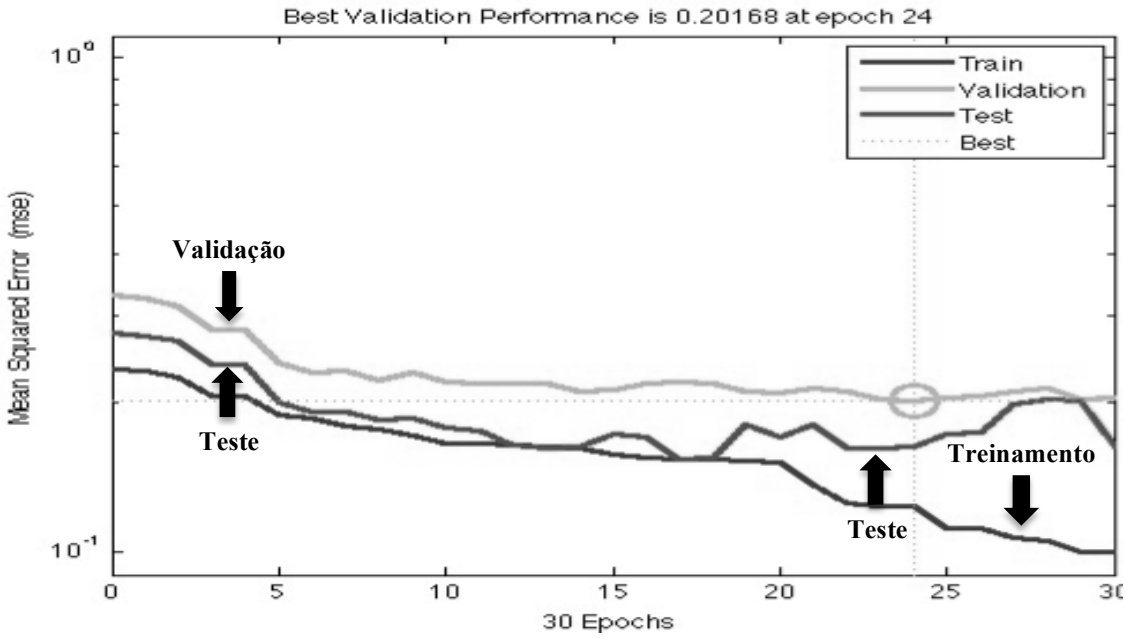

Figura 5.18 - Performance do treinamento da RNA utilizada na identificação do idioma

A comparação entre as ocorrências classificadas como verdadeiro positivo e falso positivo para cada subfase é mostrada por meio da curva ROC (*Receiver Operating Characteristic*), exibida na figura 5.19.

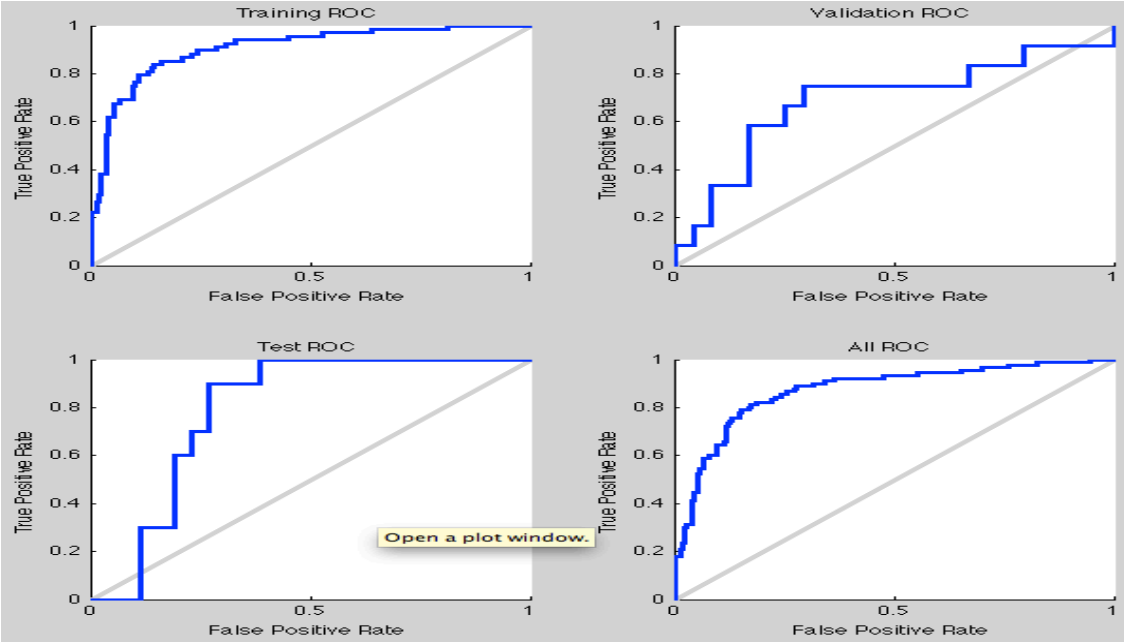

Figura 5.19 - Curva ROC do treinamento da RNA utilizada na identificação do idioma

5.4.2. Identificação de Locutor

Para esta fase foram reservados 90 locutores, do total de 100. A matriz de entrada possuía dimensão de 160 x 90 e a matriz de valores desejados tinha a dimensão de 1 x 90.

A matriz de entradas foi submetida à mesma função de ativação da RNA de identificação de idiomas, mostrada na seção anterior.

As redes foram configuradas para apresentar respostas binárias, nas quais a saída deveria apresentar 1 para o locutor procurado e 0 para os demais.

A arquitetura escolhida para a rede também foi a progressiva, com *perceptrons* multicamadas. A rede, apresentada na figura 5.20, era composta por uma camada de entrada, formada por 160 entradas, por uma camada oculta, composta de 10 neurônios, e por uma camada de saída, formada por um neurônio. A função de transferência utilizada em cada camada foi a sigmoide.

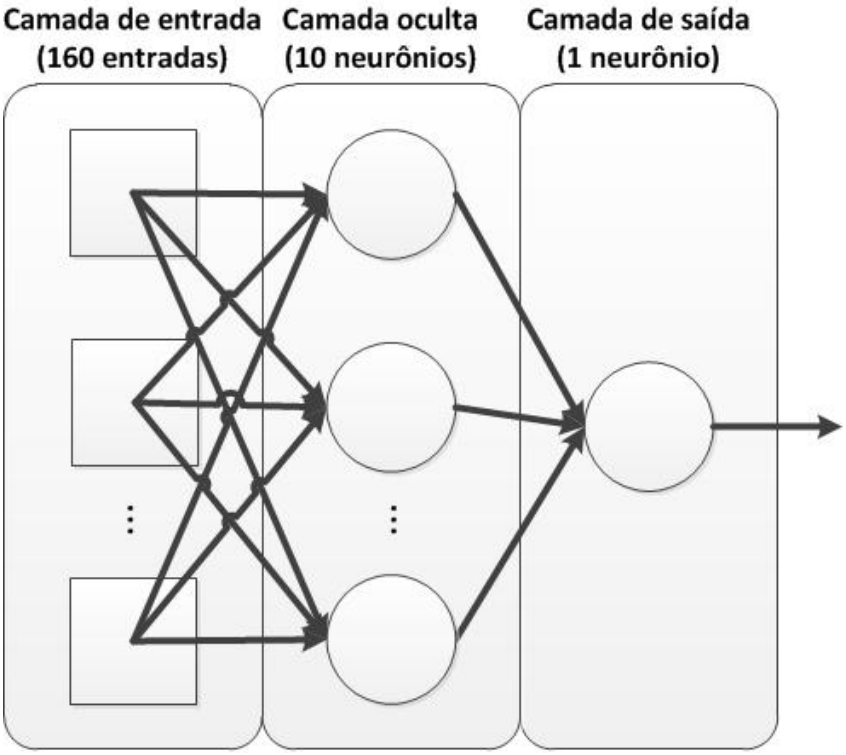

Figura 5.20 - Arquitetura da rede utilizada para a identificação do locutor

A configuração que apresentou melhores resultados foi por meio do treinamento supervisionado, com a utilização do algoritmo SCG (*scaled conjugate gradient backpropagation*), uma variação do *backpropagation* tradicional, mostrado na seção anterior.

As 90 gravações utilizadas durante esta fase foram divididas aleatoriamente nos mesmos subconjuntos mencionados na seção anterior: treinamento (80%), validação (10%) e teste (10%). Os resultados obtidos podem ser vistos nas matrizes de confusão de cada subfase, nas quais "1" representa a Presidente Dilma Rousseff e "0" os demais locutores, apresentadas nas figuras 5.21, 5.22 e 5.23.

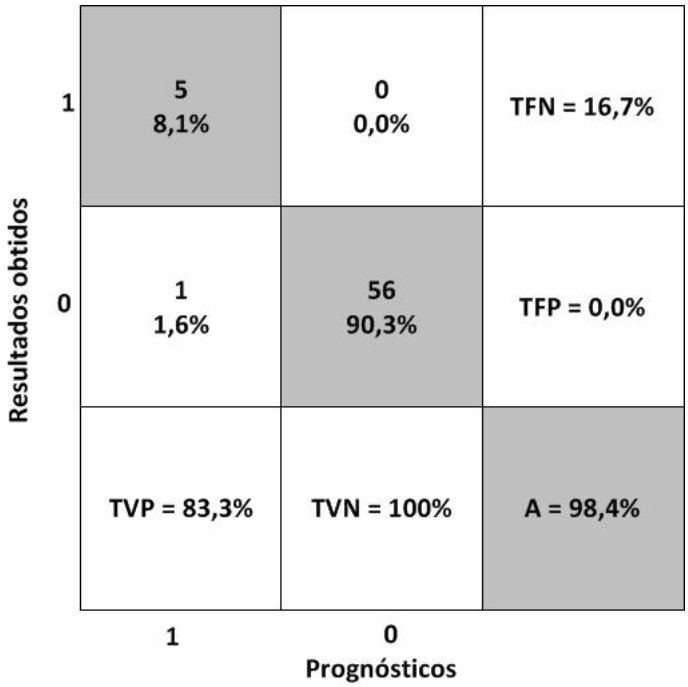

Figura 5.21 - Matriz de confusão do treinamento da RNA utilizada na identificação do locutor

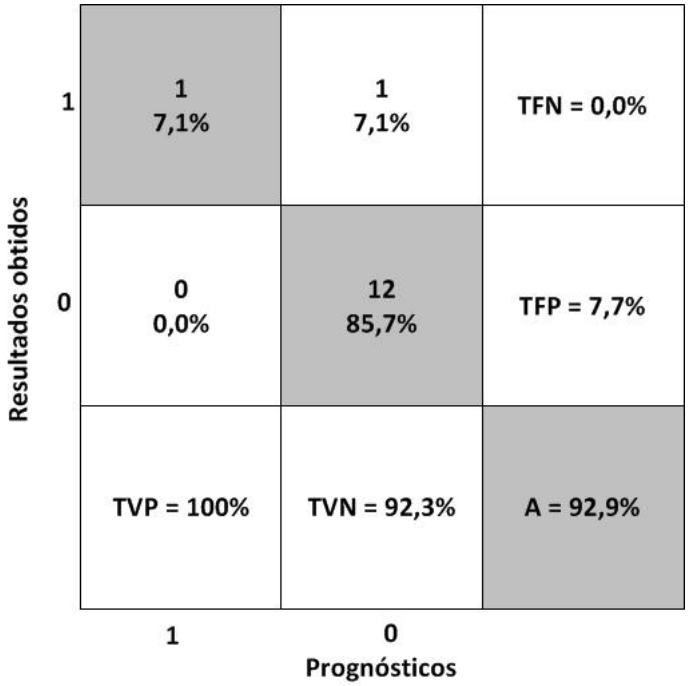

Figura 5.22 - Matriz de confusão da validação da RNA utilizada na identificação do locutor

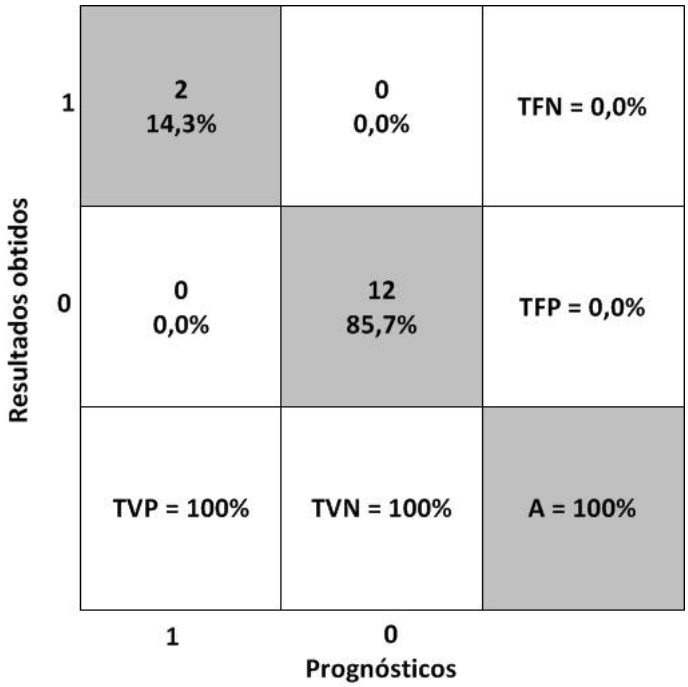

Figura 5.23 - Matriz de confusão do teste da RNA utilizada na identificação do locutor

A matriz de confusão das subfases consolidadas pode ser vista na figura 5.24.

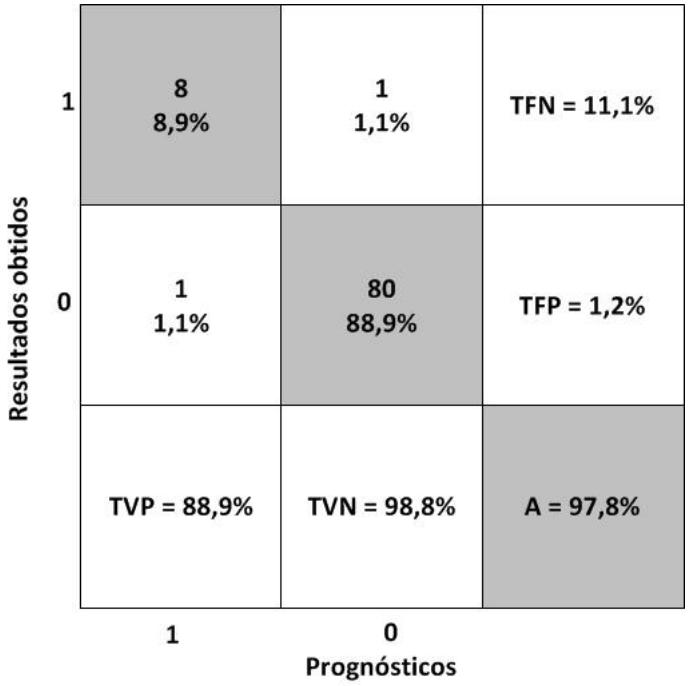

Figura 5.24 - Matriz de confusão consolidada das subfases da RNA utilizada na identificação do locutor

De acordo com a medição de performance da rede, realizada durante a subfase de validação, na 2ª interação a RNA obteve o menor erro quadrático médio, conforme apontado, por meio de círculo, na figura 5.25.

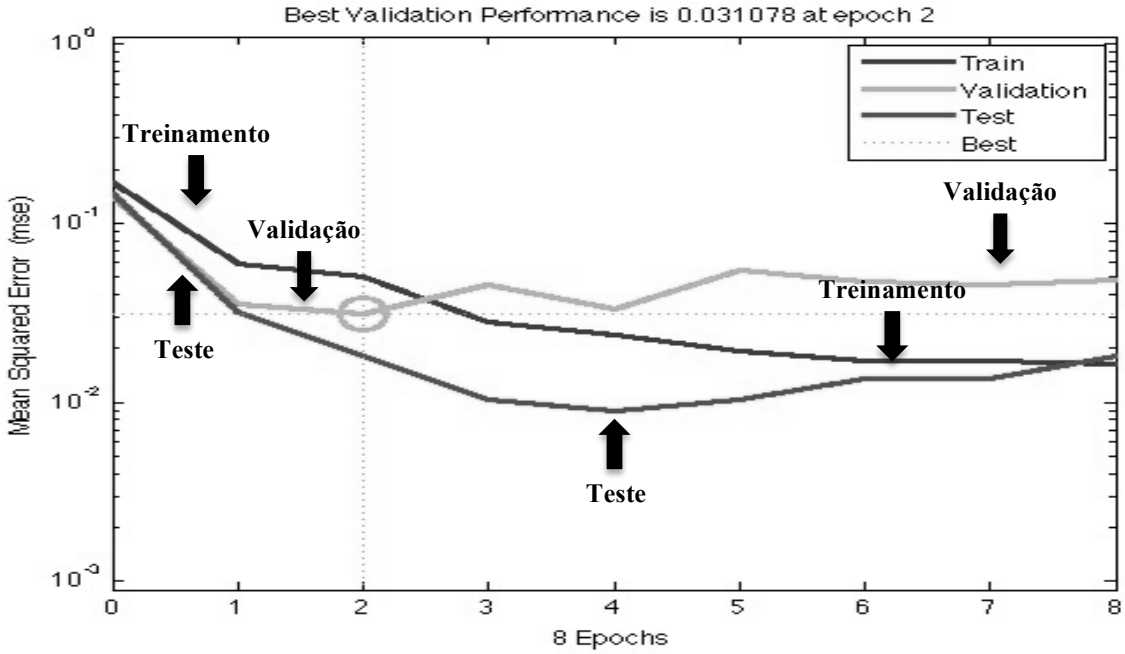

Figura 5.25 - Performance do treinamento da RNA utilizada na identificação do locutor

A curva ROC (*Receiver Operating Characteristic*) do treinamento é mostrada na figura 5.26.

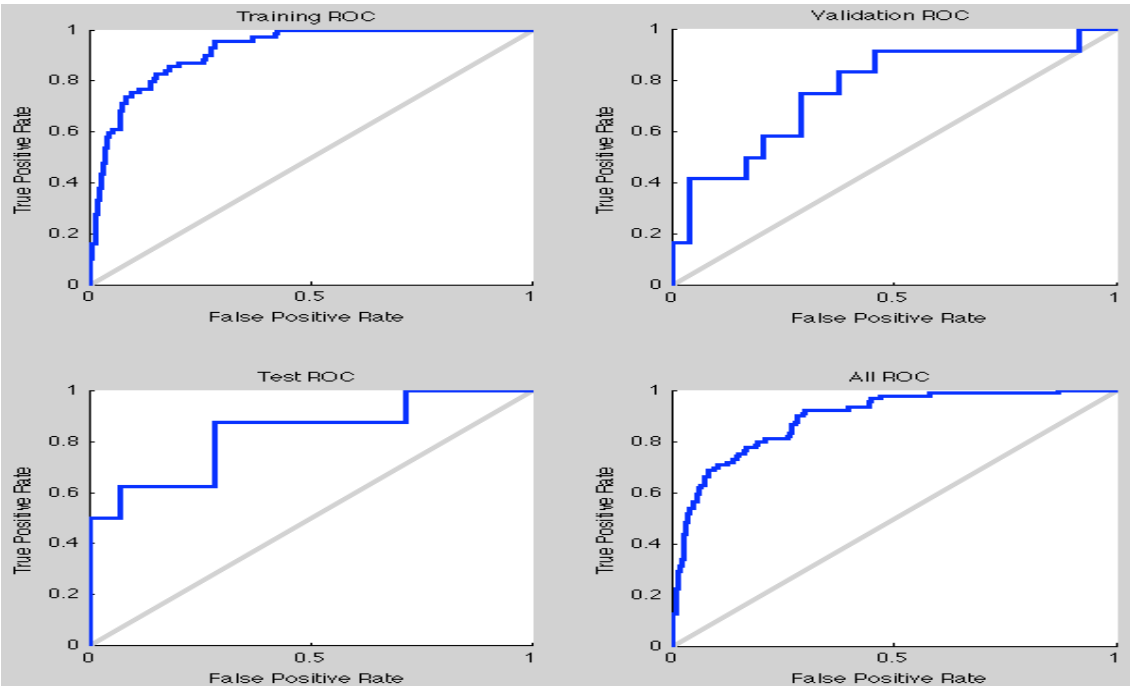

Figura 5.26 - Curva ROC do treinamento da RNA utilizada na identificação do locutor

5.5. RESULTADOS

Após o treinamento, um subconjunto do *corpus*, que ainda não havia sido utilizado, foi submetido à RNA. O resultado atingiu a marca de 80,6% de detecção correta do idioma, um pouco superior ao índice de estimativa aleatória, que seria de 75%. A figura 5.27 exibe a matriz de confusão do resultado, sendo que "1" representa o idioma português e "0" os demais.

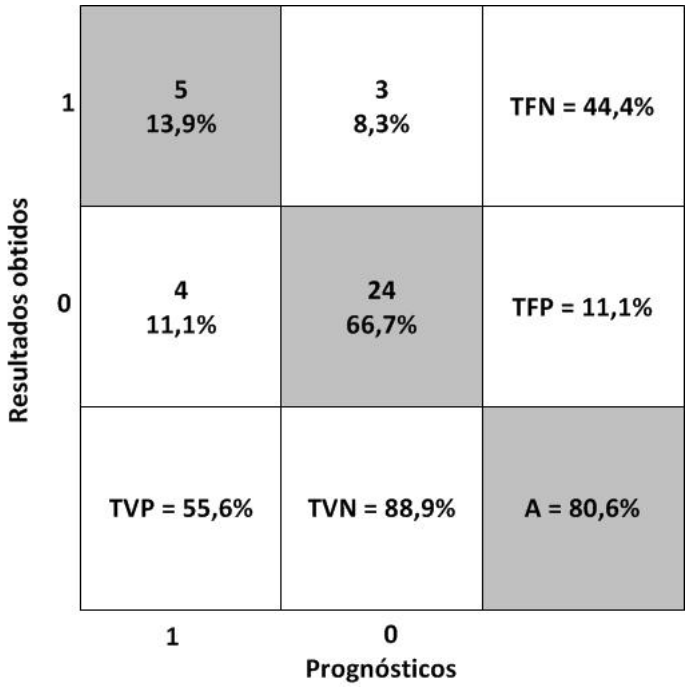

Figura 5.27 - Matriz de confusão do resultado da detecção de idioma

Concluído o treinamento da rede responsável pela identificação do locutor, o novo conjunto de arquivos, composto por 10 locutores, dos quais um era o locutor que deveria ser identificado. Desta vez o resultado alcançado foi de 90%, superando a taxa de estimativa aleatória, que seria de 10%. A figura 5.28 exibe a matriz de confusão do resultado, sendo que o valor "1" representa a Presidente Dilma Rousseff e "0" os demais locutores.

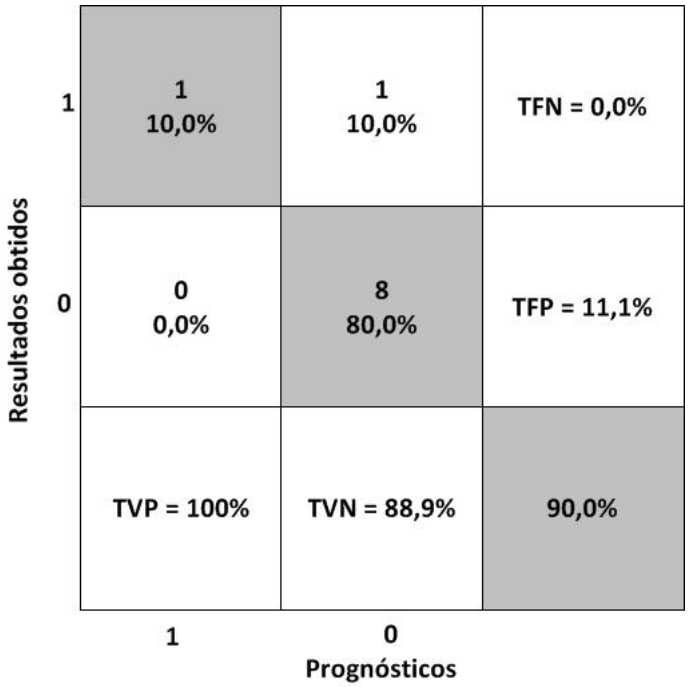

Figura 5.28 - Matriz de confusão do resultado da detecção de locutor

Testes adicionais foram realizados para verificar a confiabilidade da rede neural artificial ao ser submetida à novas gravações contendo diferentes distribuições de locutores. No primeiro foi utilizado um conjunto contendo 5 novas gravações da Presidente Dilma Rousseff e 5 de locutores desconhecidos pela rede. O resultado alcançado foi de 90%, conforme mostrado na matriz de confusão exibida na figura 5.29.

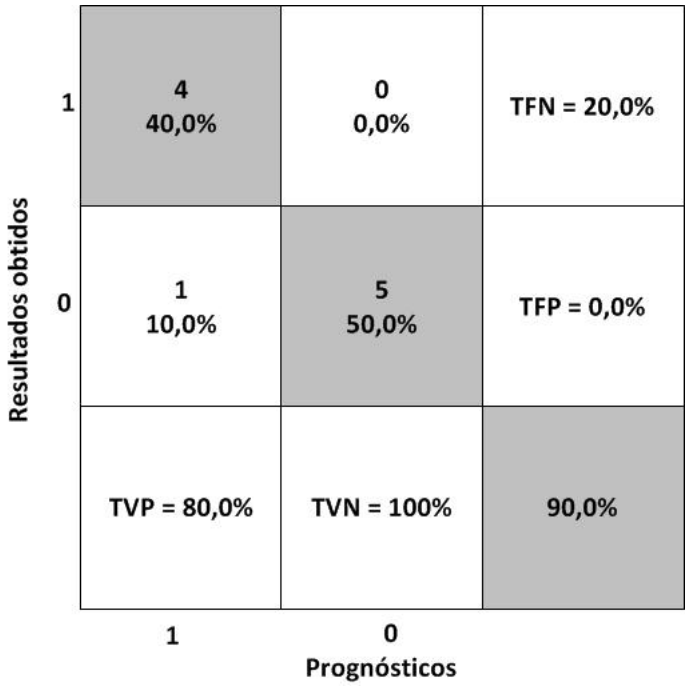

Figura 5.29 - Matriz de confusão do resultado da detecção de locutor (5:5)

No segundo teste adicional, foram utilizadas 10 gravações inéditas, todas da Presidente Dilma Rousseff. Neste caso, o índice de classificação correta atingiu 80%, conforme mostrado na matriz de confusão exibida na figura 5.30.

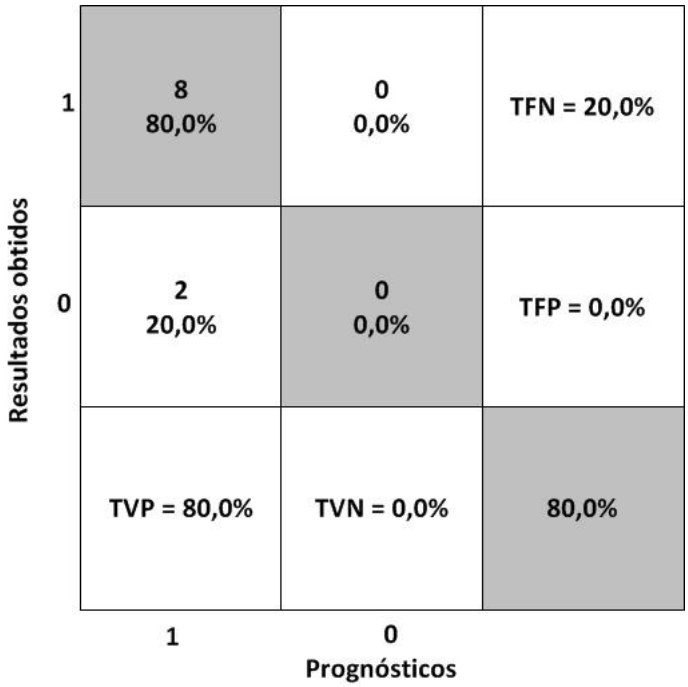

Figura 5.30 - Matrix de confusão do resultado da detecção de locutor (10:0)

6. CONCLUSÃO

O presente trabalho demonstrou que é possível utilizar o tráfego de rede, produzido durante uma chamada VoIP, para revelar informações sobre o seu conteúdo, mesmo com a utilização de criptografia.

Os testes realizados também permitiram concluir que o tráfego de rede produzido pela transmissão de uma mesma gravação, por meio das versões para Windows e Mac OS X do Skype, é semelhante.

Foi apresentado, ainda, um método para que o idioma e o locutor pudessem ser classificados independentemente da quantidade de pacotes produzidos durante uma chamada, aproximando o experimento de uma situação real, na qual a duração da chamada é indeterminada.

Durante a elaboração deste estudo, também foram apresentados alguns detalhes sobre o funcionamento do componente de codificação e decodificação de voz SILK, utilizado atualmente pelo Skype.

Este foi o primeiro trabalho que abordou a identificação de idioma e locutor de chamadas realizadas por intermédio do Skype, em conjunto com o *codec* SILK, utilizando redes neurais artificiais como classificador. Esta abordagem permitiu atingir taxas de acerto superiores aos estudos realizados anteriormente.

Os experimentos realizados permitiram identificar corretamente se o idioma utilizado em uma chamada era o português, em comparação ao alemão, ao espanhol e ao inglês, em 80,6% dos casos testados. Também foi possível identificar se um locutor monitorado anteriormente estaria realizando uma nova chamada. Neste caso, foi possível atingir uma taxa de até 90% de acurácia.

6.1. LIMITAÇÕES

Um ponto importante que deve ser considerado é que a detecção do tráfego de rede produzido pelo Skype não foi abordada, uma vez que, durante os experimentos, o protocolo e as portas de comunicação utilizados durante as chamadas eram previamente conhecidos. Existem, entretanto, outros estudos que abordam especificamente este tópico (Molnár, S.; Perényi, M., 2011).

Até a publicação deste trabalho, o codec SILK ainda possuía a condição de esboço, não sendo possível afirmar que todas suas características atuais serão mantidas na versão final.

No que diz respeito ao processo de identificação de idioma, é importante ressaltar que foram utilizados quatro idiomas para o treinamento da rede neural artificial e, portanto, a inclusão de novos idiomas pode reduzir a taxa de acerto.

6.2. TRABALHOS FUTUROS

Podem ser realizados novos testes para avaliar o comportamento do modelo apresentado, utilizando as versões mais recentes do Skype.

Outro proposta interessante é a expansão do conjunto de gravações, visando a inclusão de idiomas e locutores.

Os experimentos podem também ser estendidos para verificar a possibilidade de identificação de palavras e sentenças pronunciadas durante a comunicação, por meio da classificação dos fonemas.

Podem ser realizados, ainda, novos experimentos visando avaliar outros sistemas VoIP, incluindo os utilizados em plataformas móveis, visando analisar os resultados alcançados com larguras de banda e condições de rede diferentes.

Finalmente, podem ser estudadas técnicas de preenchimento dos pacotes visando uniformizar seu tamanho e, consequentemente, atenuar o impacto dos métodos apresentados neste trabalho.

REFERÊNCIAS BIBLIOGRÁFICAS

Abbasi, T. et al. (2005). A comparative study of the SIP and IAX VoIP protocols. Canadian Conference on Electrical and Computer Engineering, 2005, pp. 179-183. IEEE.

Alexander, J. et al. (2005). Cisco CallManager Fundamentals. Cisco Press.

IRMA – Information Resources Management Association (2010). Networking and Telecommunications: Concepts, Methodologies, Tools and Applications. IGI Global.

Baugher, M. et al. (2004). The Secure Real-time Transport Protocol (SRTP). Request for Comments (RFC) 3711. Networking Working Group.

Beale, M. H.; Hagan, M. T.; Demuth, H. B. (2010). Neural Network Toolbox 7 User's Guide. Network. The MathWorks, Inc.

Buckland, M. (2002). AI Techniques for Game Programming. Premier Press.

Dupasquier, Benoît et al. (2010). Analysis of information leakage from encrypted Skype conversations. International Journal of Information Security, 9(5), pp. 313-325.

Fries, B.; Fries, M. (2005). Digital Audio Essentials. O'Reilly Media, Inc.

Haykin, S. (1994). Neural networks: a comprehensive foundation. Prentice Hall.

ITU-T. (2009). Packet-based multimedia communications systems. International Telecommunication Union.

Lella, T.; Bettati, R. (2007). Privacy of encrypted voice-over-IP. 2007 IEEE International Conference on Systems, Man and Cybernetics, pp. 3063-3068. IEEE.

Lu, Y. (2007). On traffic analysis attacks to encrypted VoIP calls. Master's thesis, Cleveland State University (November 2009). Cleveland State University.

Marques, O. (2011). Practical Image and Video Processing Using MATLAB®. Wiley-IEEE Press.

Molnár, S.; Perényi, M. (2011). On the identification and analysis of Skype traffic. International Journal of Communication Systems, 24(1), pp. 94-117.

Møller, M. F. (1993). A scaled conjugate gradient algorithm for fast supervised learning. Neural Networks, 6(4), pp. 525-533.

Oweiss, K. (2010). Statistical Signal Processing for Neuroscience and Neurotechnology. Academic Press.

Perkins, C. (2003). RTP: Audio and Video for the Internet. Addison-Wesley Professional.

Rojas, R. (1996). Neural networks: a systematic introduction. Springer.

Rosenberg, J. et al. (2002). SIP: Session Initiation Protocol. Request for Comments (RFC) 3261. Network Working Group.

Schulzrinne, H. et al. (2003). RTP: A Transport Protocol for Real-Time Applications. Request for Comments (RFC) 3550. Network Working Group.

Simionovich, N. (2008). AsteriskNOW. (P. Publishing, Ed.).

Skype. (2010). Skype IT Administrators Guide.

Spencer, M. et al. (2010). IAX: Inter-Asterisk eXchange Version 2. Request for Comments (RFC) 5456. Independent Submission.

Spittka, J.; Astrom, H.; Vos, K. (2010). RTP Payload Format and File Storage Format for SILK Speech and Audio Codec.

Stallings, W. (2010). Cryptography and Network Security - Principles and Practice (5th ed.). Prentice Hall.

Vos, K.; Jensen, S.; Soerensen, K. (2010). SILK Speech Codec.

Wright, Charles V. et al. (2007). Language identification of encrypted VoIP traffic: Alejandra y Roberto or Alice and Bob? Proceedings of 16th USENIX Security Symposium on USENIX Security Symposium, 4. USENIX Association.

Wright, Charles V. et al. (2008). Spot Me if You Can: Uncovering Spoken Phrases in Encrypted VoIP Conversations. 2008 IEEE Symposium on Security and Privacy (sp 2008), pp. 35-49. IEEE.

Zhang, D. et al. (2009). Advanced Pattern Recognition Technologies with Applications to Biometrics. IGI Global.

Zhang, M. (2010). Artificial Higher Order Neural Networks for Computer Science and Engineering: Trends for Emerging Applications. IGI Global.