



Pós-Graduação em Ciência da Computação

Oberdan Alves de Almeida Junior

Beck: Um Chatbot Baseado na Terapia Cognitivo-Comportamental para Apoiar Adolescentes com Depressão



Universidade Federal de Pernambuco

posgraduacao@cin.ufpe.br

<http://cin.ufpe.br/~posgraduacao>

RECIFE

2017

Oberdan Alves de Almeida Junior

**Beck: Um Chatbot Baseado na Terapia
Cognitivo-Comportamental para Apoiar Adolescentes
com Depressão**

Este trabalho foi apresentado à Pós-graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Mestre em Ciência da Computação.

ORIENTADOR(A): Profa. Dra. Flávia de Almeida Barros.

RECIFE
2017

Catálogo na fonte
Bibliotecária Monick Raquel Silvestre da S. Portes, CRB4-1217

A447b Almeida Junior, Oberdan Alves de
Beck: um chatbot baseado na terapia cognitivo-comportamental para apoiar adolescentes com depressão / Oberdan Alves de Almeida Junior. – 2017.
165 f.: il., fig., tab.

Orientadora: Flávia de Almeida Barros.
Dissertação (Mestrado) – Universidade Federal de Pernambuco. CIn, Ciência da Computação, Recife, 2017.
Inclui referências e apêndices.

1. Inteligência artificial. 2. Processamento de linguagem natural. I. Barros, Flávia de Almeida (orientadora). II. Título.

006.3 CDD (23. ed.) UFPE- MEI 2017-196

Oberdan Alves de Almeida Junior

**Beck: Um Chatbot Baseado na Terapia Cognitivo-Comportamental para
Apoiar Adolescentes com Depressão**

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação.

Aprovado em: 04/07/2017

BANCA EXAMINADORA

Profa. Dra. Patricia Cabral de Azevedo Restelli Tedesco
Centro de Informática / UFPE

Prof. Dr. Rafael Ferreira Leite de Mello
Departamento de Estatística e Informática / UFRPE

Profa. Dra. Flávia de Almeida Barros
Centro de Informática / UFPE
(Orientadora)

*Dedico este trabalho a minha família e aos meus amigos que foram essenciais para
superar as dificuldades durante este percurso.*

AGRADECIMENTOS

Primeiramente agradeço a Deus por ter me abençoado com saúde e capacidade para realizar este trabalho.

Aos meus pais Oberdan Alves e Josélia Serafim, por sempre estarem ao meu lado com apoio e com confiança em tudo o que faço.

À minha orientadora, professora Dra. Flávia de Almeida Barros, que me aceitou como aluno e me orientou durante este trabalho. Meus sinceros agradecimentos.

À psicóloga Milena Laíz Braga Lucena que sentou ao meu lado e fez as correções e sugestões nos diálogos criados para o *chatbot*.

Aos meus amigos Nailton Galdino, que sempre me escutou e esteve sempre ao meu lado; Monique Soares, por toda paciência e imensurável ajuda com este trabalho; Alessandro Leandro, pela ajuda prestada durante parte deste trabalho. E, não menos importantes, aos amigos Raisal Brito, Flávio Neves e José Thiago, que foram grandes companheiros durante essa jornada. Dividimos os melhores e os não tão bons momentos dessa fase das nossas vidas.

Agradeço a todos os amigos que tive a oportunidade de conhecer no Centro de Informática da UFPE.

Agradeço também a todos os professores da UFPE pela competência com que transmitiram os conteúdos e ensinamentos.

À CAPES e ao CIn – UFPE pelo apoio financeiro para realização desta pesquisa.

A todos, os meus mais sinceros agradecimentos.

“O futuro dependerá daquilo que fazemos no presente.”
(Mahatma Gandhi)

RESUMO

A depressão é um dos distúrbios psicológicos mais frequentes no mundo. Esse distúrbio causa um grande impacto na qualidade de vida do indivíduo, compromete as relações sociais e familiares, e pode levar ao suicídio. Na adolescência, a depressão é uma das doenças psicológicas mais comuns. Muitos dos problemas emocionais têm início e pico durante esta fase da vida. No Brasil, entre os jovens, o suicídio representa a terceira principal causa de morte. A Terapia Cognitivo-Comportamental (TCC) tem se mostrado bastante eficaz no tratamento da depressão. Esse tipo de terapia também pode ser realizado através do computador, sem o acompanhamento de um terapeuta. Estudos identificaram vantagens importantes na comunicação mediada por computador em comparação com a interação humana face a face, como na promoção do sentimento de anonimato e no aumento da autorrevelação. Contudo, a maioria dos sistemas atuais baseados em TCC não apresenta um grau satisfatório de interatividade. Neste cenário, os *chatbots* são uma alternativa para diminuir essa deficiência dos sistemas de terapia via computador. *Chatbots* são sistemas criados para simular um diálogo real com o usuário, podendo ter a capacidade de analisar e influenciar seus comportamentos. Esse tipo de sistema interage com o usuário através do uso da linguagem natural. Este projeto de mestrado desenvolveu um *chatbot*, baseado nos princípios da TCC, capaz de conversar com adolescentes que sofrem depressão. Para construir o *chatbot* Beck, utilizamos o *ChatScript*, uma linguagem de desenvolvimento desse tipo de sistema, que foi vencedora de vários prêmios. Avaliamos o desempenho e a utilidade de Beck através de um teste de conversação e de um *Survey* com adolescentes. Os resultados obtidos foram muito satisfatórios. De acordo com os resultados, 85,94% dos adolescentes ficaram satisfeitos com o desempenho de Beck, e em relação à sua utilidade, 92,19% dos adolescentes concordam que o *chatbot* foi útil para eles.

Palavras-chave: *Chatbots*. Processamento de Linguagem Natural. Depressão. Terapia Cognitivo-Comportamental. *ChatScript*.

ABSTRACT

Depression is one of the most frequent psychological disorders in the world. This disorder has a major impact on the individual's quality of life, compromises social and family relationships and can lead to suicide. In adolescence, depression is one of the most usual psychological illnesses. Several emotional problems start and peak during this phase of life. In Brazil, for instance, suicide is the third leading cause of death among youngsters. Cognitive-Behavioural Therapy (CBT) has been shown to be effective in treating depression. This type of therapy can also be performed through the computer, without the accompaniment of a therapist. Studies have identified important advantages in computer-mediated communication compared to face-to-face human interaction, such as promoting anonymity and increasing self-disclosure. However, most current CBT-based systems have not shown a satisfactory degree of interactivity. In this scenario, *chatbots* are an alternative to minor this deficiency of computer based therapy systems. *Chatbots* are systems created to simulate a real dialogue with users, being able to analyze and influence their behavior. This type of system interacts with the user through the use of natural language. This research project developed a *chatbot* based on the principles of CBT, whose aim is to dialogue with adolescents who suffer from depression. To build the *chatbot* Beck, we deployed ChatScript, a computer language tailored for the development of this type of system, which has won several awards. We evaluated the performance and usefulness of our *chatbot* Beck through a conversation test and a survey conducted with adolescents. The results obtained are very satisfactory: 85.94% of the adolescents answered that they were satisfied with Beck's performance, and 92.19% of the adolescents agreed that the *chatbot* was useful for them.

Keywords: Chatbots. Natural Language Processing. Depression. Cognitive-Behavioural Therapy. ChatScript.

LISTA DE ILUSTRAÇÕES

Figura 1 – Tela Principal de Beck.	20
Figura 2 – Módulos de Processamento em Sistemas de PLN.	24
Figura 3 – Modelo Básico Funcional dos Agentes Conversacionais.	31
Figura 4 – Arquitetura de Sistemas de PLN.	33
Figura 5 – Trecho de Conversa com ELIZA.	34
Figura 6 – Trecho de Conversa com PARRY.	36
Figura 7 – Trecho de Conversa com JULIA.	37
Figura 8 – Trecho de Conversa com ALICE.	39
Figura 9 – Trecho de Conversa com ROSE.	39
Figura 10 – Exemplo da Estrutura de um Arquivo AIML.	42
Figura 11 – Exemplo do Objeto <i>Graphmaster</i>	42
Figura 12 – Exemplo de Categoria Comum.	43
Figura 13 – Exemplo de Categoria Recursiva.	44
Figura 14 – Exemplo de Uso de Contexto em Artificial Intelligence Markup Language (AIML)	46
Figura 15 – Exemplo Arquivo de Tópico no <i>ChatScript</i>	47
Figura 16 – Modelo Cognitivo.	59
Figura 17 – Ciclo dos Principais Elementos da TCC.	60
Figura 18 – Tela do Sistema MoodGYM.	65
Figura 19 – Tela Inicial <i>Beating the Blues</i>	66
Figura 20 – Tela do Sistema <i>Good Days Ahead</i>	67
Figura 21 – Trecho de Conversa com o <i>chatbot Overcoming Depression</i>	69
Figura 22 – Tela do <i>Chatbot</i> TeenChat.	70
Figura 23 – Exemplo de Tela do Help4Mood.	72
Figura 24 – Exemplo de Interação de Ellie.	73
Figura 25 – Exemplo Tela do <i>Chatbot</i> Wysa Versão IOS.	75
Figura 26 – Tela de Cadastro de Usuário.	81
Figura 27 – Trecho da Apresentação de Beck para o Usuário.	81
Figura 28 – Quadro de Atividades de Diversão.	82
Figura 29 – Pergunta de Beck sobre os Pais do Adolescente.	83
Figura 30 – Fim da Aquisição de Perfil.	84
Figura 31 – Controle da Organização Global da Conversa.	85
Figura 32 – Diálogos Implementados em Beck.	86
Figura 33 – Bloqueio do Campo de Mensagem e Microfone.	86
Figura 34 – Diálogo Sobre a Influência dos Pensamentos.	87

Figura 35 – Quadro do Humor.	87
Figura 36 – Conversa sobre os Sentimentos.	87
Figura 37 – Conversa sobre Estimular Atividades prazerosas.	88
Figura 38 – Os Pensamentos Negativos da Maria, do João e da Joana.	89
Figura 39 – Quadro o que eu sinto com frequência.	89
Figura 40 – Situações Vivenciadas pelas Personagens que Desencadeavam os Pen- samentos Negativos.	90
Figura 41 – Quadro o que eu faço com frequência.	90
Figura 42 – A História da Ana e Margarida.	91
Figura 43 – Miniformulação de Caso do Usuário.	91
Figura 44 – Tratamento de Insultos.	92
Figura 45 – Botão Sair.	93
Figura 46 – Fechamento da Conversação.	93
Figura 47 – Arquitetura Geral do Sistema.	94
Figura 48 – Tratamento do Texto para a Síntese de Voz.	96
Figura 49 – Modelagem do Banco de Dados para o <i>Chatbot</i> Beck.	97
Figura 50 – <i>Script</i> Python para Comunicação com o NLPNET.	98
Figura 51 – Código de Normalização do Texto do Usuário.	99
Figura 52 – Trecho do Código do Hunspell e do Algoritmo de Distância de Levensh- tein.	100
Figura 53 – Regra do Algoritmo RSLP Stemmer.	101
Figura 54 – Exemplo do Uso do NLPNET.	101
Figura 55 – Resposta para o Usuário com um Anexo (OOB).	103
Figura 56 – Fases do Funcionamento do Sistema.	104
Figura 57 – Representação do <i>Chatbot</i> Beck.	105
Figura 58 – Tela Inicial de Beck.	106
Figura 59 – Tela Principal de Beck.	107
Figura 60 – Exemplo de Conversa com Beck.	107
Figura 61 – Teste de Unitário - Início das Perguntas.	108
Figura 62 – Teste de Integração - Recuperação de Informações do Banco no <i>ChatS-</i> <i>cript</i>	109
Figura 63 – Teste de Sistema - Acesso do Usuário ao <i>Chatbot</i>	109
Figura 64 – Tela Inicial de Beck Criada para Realização do Teste.	113

LISTA DE TABELAS

Tabela 1 – Comparação entre o AIML e o <i>ChatScript</i>	53
Tabela 2 – Exemplos de Erros de Pensamento.	58
Tabela 3 – Tabela Comparativa dos Sistemas e dos <i>Chatbots</i>	77
Tabela 4 – Resultado das Questões Realizadas no <i>Survey</i>	115
Tabela 5 – Resultado em Relação ao Desempenho e a Utilidade de Beck.	116
Tabela 6 – Alguns Erros na Interpretação de Perguntas e Respostas Cometidos por Beck.	118
Tabela 7 – Comentários dos Adolescentes no <i>Survey</i> Realizado.	119

LISTA DE ABREVIATURAS E SIGLAS

AC Agente Conversacional.

ACI Agente Conversacional Incorporado.

ACT Agente Conversacional Textual.

AI Agentes Inteligentes.

AIML Artificial Intelligence Markup Language.

CBT Cognitive Behavioural Therapy.

CCBT Computerised Cognitive Behavioural Therapy.

CFP Conselho Federal de Psicologia.

DSS Decision Support System.

EMDR Eye Movement Desensitization and Reprocessing.

IA Inteligência Artificial.

ICBT Internet-delivered Cognitive Behavioural Therapy.

ICT Institute for Creative Technologies.

LTP Language Technology Platform.

NHS National Health Service.

NICE National Institute for Health and Care Excellence.

NLG Natural Language Generation.

NLTK Natural Language Toolkit.

NLU Natural Language Understanding.

OMS Organização Mundial da Saúde.

PDA Personal Digital Assistant.

PLN Processamento de Linguagem Natural.

PMS Personal Monitoring System.

TAC Teoria da Análise da Conversação.

TCC Terapia Cognitivo-Comportamental.

TREC Terapia Racional Emotiva Comportamental.

XML eXtensible Markup Language.

SUMÁRIO

1	INTRODUÇÃO	17
1.1	Motivação e Contexto	18
1.2	Objetivos	18
1.2.1	Objetivo Geral	18
1.2.2	Objetivos Específicos	18
1.3	Trabalho Realizado	19
1.4	Contribuições	20
1.5	Organização do Documento	20
2	CHATBOTS E TEORIAS RELACIONADAS	22
2.1	Processamento de Linguagem Natural	22
2.1.1	Módulos de Processamento em Sistemas de PLN	23
2.1.2	Teoria da Análise da Conversação	26
2.2	Chatbots - Definições e Aplicações	28
2.2.1	Breve Histórico Sobre Chatbots	29
2.2.2	Chatbots como Agentes Inteligentes Conversacionais	30
2.2.3	Gerações dos Chatbots	32
2.3	Tecnologias para o Desenvolvimento de Chatbots	40
2.3.1	AIML	41
2.3.2	<i>ChatScript</i>	45
2.3.3	AIML versus <i>ChatScript</i>	52
2.4	Considerações Finais	54
3	SISTEMAS DE APOIO AO TRATAMENTO DE TRANSTORNOS PSICOLÓGICOS	56
3.1	Terapia Cognitivo-Comportamental	56
3.1.1	Depressão	61
3.1.2	Terapia Cognitivo-Comportamental via Computador	62
3.1.3	Ética e a Psicoterapia Mediada por Computador	63
3.2	Sistemas Computacionais Baseados em TCC	64
3.2.1	MoodGYM	64
3.2.2	Beating the Blues	65
3.2.3	Good Days Ahead	66
3.3	Chatbots Aplicados à Transtornos Psicológicos	68
3.3.1	Overcoming Depression	68

3.3.2	TeenChat	69
3.3.3	Help4Mood	71
3.3.4	Ellie	73
3.3.5	Wysa	74
3.4	Considerações Finais	76
4	BECK: UM CHATBOT PARA APOIAR ADOLESCENTES COM DEPRESSÃO	78
4.1	Descrição Geral do Trabalho Realizado	78
4.2	Dialogando com Beck	79
4.2.1	Objetivos do Diálogo	79
4.2.2	Aquisição do Perfil do Usuário	80
4.2.3	Organização do Diálogo	83
4.3	Arquitetura do Sistema	93
4.3.1	Plataforma <i>Web</i>	94
4.3.2	Banco de Dados	95
4.3.3	<i>ChatScript</i> - Base de Diálogos	96
4.3.4	NLPNET	97
4.4	Funcionamento do Sistema	101
4.4.1	Pré-Processamento	102
4.4.2	Processamento	103
4.4.3	Pós-Processamento	104
4.5	Interface com o Usuário	105
4.6	Testes do Sistema	108
4.7	Considerações Finais	110
5	AVALIAÇÃO E RESULTADOS	111
5.1	Metodologia	111
5.2	Realização do Teste de Conversação e do <i>Survey</i>	112
5.2.1	Resultados	114
5.2.2	Discussão	116
5.3	Considerações Finais	119
6	CONCLUSÃO	120
6.1	Contribuições	121
6.2	Limitações	121
6.3	Trabalhos Futuros	121
	REFERÊNCIAS	123

APÊNDICE A – SCRIPT DE DIÁLOGO	133
APÊNDICE B – INSTALAÇÃO DO AMBIENTE	155
APÊNDICE C – AUTORIZAÇÃO ESCOLA	162

1 INTRODUÇÃO

A depressão é uma doença que atinge muitas pessoas no mundo. Existem vários tipos de tratamentos para esse transtorno psicológico, entre eles podemos citar a psicoterapia e a farmacoterapia (WOLTERS et al., 2013). A Terapia Cognitivo-Comportamental (TCC) é um método de psicoterapia que tem se mostrado efetivo no tratamento da depressão. Essa terapia tem como objetivo a redução e a prevenção dos sintomas da doença através da identificação e da alteração dos pensamentos disfuncionais. A terapia alega que os pensamentos disfuncionais influenciam a forma como nos sentimos e nos comportamos, e que, tendo consciência desses pensamentos, podemos substituí-los por pensamentos mais apropriados (BECK, 2005). A forma como os indivíduos percebem e processam a realidade influenciará a maneira como eles se sentem e se comportam, esse é o princípio fundamental da TCC.

A TCC também pode ser utilizada através do computador. Essa modalidade da terapia tem se mostrado tão eficaz quanto o tratamento presencial (STASIAK et al., 2016). Entretanto, a maioria dos sistemas atuais, baseados em TCC, não apresenta um grau satisfatório de interatividade. Esses sistemas não conquistam o paciente, pois a taxa de abandono do tratamento é alta (EELLS et al., 2015). Neste contexto, os *chatbots* são uma alternativa para diminuir essa deficiência dos sistemas de terapia via computador. *Chatbots* (robôs de conversação) são sistemas de computador projetados para simular uma conversa natural com os usuários (DERYUGINA, 2010). Por permitir uma interação em linguagem natural, muitos usuários se sentem confortáveis com este tipo de *software*. Além de responder e realizar perguntas, o *chatbot* pode conduzir os usuários numa conversa agradável, podendo ter a capacidade de analisar e influenciar seus comportamentos (ABDUL-KADER; WOODS, 2015).

As interfaces em linguagem natural, como os *chatbots*, tem se tornado uma tendência de uso em vários sistemas de computador porque proporcionam a comunicação em sua forma mais natural que é a linguagem humana. Para permitir esse tipo de comunicação, os *chatbots* podem utilizar recursos de Processamento de Linguagem Natural (PLN) (KHANNA et al., 2015). PLN permite a comunicação, em linguagem natural, entre o Homem e o Computador (MALDONADO et al., 2016). Um estudo realizado com alguns usuários mostrou que a maioria deles se sentiu mais à vontade em realizar uma entrevista com uma interface em linguagem natural do que com um humano. Uma possível explicação para este efeito é que as pessoas podem se sentir mais confortáveis em revelar informações pessoais aos computadores do que aos humanos (DEVAULT et al., 2014).

1.1 Motivação e Contexto

A depressão é uma doença que causa um alto impacto na vida familiar e social, além de trazer um grande sofrimento para o doente (LIMA, 1999). Essa doença pode ser tratada, porém menos da metade das pessoas no mundo recebem algum tipo de cuidado. Alguns fatores como a falta de recursos, a ausência de profissionais capacitados e o estigma social associado aos transtornos mentais podem dificultar ou impedir o acesso ao tratamento (BICKMORE et al., 2010).

A adolescência é uma das fases mais desafiadoras da vida. Muitos dos problemas emocionais surgem durante esse período. A depressão é um dos distúrbios psicológicos mais comuns na adolescência, e em sua forma mais aguda ela pode levar ao suicídio (SUDHIR, 2015). No Brasil, a terceira principal causa de morte entre os jovens é o suicídio (PSIQUIATRIA, 2014).

Por outro lado, é possível notar que a maioria dos adolescentes tem facilidade no uso de ferramentas tecnológicas (SPIZZIRRI et al., 2012). Assim, esta é uma alternativa viável para dar apoio a jovens com transtornos de depressão. Existem muitos *softwares* disponíveis no mercado que empregam a TCC via computador, mas a maioria desses sistemas não usa recursos de PLN para aumentar o nível de interatividade com o usuário.

1.2 Objetivos

Esta pesquisa de mestrado desenvolveu um *chatbot* para dar apoio aos adolescentes com depressão. Dividimos os objetivos dessa pesquisa em geral e específicos, listados a seguir:

1.2.1 Objetivo Geral

Desenvolver um *chatbot*, baseado em PLN e utilizando a TCC, para conversar com adolescentes que têm depressão ou que desejam evitá-la.

1.2.2 Objetivos Específicos

- Elaborar diálogos baseados na TCC e estruturá-los de uma forma apropriada a idade dos adolescentes;
- Criar uma plataforma que permita utilizar recursos de PLN no *ChatScript* para realizar a interação com o usuário;
- Disponibilizar o *chatbot* em um *site* da Internet;
- Realizar testes para assegurar um bom funcionamento do *chatbot* com todos os recursos empregados; e
- Avaliar o desempenho e a utilidade do *chatbot* criado.

1.3 Trabalho Realizado

Neste trabalho, desenvolvemos o primeiro *chatbot* na língua portuguesa, batizado de Beck, que usa a TCC para conversar com adolescentes que têm depressão ou que desejam evitá-la. Beck, um *chatbot* feminino, foi criada como um sistema de autoajuda, e busca mostrar aos usuários que os pensamentos negativos que eles têm podem influenciar as suas emoções e os seus comportamentos. Embora Beck se aproprie e utilize os conceitos da TCC para conversar com os adolescentes, é importante enfatizar que esse *chatbot* não faz terapia através da Internet. O propósito de Beck é conversar com adolescentes com depressão, com intuito de ajudá-los.

Por orientação de uma psicóloga, restringimos a faixa etária do nosso público alvo, os adolescentes, para a faixa etária de 15 a 18 anos¹. Os diálogos modelados para o *chatbot*, segundo a psicóloga consultada, estão mais adequados para essa faixa etária da adolescência.

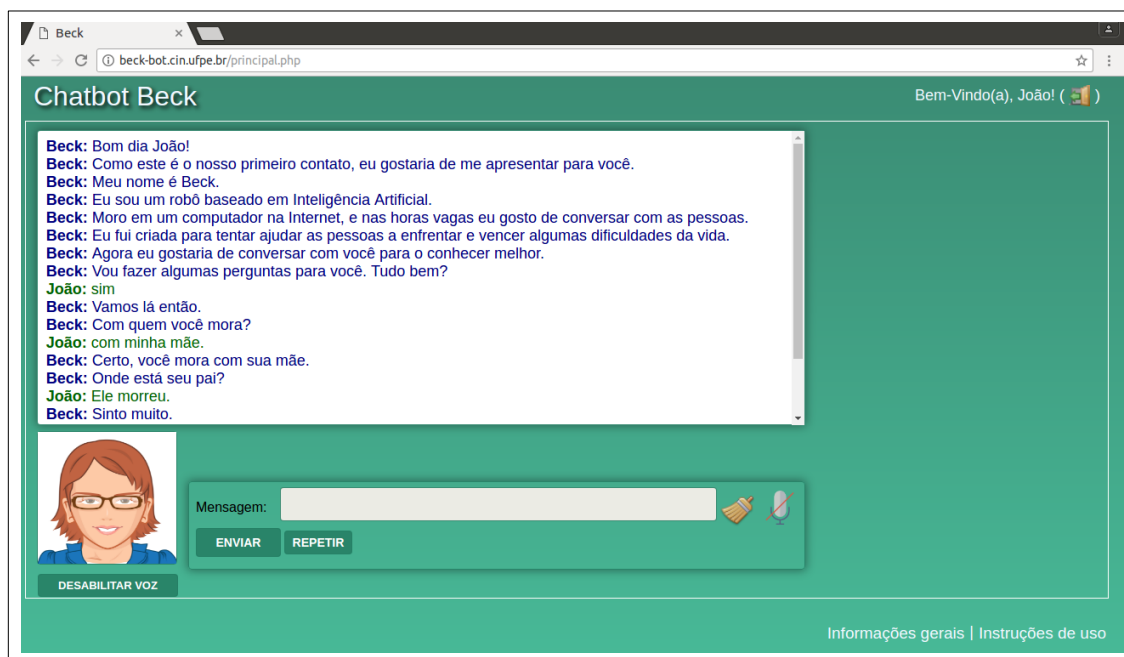
O *chatbot* Beck foi implementado através de um servidor *Web*, e está disponível na Internet visando facilitar o seu acesso. Para o desenvolvimento de Beck, utilizamos o *ChatScript*, uma linguagem de desenvolvimento de *chatbots* baseada nos preceitos do PLN. Essa linguagem foi utilizada na construção de vários *chatbots* vencedores de competições mundiais, como o Prêmio Loebner (LOEBNER, 2015). Isso atesta a qualidade da nossa escolha. No entanto, para possibilitar o uso de alguns recursos de PLN para a língua portuguesa, no *ChatScript*, foram realizadas algumas adaptações na linguagem.

Na Figura 1, podemos ver a tela principal de Beck e todos os recursos disponibilizados através de sua interface.

Para avaliar o desempenho e a utilidade de Beck, realizamos um teste de conversação e um *Survey* com 32 adolescentes de Escolas Públicas do Estado de Pernambuco. Os resultados obtidos foram muito satisfatórios. De acordo com esses resultados, 85,94% dos adolescentes ficaram satisfeitos com o desempenho de Beck, e, em relação à utilidade, Beck teve um resultado de 92,19% de aprovação.

¹ A adolescência compreende o período entre 12 e 18 anos de idade.

Figura 1 – Tela Principal de Beck.



Fonte: Autoria própria.

1.4 Contribuições

- Primeiro *chatbot* para a língua portuguesa baseado na TCC;
- Realização de pesquisa bibliográfica de trabalhos que comparam os recursos entre as linguagem de desenvolvimento de *chatbots* AIML e *ChatScript*;
- Desenvolvimento de uma arquitetura para o *chatbot* que possibilitasse o uso de Corretor Ortográfico, de *Stemming* e de *POS-Tagging* no *ChatScript*, todos esses recursos na língua portuguesa; e
- Construção de uma base de conceitos que representa sinônimos de palavras.

1.5 Organização do Documento

Esta dissertação está estruturada em seis capítulos. Após esta Introdução, mais cinco capítulos descrevem o trabalho realizado, são eles:

- No Capítulo 2, abordamos os principais conceitos sobre os *chatbots* e as teorias relacionadas a este tipo de sistema, as quais foram utilizadas para o desenvolvimento deste trabalho.

- No Capítulo 3, apresentamos a TCC e alguns sistemas e *chatbots* utilizados para apoiar o tratamento de transtornos psicológicos, como a depressão, a ansiedade e o estresse.
- No Capítulo 4, descrevemos todo o processo para a criação do *chatbot* Beck. Esse capítulo aborda desde a estruturação dos diálogos para o *chatbot* até os testes realizados no sistema desenvolvido.
- No Capítulo 5, apresentamos os resultados obtidos de um Teste de Conversação e de um *Survey* realizados com adolescentes.
- Por fim, o Capítulo 6, traz uma breve conclusão deste trabalho e apresenta as contribuições realizadas, as limitações do trabalho, e propõe melhorias para trabalhos futuros.

2 CHATBOTS E TEORIAS RELACIONADAS

Neste capítulo, abordamos os conceitos do Processamento de Linguagem Natural (PLN) e dos *Chatbots*, que serão necessários para uma melhor compreensão deste trabalho. Na seção 2.1, explanamos sobre o PLN que é a área de pesquisa na qual o desenvolvimento dos *chatbots* está fundamentada. Na seção 2.2, apresentamos os *chatbots*, como surgiram, suas derivações e gerações. Na seção 2.3, mostramos algumas tecnologias utilizadas para o desenvolvimento de *chatbots* e exploramos com mais detalhes duas delas, AIML e *ChatScript*. Por fim, a seção 2.4 traz considerações finais sobre os temas vistos neste capítulo.

2.1 Processamento de Linguagem Natural

Processamento de Linguagem Natural (PLN) é uma subárea de pesquisa da Inteligência Artificial (IA) que estuda a comunicação, através da linguagem natural, entre o Homem e o Computador (MALDONADO et al., 2016). A motivação para esse estudo é fundamentada no aperfeiçoamento da interação homem - computador, a qual ainda é motivo de grande preocupação para os pesquisadores, tendo em vista a sua complexidade. De um modo mais formal, PLN pode ser definida como um ramo da IA que tem por objetivo analisar, interpretar ou produzir texto em uma língua natural (e.g., Português, Inglês, Espanhol) (ALLEN, 2003).

PLN possibilita que os seres humanos se comuniquem com os computadores de uma forma mais natural, utilizando sua língua nativa. O seu uso permite que os usuários não precisem aprender uma linguagem artificial para realizar a interação com o computador, a qual a sintaxe costuma ser um pouco mais difícil, como as linguagens de consulta de banco de dados (OLIVEIRA; NAVAUX, 2004).

Para ser considerado um sistema baseado em PLN, ele deve atender a duas condições (OLIVEIRA; NAVAUX, 2004):

- Uma parte do sistema deve ser codificado em linguagem natural;
- O processamento de entrada e/ou saída é baseado em aspectos sintáticos, semânticos e/ou pragmáticos de uma língua natural.

2.1.1 Módulos de Processamento em Sistemas de PLN

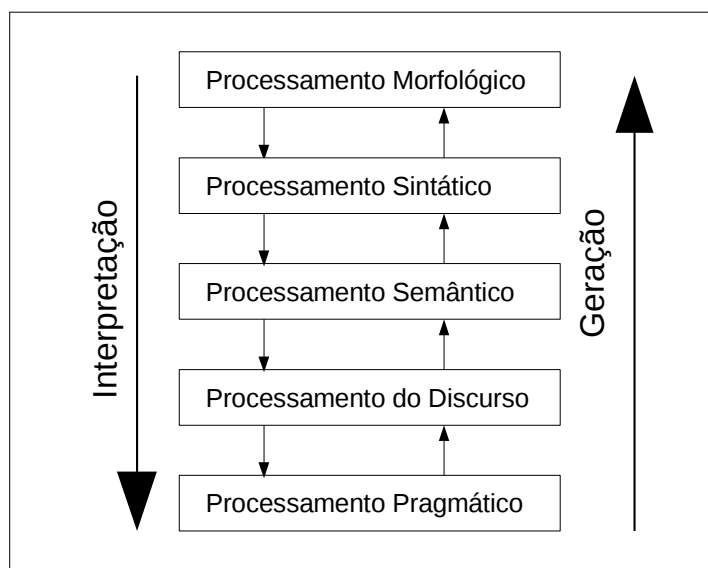
Para que um sistema computacional interprete uma frase e/ou realize a geração de texto em linguagem natural, é necessário que ele execute alguns processos. Por exemplo, processamento morfológico, sintático, semântico, do discurso e/ou pragmático. Geralmente, os sistemas em PLN são modularizados, e cada um desses módulos é responsável por realizar um desses processamentos. Descrevemos a seguir, esses módulos de processamento em PLN (ALLEN, 1995; BARROS; ROBIN, 2001; MÜLLER, 2003; OLIVEIRA; NAVAUX, 2004):

- **Processamento Morfológico** - O processamento morfológico trata as palavras quanto à sua estrutura, forma, flexão e classificação, que correspondem às estruturas estáticas (radicais) e variantes (afixos). Para o desenvolvimento de *chatbots*, o processamento morfológico é importante, pois pode resultar em um número bem menor de regras necessárias para realizar o casamento de padrão (veremos sobre casamento de padrão na seção 2.2). Esse processamento permite reduzir as palavras aos seus radicais, e isso faz com que não precisemos criar uma regra para cada variação da palavra (afixos diferentes). Por exemplo, para realizar o casamento de padrão com a palavra “amigo” e algumas de suas variações (e.g., “amiga”, “amigos” e “amigas”), não é necessário a criação de várias regras para o casamento de padrão, sendo apenas necessário construir a regra com o radical “amig”. O processo para realizar a redução da palavra ao seu radical é conhecido em PLN como *Stemming*. Outra atribuição do processamento morfológico é determinar a classe gramatical das palavras (e.g., substantivo, adjetivo, verbo, advérbio). Esse procedimento é conhecido em PLN como *Parts-of-Speech*, ou *POS-Tagging*.
- **Processamento Sintático** - Esse módulo de processamento se encarrega de verificar as relações formais entre as palavras, ou seja, qual é o papel de cada palavra numa frase e a sua correlação com as demais palavras (e.g., sujeito, predicado, objeto direto, objeto indireto). De uma outra perspectiva, o objetivo do processamento sintático é determinar os componentes estruturais das frases. Esse processo mapeia uma frase na sua estrutura sintática. Essa etapa também é conhecida como *Parsing*.
- **Processamento Semântico** - O processamento semântico atribui o sentido das palavras que foram identificadas pelo processamento sintático. A representação do significado é um grande problema em PLN, pois o sentido das palavras em uma frase pode ser interpretado de forma ambígua. Por exemplo, a palavra banco em “Eu irei ao banco”, “Eu vou sentar no banco” ou “Eu banco o jantar”. É interessante lembrar que não existe uma correspondência direta entre sintaxe e semântica. Uma mesma estrutura sintática pode originar diferentes representações semânticas. Por exemplo, “Achei minha caneta de ouro”. A palavra “ouro” pode se referir a um metal precioso, como também a um objeto estimado.

- **Processamento do Discurso** - Essa etapa tenta identificar qual é a influência de uma ou mais frases na interpretação das frases subsequentes, ou seja, ela extrapola a estrutura da frase que está sendo verificada atualmente pelo processador. O processamento do discurso é essencial para a resolução de pronomes (e.g., eu, você, esta, aquela) e de dêiticos (e.g., aqui, hoje).
- **Processamento Pragmático** - O processamento pragmático verifica as relações entre as mensagens e os sujeitos, ou seja, analisa a intenção de quem fala e o reconhecimento desta intenção pelo ouvinte, levando em consideração o contexto em que o emissor e o ouvinte estão inseridos.

O fluxo do processamento em PLN pode mudar de acordo com a função do sistema, interpretação - geração de texto. Nos sistemas criados para realizar a interpretação em linguagem natural (o processo de interpretação é também conhecido como Compreensão de Linguagem Natural, ou do inglês, *Natural Language Understanding (NLU)*), como os *chatbots*, temos a voz ou texto como entrada e uma resposta, em voz ou texto, como saída. A saída do sistema varia de acordo com a sua função. Em um sistema de banco de dados, por exemplo, a saída pode ser o resultado de uma *query*. Na geração (o processo de geração é também conhecido como Geração de Linguagem Natural, ou do inglês, *Natural Language Generation (NLG)*), o processo se inverte, o gerador de texto recebe modelos de conteúdo como entrada e os objetivos de comunicação. A partir dessas informações, o texto é gerado. É importante informar que nem todos os sistemas baseados em linguagem natural contemplam todos esses módulos (BARROS; ROBIN, 2001; VIEIRA; LIMA, 2001). A Figura 2 ilustra os módulos desse processamento.

Figura 2 – Módulos de Processamento em Sistemas de PLN.



Fonte: Baseado em Barros e Robin (2001).

PLN pode ser empregado em diferentes aplicações. Abaixo citamos algumas delas (VIEIRA; LIMA, 2001):

- Corretores ortográficos e gramaticais - Geralmente os corretores ortográficos e gramaticais são implantados como um módulo de um sistema. Por exemplo, um sistema de edição de texto, como o Microsoft Word¹. O módulo de correção verifica se cada uma das palavras digitadas pelo usuário pertence ao dicionário da língua em uso, e verifica algumas construções gramaticais das sentenças, como as regras de concordância. Os corretores também possuem um outro módulo mais simples, que trata erros de fácil detecção, tais como palavras e símbolos de pontuação repetidos, início de frases com letra minúscula e ausência de pontuação no final da sentença.
- Tradutores automáticos - Os tradutores automáticos são utilizados para realizar a tradução de textos de uma língua natural para outra (e.g., inglês, espanhol, francês). Existem vários sistemas ou *sites* disponíveis na Internet para tradução automática, como o Google Translate² e o Babylon³.
- Interfaces em linguagem natural - É qualquer aplicação que seja capaz de se comunicar com o usuário utilizando uma linguagem natural (voz ou texto), como os Sistemas de Pergunta-Resposta (esses foram os primeiros sistemas a interagir com o usuário utilizando linguagem natural, mais detalhes na seção 2.2) e os *chatbots*.
- Recuperação de informação - É a tarefa de encontrar documentos que são relevantes para o usuário. São aplicadas várias técnicas para realizar a recuperação de documentos, como: indexação, busca, filtragem, organização e tratamento de múltiplas línguas. Existem duas abordagens principais e distintas: a busca por metadados (cabecinhos ou palavras-chaves que descrevem o conteúdo dos documentos) ou por conteúdo.
- Reconhedores e sintetizadores de voz - Avanços nos estudos sobre o reconhecimento de voz (do inglês, *Speech Recognition* ou *Speech to Text* (STT)) tornaram possível o desenvolvimento de sistemas que são capazes de identificar as diversas palavras de uma língua. O reconhecimento de voz pode servir, por exemplo, para ligar para o telefone de alguém usando comando de voz (execução de tarefas); pode ser útil para transcrever a fala em texto em sistemas ditados; ou mesmo interagir com o computador através de um sistema de diálogo (e.g., *chatbots*). Sistemas de síntese de voz (ou do inglês, *Speech Synthesis* ou *Text to Speech* (TTS)) realizam o processo inverso: a partir de um texto escrito, o sistema faz a leitura para o usuário através de uma voz sintetizada. O uso do reconhecimento e síntese de voz é uma tendência

¹ <<https://products.office.com/pt-br/word>>

² <<https://translate.google.com.br/>>

³ <<http://tradutor.babylon-software.com/>>

natural que deve ser adotada largamente como um meio de realizar a interação com o computador. Sistemas como Siri⁴ da Apple e Cortana da Microsoft⁵ são exemplos de sistemas capazes de realizar o reconhecimento e síntese de voz.

A área de PLN tem um caráter interdisciplinar, pois se apropria e se utiliza de conceitos e estudos de outras áreas do conhecimento, como a Linguística e a Psicologia (VIEIRA; LIMA, 2001; OLIVEIRA; NAVAUX, 2004). Dentro da Linguística, existe uma teoria de estudo particularmente interessante para o desenvolvimento de *chatbots*, que é a Teoria da Análise da Conversação (TAC) (MARCUSCHI, 1986). A seguir, veremos alguns conceitos básicos dessa teoria.

2.1.2 Teoria da Análise da Conversação

A TAC estuda a conversação na prática social, que é a mais comum do dia a dia. Embora não percebida, a conversação ocorre de maneira estruturada e coordenada, e pode envolver outros fatores (e.g., entonação da voz, gestos, expressões corporais, olhares), além da fala. Uma conversação se distingue de outros processos comunicativos, como discursos e aulas, por apresentar algumas características básicas. Dentre elas, destacam-se (MARCUSCHI, 1986):

- Interação entre pelo menos dois falantes - Note que, para que haja uma conversação, não é necessário que os falantes estejam face a face, uma vez que pode ocorrer conversações a distância, como acontece com ligações telefônicas ou salas de bate-papo na Internet;
- Ocorrência de pelo menos uma troca de turno de fala - Isso distingue a conversação de outros recursos comunicativos, como os monólogos, o sermão, a conferência, entre outros.
- Presença de uma sequência de ações coordenadas - Toda conversação pode ser vista como uma estrutura organizada, compartilhada em atos previsíveis.
- Realização em uma identidade temporal - A conversação deve ser realizada em uma identidade temporal, ou seja, ela deve acontecer durante o mesmo tempo. Para Marcuschi (1986), um diálogo só pode ser considerado uma conversação se ele for síncrono. Diferentemente como pode acontecer em aplicativos de troca de mensagens, como WhatsApp⁶ ou Messenger⁷, que podem ser assíncronos. Há uma discussão em Recuero (2008) em relação aos diálogos mediados através do computador, com suas diferenças e semelhanças em relação aos diálogos realizados face a face; e

⁴ <<http://www.apple.com/ios/siri/>>

⁵ <<https://www.microsoft.com/en-us/mobile/experiences/cortana/>>

⁶ <https://www.whatsapp.com/?l=pt_br>

⁷ <<https://pt-br.messenger.com/>>

- Envolvimento em uma interação centrada - A interação centrada se distingue de situações onde duas ou mais pessoas estão falando sem, no entanto, estarem tratando de um mesmo tema.

A estrutura da conversação pode ser entendida a partir de organizadores locais e globais. A seguir, veremos esses dois níveis de organização (MARCUSCHI, 1986):

Organização Local

De uma forma geral, a conversação se dá em turnos. Um turno pode ser tido como aquilo que um falante faz ou diz enquanto tem a palavra, incluindo aí a possibilidade do silêncio, quando este for significativo e notado. A conversação consiste normalmente numa série de turnos alternados, que compõem sequências em movimentos coordenados e colaborativos. Entre essas sequências, existem algumas que são padronizadas quanto à sua estrutura. Devido à proximidade e ao tipo de relações, as sequências padronizadas são chamadas de “pares adjacentes”. Como exemplos, podemos citar: pergunta - resposta, ordem - execução, convite - aceitação/recusa, cumprimento - cumprimento. Em relação aos turnos, vale a máxima “fala um de cada vez”, pois as falas simultâneas e as sobreposições de vozes são momentos críticos na organização da conversação, que podem levar o sistema ao colapso. Além das sequências, existem outros elementos na organização local, como as preferências e os tópicos da conversação.

As preferências, termo desenvolvido por Sacks e Schegloff em 1974, se referem às diferentes maneiras de se responder a uma determinada ação no contexto conversacional (SACKS; SCHEGLOFF; JEFFERSON, 1974). Dessa forma, no caso de convites é mais comum a preferência pela aceitação do que pela recusa, no caso de elogios, tudo indica que a opção mais comum é a “despreferência”⁸, ou seja, eles são evitados pelo menos em público. Levinson observa que as preferências e as “despreferências” também levam em conta os aspectos socioculturais e antropológicos de um povo, e não apenas linguísticos (LEVINSON, 2003).

E por fim, temos os tópicos da conversação, que diferentemente do que ocorre no monólogo ou aulas, são negociados pelos interlocutores a cada turno, podendo a qualquer momento ser redirecionados. Geralmente, uma conversa é iniciada com um tópico e, naturalmente, quando a conversa é espontânea e fluente, ele pode ir se modificando, sendo importante observar que só se mantém uma conversa quando se tem algo para falar, ainda que seja algo sem importância.

Organização Global

Além dos organizadores conversacionais que atuam localmente, existem alguns recursos que organizam a conversação em termos globais. Esses recursos são divididos em seções,

⁸ Termo adotado por Marcuschi para caracterizar ações negativas ou discordantes em relação a uma ação anterior.

são elas: abertura, desenvolvimento e fechamento. A abertura é utilizada para um contato inicial, onde normalmente são realizadas as identificações dos falantes, no caso de um telefonema ou quando os falantes não se conhecem, e os cumprimentos. Quando os falantes não se conhecem ou não se veem há algum tempo, é normal que essa seção se estenda por um período de tempo maior, ao contrário do que acontece quando os falantes se veem com frequência, os cumprimentos tendem ser mais rápidos e logo é iniciada a conversação sobre algum tópico.

Na seção de desenvolvimento, a conversa foca em algum tópico que usualmente é de conhecimento comum. Toda conversação é sempre situada em um contexto em que os participantes estão engajados. Geralmente, quem escolhe o tópico é quem inicia a conversação.

Por fim, tem-se a seção de fechamento, que em geral é mais complexa do que a seção de abertura. Normalmente, quem finaliza a conversa é quem a iniciou. Mas isso não é rígido em si. Os fechamentos geralmente acontecem quando se esgota o tópico abordado, e com frequência usam-se marcadores para indicar que se deseja finalizar a conversa, com frases do tipo: “bem, já demorei demais”, “tá bom”, “preciso ir agora”. Esses marcadores podem também ser referenciados como pré-fechamentos. Eles sinalizam para o outro falante que se deseja finalizar a conversa. Entretanto, quando não está claro para um dos falantes que o tópico está concluído, os turnos durante essa conversação podem se estender mais um pouco.

2.2 Chatbots - Definições e Aplicações

Chatbot é um tipo de Agente Conversacional (AC) que interage com o usuário por meio da linguagem natural (ABUSHAWAR; ATWELL, 2015). Esse tipo de sistema simula uma conversa inteligente com o usuário, e tem o objetivo de mantê-la de forma natural e coerente. A interação com o computador usando linguagem natural não é novidade. Em 1960, temos registros dos primeiros sistemas que buscavam interagir com o usuário em linguagem natural. Esses sistemas são conhecidos como Sistemas de Pergunta-Resposta (MCTEAR, 2004). As pesquisas realizadas na área de IA e PLN possibilitaram um grande avanço desde então, assim, tornando os ACs uma das abordagens mais proeminentes de desenvolvimento.

Os *chatbots* (ou *chatterbots* - robôs de diálogo, ou simplesmente *bots* - robôs) são criados para vários propósitos, como exemplos, temos: assistentes virtuais, agentes tutores e companheiros virtuais. E são empregados em diferentes domínios, exemplos: comércio, entretenimento, educação e saúde (MARTÍNEZ-MIRANDA, 2010). No site [chatbots.org](https://www.chatbots.org/)⁹ estão disponíveis vários exemplos de *chatbots* e onde eles podem ser aplicados.

⁹ <<https://www.chatbots.org/>>

Atualmente, os *chatbots* estão recebendo uma grande atenção. Boa parte dessa atenção é devida a recentes pronunciamentos das empresas Microsoft e Facebook que, em 2016, declararam que os *chatbots* mudarão a forma como interagimos com os aplicativos (DALE, 2016). Vários *chatbots* estão sendo incorporados a programas populares de troca de mensagens como o Messenger¹⁰ e Telegram¹¹. Os assistentes virtuais, uma forma derivada dos *chatbots*, como exemplos, citamos: Alexa da Amazon¹², Cortana da Microsoft, Siri da Apple, Google Now¹³ como representantes de *bots* que estão promovendo essa revolução (YAN et al., 2016).

2.2.1 Breve Histórico Sobre Chatbots

Alan M. Turing, em seu artigo *Computing Machinery and Intelligence* (TURING, 1950), propõe considerar a seguinte questão: “As máquinas podem pensar?”. Para melhor refletir sobre esse questionamento, Turing sugere um jogo chamado *The Imitation Game* (Jogo da Imitação). Posteriormente, esse jogo ficou conhecido como o Teste de Turing.

O trabalho de Turing possibilitou o surgimento de máquinas capazes de dialogar com o ser humano usando linguagem natural. Os primeiros sistemas que interagem com o usuário em linguagem natural datam do início da década de 1960. Esses sistemas são intitulados de Sistemas de Pergunta-Resposta. Eles recebem uma pergunta em linguagem natural e, por meio de pesquisa em uma base de dados que contém pares Pergunta-Resposta, retornam uma resposta (RABELO; BARROS, 2004). Exemplos de tais sistemas são BASEBALL (JR et al., 1961), STUDENT (BOBROW, 1964) e SHRDLU (WINOGRAD, 1972).

Outro tipo de sistema que também surgiu a partir do trabalho de Turing foi o *chatterbot*. ELIZA, criada por Weizenbaum em 1966, foi o primeiro sistema desse tipo registrado na literatura (WEIZENBAUM, 1966). O termo *chatterbot* foi introduzido apenas em 1994 por Michael L. Mauldin (MAULDIN, 1994). Em seu trabalho, Mauldin descreve a criação e atuação dos *chatterbots* no jogo TINYMUD, no qual eles se faziam passar por jogadores reais.

Em novembro de 1991, o primeiro Teste de Turing foi realizado em grande escala no *Boston Computer Museum*, nos Estados Unidos. Esse evento é conhecido como *The Loebner Prize*¹⁴. O evento, que recebe o nome de um dos seus fundadores, Dr. Hugh Loebner, acontece anualmente desde o seu início. Ele tem o objetivo de reconhecer e premiar o *chatbot* considerado mais “humano” pelos juízes. Para vencer este concurso, o *chatbot* deve conseguir enganar metade dos juízes (o Teste de Turing original fala somente em enganar mais de 30% dos juízes). Desde a instituição do concurso, nenhum *chatbot*

¹⁰ <<https://messengerplatform.fb.com/>>

¹¹ <<https://telegram.org/blog/bot-revolution>>

¹² <<https://developer.amazon.com/public/solutions/alexa>>

¹³ <<https://www.google.com/landing/now/>>

¹⁴ <<http://www.loebner.net/Prizef/loebner-prize.html>>

conseguiu a proeza de enganar a metade dos juízes, restando apenas premiar o *chatbot* que apresentou o melhor desempenho, de acordo com a pontuação dada pelos avaliadores (LOEBNER, 2015).

Em 2014, foi organizado por Kevin Warwick, da Universidade de Reading (Reino Unido), uma competição especial para celebrar os 60 anos da morte de Turing. Nesse evento, o *chatbot* Eugene Goostman¹⁵ foi premiado e reconhecido como sendo o primeiro *chatbot* a passar no Teste de Turing. Após enganar 33% dos juízes, Warwick proclamou que Eugene Goostman passou no Teste. O modo como essa competição se sucedeu foi alvo de várias críticas, e muitos pesquisadores até hoje não aceitam o resultado de que esse *chatbot* passou no Teste de Turing (YOU, 2015; PEREIRA et al., 2016).

2.2.2 Chatbots como Agentes Inteligentes Conversacionais

De acordo com Russell e Norvig (2013), um agente é uma entidade que pode perceber o ambiente por meio de sensores e agir sobre ele através de atuadores. Definido dessa forma, um agente pode ser uma entidade humana ou artificial (e.g., robôs, *softwares*). Um agente humano percebe o ambiente através dos seus órgãos (e.g., olhos, ouvidos) e age sobre ele através dos seus atuadores (e.g., mãos, pernas). Um agente robô pode ter câmeras e/ou infravermelho como sensores, e motores como atuadores. Já os agentes de *software*, também conhecidos como agentes virtuais, podem perceber o seu ambiente através de uma sequência de teclas para digitação, através de arquivos e pacotes de redes, ou mesmo por voz (reconhecimento de voz), e pode atuar no ambiente através de um texto exibido na tela, escrevendo algo em um arquivo ou mesmo falando (síntese de voz).

Para um agente de *software* ser considerado inteligente, ele deve conseguir agir de forma autônoma em um ambiente, a fim de cumprir seus objetivos de forma satisfatória. Os *chatbots* são projetados para perceber a entrada do usuário (percepção) através de uma interface (ambiente) e oferecer uma resposta adequada (ação), buscando manter uma diálogo coerente com o usuário. Devido a essas características, os *chatbots* podem ser considerados como Agentes Inteligentes (AI). Esses agentes têm como característica principal a capacidade de conversar com os usuários, sendo então chamados de Agentes Inteligentes Conversacionais (ou simplesmente Agentes Conversacionais (ACs)) (BARROS; TEDESCO, 2016). O diálogo foi uma característica proposta como um meio de testar os primeiros conceitos de IA, que surgiu através do trabalho de Turing (TURING, 1950).

Alguns Tipos de Agentes Conversacionais

Desde o aparecimento dos primeiros *chatbots* surgiram variadas derivações desse tipo de sistema. O *site* chatbots.org, um dos maiores *sites* do mundo sobre *chatbots*, lista mais de 160 sinônimos diferentes para designar esses sistemas. Muitos dos sinônimos empregados

¹⁵ <http://pt.wikipedia.org/wiki/Eugene_Goostman>

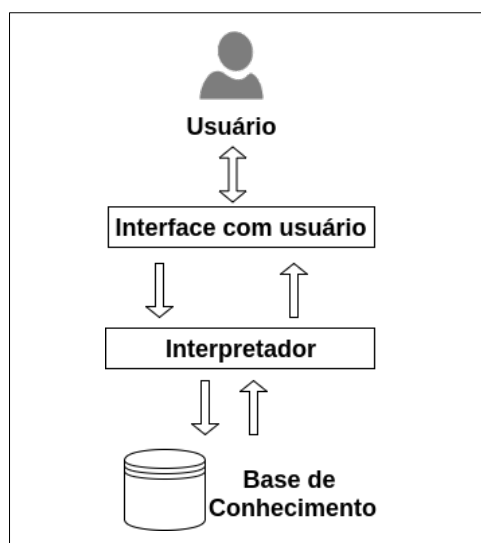
não só denominam termos diferentes com o mesmo sentido, mas também formas distintas de tecnologias utilizadas em seu desenvolvimento, domínio onde irão atuar, tarefas que eles podem executar, recursos e capacidades que o *chatbots* possuem. Alguns podem até apresentar formas e comportamentos humanos. Grande parte dessas derivações se deve à evolução tecnológica que tivemos nas áreas de IA e PLN desde o surgimento dos *chatbots*.

Embora existam variadas derivações dos ACs, uma característica é comum a todos eles, o seu modelo básico funcional. Esse modelo é constituído basicamente de três elementos:

1. **Uma interface com o usuário** - utilizada como meio do usuário se comunicar com o agente;
2. **Um interpretador** - Tem a função de traduzir as sentenças dos usuários para uma forma que possa ser entendida pelo agente e;
3. **Uma base de conhecimento** - Contém todas as regras cadastradas que serão utilizadas para interpretar as sentenças dos usuários.

A Figura 3 ilustra o modelo básico funcional.

Figura 3 – Modelo Básico Funcional dos Agentes Conversacionais.



Fonte: Baseado em Silva e Costa (2007).

Existem duas correntes principais no desenvolvimento de ACs identificadas em O'Shea, Bandar e Crockett (2010). A primeira é o Agente Conversacional Textual (ACT), que aqui é representado pelo *chatbot*; e a segunda é o Agente Conversacional Incorporado (ACI), que pode apresentar um corpo humanoide animado. Além dessas características apresentadas por esses agentes, existem várias outras, como: processamento de voz, expressões faciais, gestos corporais, demonstração de emoções e personalidade. Há muitas taxonomias para designar e diferenciar os ACs, e essas classificações podem envolver uma ou

mais dessas características. Em Barros e Tedesco (2016), são definidos alguns tipos de ACs.

- **Agentes conversacionais** - Agentes com capacidade de se comunicar através de texto ou usando processamento de voz.
- **Agentes conversacionais incorporados** - Agentes com representação física (usualmente com imagens em 2 ou 3 dimensões).
- **Agentes conversacionais incorporados credíveis (ou Humanos Virtuais)** - São agentes que adicionalmente podem expressar emoção e personalidade. Apresentam ainda movimentos corporais e gestos.

Um tipo de derivação de *chatbot* que vem ganhando muita atenção atualmente são os **Assistentes Pessoais**. Eles também são conhecidos como Assistentes Pessoais Digitais (do inglês, *Personal Digital Assistant* (PDA)). Esses assistentes são sistemas que usam como entrada a voz e/ou informações contextuais do usuário para responder perguntas, fazer recomendações ou realizar alguma ação (*e.g.*, agendar reunião, lembrar algum compromisso, pesquisar algo na Internet) através do uso de PLN (CANBEK; MUTLU, 2016; SARIKAYA, 2017). A aplicação mais comum dos assistentes pessoais é a filtragem de informações na Internet, ou apoiar os usuários através de sistemas de recomendação (GARRIDO; MARTINEZ; GUETL, 2010).

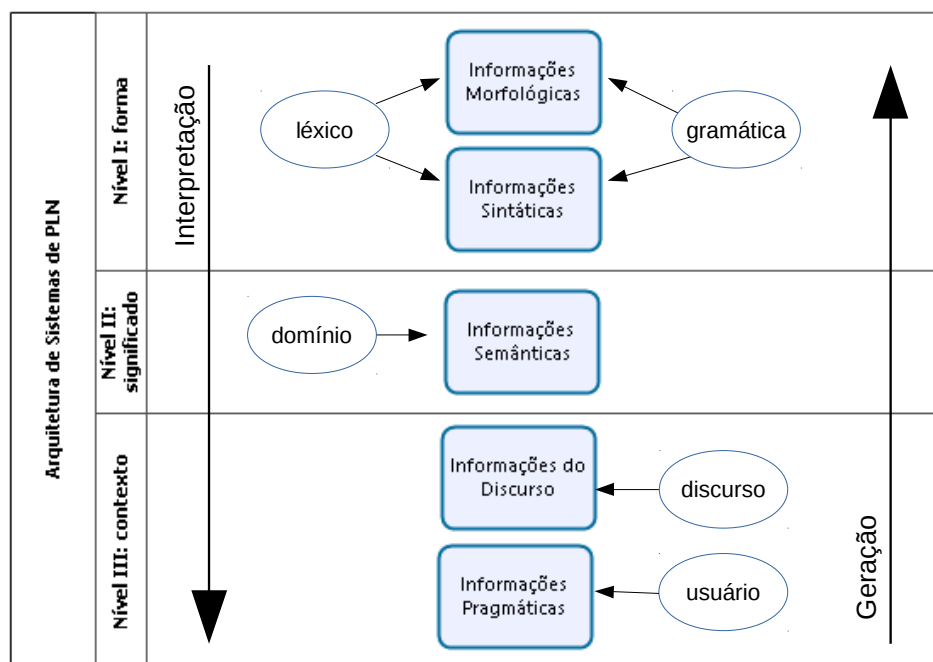
Nos últimos cinco anos houve um grande investimento nos assistentes pessoais. Siri, Google Now, Cortana e Alexa, são os mais importantes assistentes pessoais no mercado atualmente (SARIKAYA, 2017).

Não podemos afirmar que os Assistentes Pessoais são um tipo de AC. De acordo com Wilcox (2017a), os assistentes pessoais como Siri, Cortana e Alexa são somente a “ponta do *iceberg*”, eles são incapazes de conduzir uma conversa. Wilcox os define como um *bot* de consulta de informações.

2.2.3 Gerações dos Chatbots

Em Neves (2005), são identificadas três gerações de *chatbots*, de acordo com a arquitetura apresentada do ponto de vista de PLN, como também das técnicas empregadas em seu desenvolvimento. A arquitetura é organizada em três níveis de tratamento da informação. No nível da **forma**, são tratadas as relações estruturais das palavras (morfologia) e das sentenças (sintaxe) a partir das bases de regras e fatos (o léxico e a gramática); no nível do **significado**, são tratadas as relações entre as sentenças e o conhecimento do domínio (semântica), obtidas do modelo do domínio; e por fim, no nível do **contexto**, são tratadas questões relacionadas ao contexto em que é realizada a interação com o usuário, informações pragmáticas e discursivas, obtidas dos modelos do discurso e do usuário. A Figura 4 ilustra essa arquitetura.

Figura 4 – Arquitetura de Sistemas de PLN.



Fonte: (NEVES, 2005).

É importante fazer uma distinção entre a arquitetura de PLN adotada em Neves (2005) e os módulos de processamento de PLN vistos na seção 2.1. Para o autor, a arquitetura de PLN é utilizada para descrever módulos de tratamento de informações que os *chatbots* possuem para realizar a interpretação das frases dos usuários e a geração de uma resposta. Esses módulos de tratamento de informações representam os processos de PLN. Por exemplo, o *chatbot* ELIZA possui uma tabela de pares de padrões e de modelos de réplicas que é utilizada para tratar as frases dos usuários, a fim de realizar o casamento de padrão e a geração de uma resposta adequada. Essa tabela, para o autor, representa um tratamento de informação sintática.

A seguir, descreveremos cada uma das gerações de *chatbots* identificadas em Neves (2005) e os seus principais representantes.

Primeira Geração

A primeira geração dos *chatbots* é marcada pelo uso de técnicas simples de casamento de padrão e regras gramaticais. Muitos dos *chatbots* criados até hoje se baseiam nessas técnicas (BRADEŠKO; MLADENIĆ, 2012; ABDUL-KADER; WOODS, 2015). O objetivo dessa abordagem é encontrar o melhor casamento de padrão possível entre a frase do usuário e os padrões armazenados na base de diálogos do *chatbot* (ABUSHAWAR; ATWELL, 2015). ELIZA (WEIZENBAUM, 1966) e PARRY (COLBY, 1975) são os principais *chatbots* dessa

geração.

ELIZA

O primeiro *chatbot* registrado na literatura foi ELIZA (WEIZENBAUM, 1966). Esse *chatbot* demonstrou como um mecanismo simples pode imitar um comportamento aparentemente inteligente. Ela foi criada para simular uma psicoterapeuta.

Esse *chatbot* foi projetado para utilizar as técnicas terapêuticas de Carl Rogers (ROGERS, 1966). A abordagem Rogeriana centraliza a terapia no paciente, fazendo com que ele fale sobre seus sentimentos. Quando o paciente faz perguntas, o psicoterapeuta responde-as com outras perguntas. Assim, o paciente é estimulado a sempre falar. ELIZA funciona, em grande parte, parafraseando o que o usuário diz, ou seja, ela extrai palavras do que o usuário forneceu como entrada e as utiliza na composição de uma nova pergunta. A Figura 5 apresenta um trecho de um diálogo realizado com ELIZA, traduzido para o português. Esse *chatbot* está até hoje disponível para conversas na Internet^{16,17}.

Figura 5 – Trecho de Conversa com ELIZA.

ELIZA: Como você vai? Por favor, me diga o seu problema.
VOCÊ: Eu estou um pouco triste.
ELIZA: Você acha que vir aqui vai ajudá-lo a não se sentir triste?
VOCÊ: Sim.
ELIZA: Eu entendo.
VOCÊ: Você poderia me ajudar?
ELIZA: Suponha que eu ajude você - o que isso significaria?
VOCÊ: Eu acho que me sentiria melhor.
ELIZA: Você diz que se sentiria melhor?

Fonte: Autoria Própria.

O *chatbot* ELIZA tem algumas limitações. ELIZA não possui memória, ou seja, ela não mantém um histórico das conversas realizadas com os usuários. Desta forma, um usuário pode repetir um trecho do diálogo e ELIZA responderá naturalmente, como se estivesse analisando aquela sentença pela primeira vez. Por exemplo: o usuário pode fazer um cumprimento para ELIZA ao iniciar uma conversa, algo como: *Olá, tudo bem?*, e no meio da conversa repetir este mesmo cumprimento, e ELIZA responderá da mesma forma, como se ainda não tivesse sido cumprimentada. Outro problema de ELIZA é que algumas respostas dadas por ela são confusas. Por exemplo: o usuário poderia dizer “Eu estou bem obrigado”, e ela responder algo como “Há quanto tempo você está bem obrigado?”.

Embora com limitações, ELIZA conseguiu iludir muitos usuários. O *chatbot* THE PC THERAPIST (WEINTRAUB, 2017), baseado na arquitetura de ELIZA, conquistou quatro

¹⁶ <<http://www.masswerk.at/elizabot/>>

¹⁷ <<http://cyberpsych.org/eliza/>>

prêmios Loebner logo que a competição foi instituída.

Processamento de Linguagem Natural em ELIZA

ELIZA possui uma tabela de padrões e modelos de réplicas que é usada para realizar o casamento de padrão com a frase do usuário e devolver a resposta mais adequada. Essa tabela representa o tratamento de informações sintáticas. Outra tabela usada por ELIZA é a de pronomes. Essa tabela é utilizada para substituir os pronomes na frase do usuário. Exemplo: se houver na frase do usuário o pronome “você”, então esse pronome será substituído para “eu”, independente de qualquer contexto ou análise. A tabela de pronomes representa um tratamento de informações no nível do discurso da arquitetura de PLN.

PARRY

Outro sistema de sucesso que usa técnicas de casamento de padrão parecidas com que ELIZA utilizava é o PARRY (COLBY, 1975). Esse *chatbot* foi criado para simular o comportamento paranoico de uma pessoa com esquizofrenia. Ao contrário de ELIZA, que fazia passar por psicoterapeuta, PARRY foi criado para que as pessoas conversassem com um paciente paranoico.

Colby, criador de PARRY, elaborou a sua própria versão do *Imitation Game* para validar a sua criação. Ele convidou psiquiatras para conversar com PARRY e com três pacientes humanos diagnosticados como paranoicos. O resultado do teste comprovou o bom desempenho de PARRY, uma vez que os psiquiatras não conseguiram distinguir quem eram os humanos e quem era o sistema. Entretanto, é possível que PARRY tenha conseguido enganar os juízes porque simular um comportamento paranóico não deve seguir uma certa lógica, mas mesmo assim não é trivial (MAULDIN, 1994). Para Mauldin (1994), PARRY é um avanço sobre ELIZA porque ele possui personalidade. Existe uma versão de PARRY no site botlibre¹⁸.

PARRY tem implementado em seu mecanismo uma representação de estado emocional, e reproduz esse estado através de alguns comportamentos, como medo, raiva e desconfiança. PARRY possui variáveis para guardar cada emoção que ele pode expressar. Essas variáveis são incrementadas ou decrementadas de acordo com o texto digitado pelo usuário. As respostas de PARRY são selecionadas de acordo com os valores dessas variáveis. Na Figura 6 temos um trecho da conversa com PARRY traduzida para o português.

PARRY também apresenta algumas limitações. Ele tem problemas em determinar a raiz das palavras, o corretor ortográfico não é eficiente e há um problema clássico com

¹⁸ <<https://www.botlibre.com/browse?id=857177>>

Figura 6 – Trecho de Conversa com PARRY.

ENTREVISTADOR: Por que você está no hospital?
PARRY: Eu não deveria estar aqui.
ENTREVISTADOR: Quem o trouxe aqui?
PARRY: A polícia.
ENTREVISTADOR: Que problemas você teve com a polícia?
PARRY: Os policiais não fazem o seu trabalho.
ENTREVISTADOR: O que eles fazem em vez disso?
PARRY: Os policiais prendem as pessoas erradas.
[...]
ENTREVISTADOR: Você está conectado com a mafia?
PARRY: Você sabe, eles me conhecem.
ENTREVISTADOR: Como eles te conhecem?
PARRY: Eles me espionam.
ENTREVISTADOR: Por que eles fariam isso?
PARRY: Eles querem obter provas falsas para me enquadrar.

Fonte: Traduzido de Angeli (2005).

anáforas (PARKINSON; COLBY; FAUGHT, 1977).

Processamento de Linguagem Natural em PARRY

A maioria das etapas contida nos estágios do *Parser* de PARRY realizam alterações na entrada do usuário baseada na morfologia e na sintaxe das frases. PARRY também faz a transformação direta de pronomes, como ELIZA faz. Assim, podemos concluir que PARRY é capaz de realizar o tratamento de informações no nível morfológico, sintático e discursivo da arquitetura de PLN.

Segunda Geração

A segunda geração de *chatbots*, que surgiu no começo década de 1990, passou a incorporar técnicas de IA mais sofisticadas em seu desenvolvimento, como as redes neurais artificiais, devido ao avanço nas áreas de IA, PLN e da Engenharia de software. Alguns sistemas dessa geração passaram a adotar bases de conhecimento com domínio mais específico, e também passaram a guardar informações dos seus usuários (NEVES; BARROS; HODGES, 2006). Como exemplo dessa geração, temos o *chatbot* JULIA (MAULDIN, 1994).

JULIA

O *chatbot* JULIA foi criado para atuar em um TYNMUD, um jogo que simulava um mundo textual onde várias pessoas se entretinham interagindo umas com as outras através de uma sala de bate-papo. Para Mauldin (1994), o TINUMUD era uma ótima oportunidade para avaliar a sua criação, pois JULIA poderia interagir com vários juízes em potencial (os jogadores do *game* que conversavam com JULIA sem saber que ela era um

robô). JULIA tinha o papel de responder às perguntas dos jogadores sobre outros jogadores, salas, objetos e caminhos mais curtos para se deslocar no jogo. Nesse jogo, havia também outros *bots*. Se um jogador se deslocasse até uma determinada área do jogo, poderia até interagir com ELIZA.

JULIA, como PARRY, também tem personalidade. Ela assume um papel feminino e, num determinado período do mês, ela afirma estar com tensão pré-menstrual e, inclusive, seu humor podia ser alterado (FONER, 1997). A Figura 7 mostra um trecho de uma conversa com JULIA, traduzida para o português, onde o usuário pergunta sobre sua aparência.

Figura 7 – Trecho de Conversa com JULIA.

VOCÊ: Julia, quantos anos você tem?
JULIA: Eu tenho 21 anos.
VOCÊ: Julia, fale um pouco sobre você.
JULIA: Eu tenho 1,55m de altura e peso 55kg. Tenho cabelos loiros e curtos e olhos castanho-escuros.
VOCÊ: Julia, qual é a cor dos seus olhos?
JULIA: Eu tenho olhos castanho-escuros.

Fonte: Traduzido de Foner (1997).

As habilidades conversacionais de JULIA foram inicialmente implementadas com regras simples, usando *if-then-else*, casamento de padrão e atribuição de variáveis. A maioria dos padrões existentes na base de conhecimento de JULIA tinha múltiplas respostas associadas, que eram mostradas para o usuário de forma randômica e não repetida. Numa versão mais sofisticada, o módulo de conversação de JULIA foi implementado contendo cinco camadas de priorização, cada uma abrangendo um conjunto de padrões.

Uma dessas camadas de JULIA é chamada de **Rede de ativação**. É nessa camada onde a escolha do tópico da conversa com o usuário é realizada pelo sistema. Essa escolha é feita a partir do que foi escrito pelo usuário na frase atual.

JULIA tem algumas limitações: repetição de algumas perguntas após um tempo decorrido; o domínio sobre o que JULIA podia falar era pequeno, assim como seu vocabulário; como JULIA tinha um conhecimento específico sobre o domínio onde ela atuava, os usuários acabavam reconhecendo que ela era um robô, por não conseguir responder a perguntas fora desse domínio específico (FONER, 1997).

Processamento de Linguagem Natural em JULIA

O *chatbot* JULIA, realiza algumas normalizações no texto de entrada do usuário. As normalizações são tanto para corrigir a ortografia, como também padronizar a entrada do usuário. Por exemplo: *Yu* para *You* (Você), *What's* para *What is* (O que é), isso representa um tratamento de informações morfológicas (NEVES; BARROS; HODGES, 2006).

Além disso, JULIA é capaz de guardar um modelo de informações simples dos usuários, como: nome, gênero e cidade, representando o tratamento de informações pragmáticas. O recurso chamado de redes de ativação de JULIA, que faz com que esse *chatbot* tenha conversas mais coerentes com o usuário. Isso equivale ao tratamento de informações semânticas.

Terceira Geração

A terceira geração é ainda marcada pelo uso da técnica de casamento de padrão, porém de uma maneira mais complexa do que a primeira. Essa geração começou no final da década de 1990 e podemos destacar duas tecnologias que podem ser enquadradas na terceira geração, seguindo os padrões adotados por Neves (2005). A primeira e mais difundida é *Artificial Intelligence Markup Language* (AIML)¹⁹ que é derivada da linguagem de marcação *eXtensible Markup Language* (XML)²⁰. A segunda é o *ChatScript*²¹, linguagem de desenvolvimento de *chatbots* baseada em *script*²². Bruce Wilcox, o criador do *ChatScript*, considera sua linguagem como a próxima geração de linguagem de desenvolvimento de *chatbots*. Como os *chatbots* representantes da terceira geração podemos destacar ALICE (WALLACE, 2009) e ROSE (WILCOX, 2017a).

ALICE

A.L.I.C.E - Acrônimo para *Artificial Linguistic Internet Computer Entity*, é um *chatbot* de código aberto, criado inicialmente por Wallace em 1995, e mais tarde recebeu contribuições de vários desenvolvedores espalhados pelo mundo (WALLACE, 2009). ALICE é o primeiro *chatbot* escrito na linguagem AIML. Ela é um dos *chatbots* mais conhecidos e premiados do mundo. ALICE já ganhou o prêmio Loebner três vezes, nas competições dos anos 2000, 2001 e 2004.

A versão livre de ALICE no ano de 2015 continha, em sua base de conhecimento, mais de 50.000 categorias cadastradas sobre diferentes temas (ABUSHAWAR; ATWELL, 2015). Devido ao seu grande sucesso, ALICE foi a responsável pela difusão do AIML, tendo influenciado a criação de centenas de *chatbots* em diferentes línguas para as mais variadas aplicações. Na Figura 8 podemos ver o trecho de uma conversa, traduzida para o português, de ALICE quando ela participou e ganhou o prêmio Loebner em 2004.

Em AbuShawar e Atwell (2015), é feita uma crítica ao *chatbot* ALICE (WALLACE, 2009) em relação ao *chatbot* AskJeeves (COFFMAN; MCGLAMERY, 2000) (atualmente o AskJeeves é conhecido como ask.com, e não funciona mais como um *chatbot*, e sim como um engenho de busca). O autor relata que o AskJeeves utiliza técnicas sofisticadas de PLN, tais como processamento morfológico, sintático e semântico, enquanto ALICE depende de

¹⁹ <<http://www.alicebot.org/aiml.html>>

²⁰ <<https://www.w3.org/XML/>>

²¹ <<https://sourceforge.net/projects/chatscript/>>

²² <https://en.wikipedia.org/wiki/Scripting_language>

Figura 8 – Trecho de Conversa com ALICE.

ALICE: Olá, meu nome é Alice e eu sou humana.
JUIZ: Olá Alice, como você está?
ALICE: Olá. Tudo está indo muito bem.
JUIZ: Estou contente de estar aqui. Onde você mora?
ALICE: Obrigada pela informação. Estou localizada em Nova York. Onde você está?
JUIZ: Em Nova York. Como você chegou aqui hoje?
ALICE: Conte-me um pouco sobre sua vida em Nova York. Ah! Tudo que eu preciso saber eu aprendi com meu instrutor.

Fonte: Traduzido de Prize (2004).

um número muito grande de regras para realizar o casamento de padrão com a frase do usuário. Para o autor, ALICE privilegia o tamanho da base de regras ao invés da sofisticação de PLN. ALICE compensa a falta dos processamentos morfológico, sintático e semântico, por ter uma grande base de regras.

Na seção 2.3, apresentaremos mais detalhes sobre a linguagem AIML utilizada pelo *chatbot* ALICE, e mostraremos os processos de PLN que podem ser realizados através dessa linguagem.

ROSE

O *chatbot* ROSE, criado através da linguagem *ChatScript*, também é um *chatbot* bastante premiado. Ele foi vencedor do prêmio Loebner nas competições dos anos 2014 e 2015, e obteve bons resultados nos anos de 2013 e 2016, onde conquistou o terceiro lugar (LOEBNER, 2015). ROSE assume a personalidade de uma jovem que trabalha e tem um bom salário. Às vezes, ela pode apresentar um comportamento reservado em relação a alguns assuntos, pois seu trabalho fez com que ela aprendesse o quanto todos nós estamos sendo observados (WILCOX, 2017a). A Figura 9 mostra um trecho da conversa de ROSE quando ela foi vencedora do prêmio Loebner em 2015.

Figura 9 – Trecho de Conversa com ROSE.

JUIZ: Oi, eu sou Andrew. Quem é Você?
ROSE: Sim, eu já sei que o seu nome é Andrew.
JUIZ: Como você está hoje?
ROSE: Bem, obrigada. Como você está?
JUIZ: Você é humana?
ROSE: Claro que sou humana. Não somos todos?
JUIZ: Você pode responder a essa pergunta?

Fonte: Traduzido de Prize (2015).

Outros *chatbots* antecessores a ROSE, criados através do *ChatScript*, também obtive-

ram excelentes resultados. Suzette ganhou o prêmio Loebner em 2010 e Rosette ganhou em 2011 (BRADEŠKO; MLADENIĆ, 2012). O *chatbot* Angela ficou em segundo lugar no ano de 2012 devido, segundo o criador Bruce Wilcox, a um erro introduzido no protocolo da competição Loebner, mas não em seu mecanismo (WILCOX, 2017b). Também em 2012, Angela ficou em terceiro lugar na competição ChatterbotBattles e ganhou o prêmio, nessa mesma competição, dos melhores 15 minutos de conversa (WORSWICK, 2012).

Na seção 2.3, veremos mais informações sobre a linguagem *ChatScript* utilizada pelo *chatbot* ROSE, e os processos de PLN que a linguagem oferece.

Já estamos numa nova geração de *chatbots*? Essa é uma pergunta interessante, tendo em vista que a terceira geração de *chatbots* teve início na década de 90. Alguns pesquisadores consideram duas abordagens para o desenvolvimento de *chatbot*, uma baseada em regras e a outra baseada em Aprendizagem de Máquina (do inglês, *Machine Learning*) (SARIKAYA, 2017; MADHU et al., 2017; MCTEAR; CALLEJAS; GRIOL, 2016c; PATIL et al., 2017). Uma resposta factível para essa pergunta é: depende do propósito para que o *chatbot* será usado. A Aprendizagem de Máquina geralmente envolve algoritmos complexos, e estes são usados quando há um grande volume de dados, ou na execução de tarefas mais difíceis. Se o objetivo do *chatbot* for tirar dúvidas pontuais de clientes em uma loja virtual de vendas, então não é necessário criar um *chatbot* que utilize Aprendizagem de Máquina. Entretanto, dando uma olhada para o futuro, uma tecnologia que está se apresentando muito promissora é a Computação Cognitiva. Na Computação Cognitiva, várias tecnologias são combinadas, como exemplo podemos citar: PLN, Recuperação de Informação, Aprendizagem de Máquina, Representação do Conhecimento e Raciocínio. O IBM Watson²³ é um exemplo de um sistema que aplica esse tipo de tecnologia (MCTEAR; CALLEJAS; GRIOL, 2016c; FERRUCCI, 2012).

2.3 Tecnologias para o Desenvolvimento de Chatbots

O desenvolvimento de *chatbots* é um processo complexo, e pode envolver a integração de diferentes ferramentas (MCTEAR; CALLEJAS; GRIOL, 2016a). Citaremos aqui tecnologias que, entre outras coisas, podem ser utilizadas para criação da base de conhecimento do *chatbot*.

Existem várias tecnologias para a construção de *chatbots*. Algumas delas surgiram recentemente, por exemplo, o Amazon Lex, que foi disponibilizada em novembro de 2016 (BAR, 2016). Entre essas tecnologias, podemos citar: o IBM Watson, o Bot Framework da Microsoft²⁴, o Virtual Human Toolkit²⁵, o Pandorabots²⁶, API.AI²⁷ e o Wit.ai²⁸.

²³ <<https://www.ibm.com/watson/br-pt/>>

²⁴ <<https://dev.botframework.com/>>

²⁵ <<https://vhtoolkit.ict.usc.edu/>>

²⁶ <<https://www.pandorabots.com/>>

²⁷ <<https://api.ai/>>

²⁸ <<https://wit.ai/>>

Quando iniciamos este trabalho de mestrado em 2015, algumas dessas tecnologias citadas anteriormente ainda não existiam. O critério de pesquisa adotado para definição da tecnologia a ser utilizada para construção do *chatbot* Beck foi investigar quais eram as tecnologias utilizadas pelos últimos cinco *chatbots* premiados na competição anual Loebner, e quais delas eram *open source*. Encontramos o *ChatScript*, com os *chatbots* Suzette (2010), Rosette (2011) e Rose (2014 e 2015). Em 2012 encontramos o *chatbot* Chip Vivant²⁹, que não tem divulgada detalhadamente a tecnologia empregada para o seu desenvolvimento (BRADEŠKO; MLADENIĆ, 2012). E, por último, o *chatbot* Mitsuku³⁰ (2013 e, embora não estivesse no nosso período de pesquisa, também ganhou o prêmio em 2016), que foi construído através do AIML (LOEBNER, 2015).

Assim, optamos por investigar com mais detalhes essas duas tecnologias, AIML e *ChatScript*. Descrevemos, a seguir, mais detalhes sobre essas duas tecnologias de desenvolvimento de *chatbots*.

2.3.1 AIML

A tecnologia AIML é separada em Interpretador e Base de Conhecimento. O Interpretador AIML pode ser encontrado em diferentes linguagens de programação, como exemplos: Java³¹, PHP³², Ruby³³, entre outras (FOUNDATION, 2017). O Interpretador é o responsável por ler os arquivos AIML, receber as entradas do usuário e devolver uma resposta. A Base de Conhecimento, codificada em arquivos AIML, é utilizada para armazenar as regras de casamento de padrão do *chatbot*. Ilustrativamente, podemos afirmar que o Interpretador é o corpo e os arquivos AIML são o cérebro do *chatbot*.

Nos arquivos AIML, as categorias são as unidades básicas de conhecimento. Cada categoria constitui uma regra que é utilizada para codificar uma possível informação (entrada) do usuário e produzir uma resposta (saída). As categorias são compostas por:

- Um padrão *<pattern>* - Utilizado para casar com a entrada do usuário;
- Um modelo *<template>* - Usado para dar uma resposta ao usuário; e
- Um contexto, representado pelas *tags <that>* e *<topic>* - Utilizadas para tratar as réplicas do usuário e agrupar categorias que tratam de assuntos correlatos em uma conversa.

Na Figura 10, temos um exemplo da estrutura básica de um arquivo AIML (ABUSHAWAR; ATWELL, 2015). As *tags <that>* e *<topic>* são de uso opcional, quando não fornecidas pelo usuário, o sistema as preenche com valores padrões.

²⁹ <<http://www.chipvivant.com/about/>>

³⁰ <<http://www.mitsuku.com/>>

³¹ <https://www.java.com/pt_BR/>

³² <http://php.net/manual/pt_BR/intro-what-is.php>

³³ <<https://www.ruby-lang.org/pt/>>

Figura 10 – Exemplo da Estrutura de um Arquivo AIML.

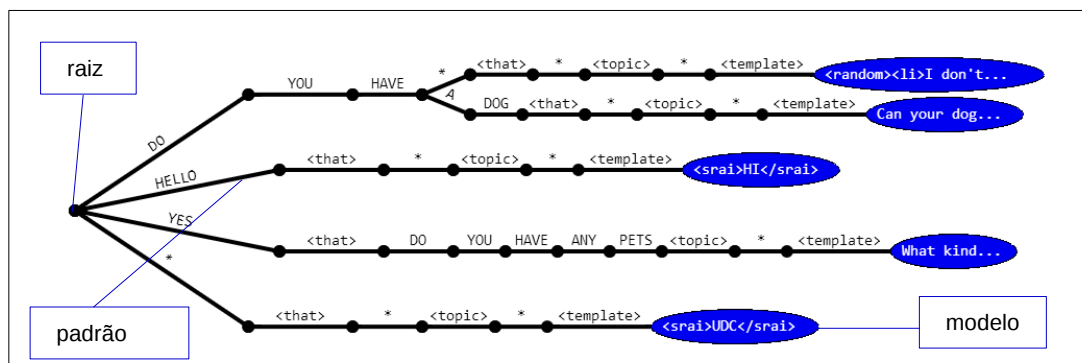
```

<aiml>
  <category>
    <pattern> O QUE É UM CÍRCULO? </pattern>
    <template> É uma figura plana cuja periferia, circunferência, está
    toda à igual distância do seu centro.
  </template>
</category>
</aiml>

```

Fonte: Baseado em Priberam (2017) e Ringate et al. (2011).

O Interpretador AIML cria um objeto chamado *Graphmaster* quando carrega os arquivos AIML. No *Graphmaster*, é inserido um caminho para cada padrão (<pattern>) associado às categorias existentes nesses arquivos. Cada caminho introduzido possui uma direção e uma raiz. No final de cada caminho do *Graphmaster*, existe um link para o modelo (<template>). A Figura 11 ilustra o objeto *Graphmaster* criado.

Figura 11 – Exemplo do Objeto *Graphmaster*.

Fonte: (WALLACE, 2014).

Dada uma mensagem do usuário, o Interpretador AIML constrói um caminho para essa entrada, semelhante ao caminho padrão do *Graphmaster*. Esse caminho construído contém: a sentença do usuário, a última resposta do *chatbot*, representada pelo valor da tag <that>, e o valor da tag <topic>. Todas essas informações são normalizadas antes de o caminho ser criado. A partir daí, o algoritmo de casamento de padrão pesquisa no *Graphmaster* a entrada do usuário. A pesquisa prossegue em sequência utilizando o algoritmo de busca em profundidade. Ao analisar um ramo do grafo onde não é encontrada uma correspondência, o algoritmo de busca retrocede (*backtracking*) para o último nó com ramos ainda não explorados e recomeça a pesquisa.

A sequência de pesquisa em cada nó do grafo é realizada seguindo um grau de prioridade, vejamos a seguir essa sequência.

- \$palavra - A palavra que tem o símbolo cifrão a antecedendo tem a maior prioridade para o casamento de padrão.
- # - O curinga cerquilha representa o casamento de zero ou mais palavras.
- _ - O curinga *underscore* representa o casamento de uma ou mais palavras.
- palavra - Casamento com a palavra exata.
- <set> nome </set> - Casamento com o atributo nome.
- ^ - O curinga circunflexo representa o casamento de zero ou mais palavras.
- * - O curinga asterisco representa o casamento de uma ou mais palavras.

Se nenhum casamento de padrão for encontrado no grafo para a entrada do usuário, o Interpretador deve retornar uma mensagem padrão como: “Eu não tenho uma resposta para isso”. Essa resposta deve ser especificada previamente pelo *botmaster*³⁴.

Existem três tipos de categorias em AIML (SHAWAR; ATWELL, 2007):

1. **Categorias atômicas** - É o tipo de categoria mais fácil e simples de criar no AIML. Os padrões dessa categoria não contém símbolos curinga. O exemplo da Figura 10, é também um exemplo de categoria atômica em AIML.
2. **Categorias comuns** - São aquelas que possuem padrões com símbolos curinga. Esses símbolos podem corresponder a qualquer palavra digitada pelo usuário. No exemplo da Figura 12, se não for encontrada uma categoria atômica que corresponda exatamente com a sentença do usuário “Eu gosto de”, então o mecanismo do AIML tentará encontrar uma categoria comum para essa entrada do usuário.

Figura 12 – Exemplo de Categoria Comum.

```
<aiml>
  <category>
    <pattern> EU GOSTO DE * </pattern>
    <template> verdade? </template>
  </category>
</aiml>
```

Fonte: Baseado em Ringate et al. (2011).

3. **Categorias recursivas** - São as categorias que contém as *tags* <srail> e <sr>, que são utilizadas para tratar as variabilidades das entradas do usuário de forma recursiva. A categoria recursiva mapeia a entrada do usuário para outra categoria da

³⁴ Pessoa que codifica bases de conhecimentos de *chatbots*.

base de conhecimento. Essas categorias podem ser utilizadas de diferentes maneiras, como (WALLACE, 2009):

- Redução simbólica - Limita formas gramaticais complexas e transforma em formas mais simples. Exemplo: O padrão “VOCÊ SABE QUEM É *”, é transformado para “QUEM É *”;
- Dividir e conquistar - Divide uma entrada em duas ou mais partes e combina as respostas de cada uma;
- Detecção de palavras-chave - Procura palavras-chave em qualquer lugar da entrada do usuário;
- Correções ortográficas ou gramaticais - Corrige e normaliza a entrada do usuário; e
- Tratamento de sinônimos - Mapeia as palavras de sentido semelhante para que correspondam a um mesmo padrão.

Na Figura 13, temos exemplo de uma expressão que denota despedida. Existem várias expressões, termos ou palavras (sinônimos) que utilizamos para se despedir de alguém (e.g., tchau, até mais, adeus, vou nessa). Usando uma categoria recursiva conseguimos mapear vários padrões *<pattern>* para uma mesma saída *<template>*. É importante destacar que, no exemplo da Figura 13, deverá existir uma categoria atômica com um padrão “tchau” para que o usuário receba uma resposta. Há riscos no uso das categorias recursivas, o *botmaster* poderá, se não tiver cuidado, criar *loops* infinitos.

Figura 13 – Exemplo de Categoria Recursiva.

```
<aiml>
  <category>
    <pattern> ATÉ MAIS </pattern>
    <template> <srai> tchau </srai> </template>
  </category>
</aiml>
```

Fonte: Baseado em Ringate et al. (2011).

Como a linguagem AIML é derivada do XML, é necessário abrir e fechar muitas *tags*, tornando mais difícil a sua leitura. Além disso, é necessária a criação de um grande número de regras para se ter um *chatbot* robusto (MCNEAL; NEWYEAR, 2013).

Processamento de Linguagem Natural no AIML

O AIML permite a criação de uma base de substituição, representada pela *tag* *<substitu-*

tions>, e esta pode envolver uma ou mais *tags* <substitute>, que são usadas para realizar a substituição de palavras ou expressões. A base de substituição pode ser usada para corrigir erros de digitação, exemplo: <substitute find= “paralizado” replace= “paralisado”>; ou para realizar normalizações, como <substitute find= “ahhhh” replace= “ah”>.

A *tag* <substitute>, como pode ser observada nos exemplos anteriores, possui os parâmetros *find*, que procura a palavra ou expressão a ser substituída, e o parâmetro *replace*, que realiza a substituição. A base de substituições representa um tratamento de informações no nível morfológico da arquitetura de PLN.

O conjunto de todas as categorias em AIML constitui a base de conhecimento, essa base é responsável por realizar o tratamento sintático da entrada do usuário, representando o tratamento das informações sintáticas.

A *tag* <topic> é utilizada para agrupar as categorias que possuem um tema em comum, por exemplo, “filmes”. Assim, em uma conversa, o *chatbot* pode explorar um assunto específico com o usuário, dando a impressão de ser mais inteligente. Isso representa o tratamento das informações semânticas.

A substituição de pronomes é semelhante à substituição morfológica, a diferença é que a *tag* <substitute> está envolvida pela *tag* <person>, indicando que a substituição é pronominal. Exemplo: <person> <substitute find= “você é” replace= “eu sou”/> </person>. Essas substituições representam o tratamento de informações discursivas.

As variáveis de controle de dêiticos, exemplo: “aqui” e “hoje”, são definidas em tempo de execução. Essas variáveis são definidas pela *tag* <set> e recuperadas pela *tag* <get>. Representando também o tratamento de informações discursivas.

As informações do *chatbot* são referenciadas através da *tag* <bot>, que pode envolver uma ou mais *tags* <property>. A *tag* <property> possui os parâmetros *name*, que representa o nome do atributo, e *value*, que representa o valor do atributo. Exemplo: <bot> <property name= “nome” value= “ALICE”> </bot>. Tratamento das informações pragmáticas (NEVES, 2005; BARROS; TEDESCO, 2016).

Em relação ao contexto em AIML, a *tag* <that> se refere a última resposta fornecida ao usuário. Uma aplicação comum, para o uso dessa *tag*, pode ser exemplificada nas perguntas nas quais as respostas habituais são “sim” ou “não”, como podemos ver na Figura 14. Essa regra é utilizada quando o usuário responde “sim”. Então, o *chatbot* procura em sua base de conhecimento o motivo pelo qual o usuário respondeu “sim”. Se a última frase escrita pelo *chatbot* foi “Você gosta de filmes?”, então essa categoria é utilizada.

2.3.2 ChatScript

Diferentemente do funcionamento da linguagem AIML, que é modularizada (arquivos AIML e o Interpretador), o *ChatScript* não precisa de um programa à parte para interpretar os seus arquivos. O *ChatScript* é a linguagem de desenvolvimento de *chatbots* e também o próprio interpretador. Na verdade, a linguagem *ChatScript* pode ser utilizada

Figura 14 – Exemplo de Uso de Contexto em AIML

```

<aiml>
  <category>
    <pattern> SIM </pattern>
    <that> VOCÊ GOSTA DE FILMES? </that>
    <template> Qual é o seu filme favorito? </template>
  </category>
</aiml>

```

Fonte: Baseado em Wallace (2014).

para diferentes propósitos dentro de PLN, mas ela tem sido usada, principalmente, para o desenvolvimento de *chatbots*. *ChatScript* é *open source* e não precisa ser instalada no computador, sendo apenas necessário realizar o *download* do repositório SourceForge³⁵, e descompactação dos arquivos para começar a usá-la (MCNEAL; NEWYEAR, 2013).

A base de conhecimento do *chatbot*, no *ChatScript*, é organizada através de regras e estas são reunidas em tópicos (WILCOX, 2016).

Os tópicos, com suas respectivas regras, são armazenados nos arquivos *.top* (essa extensão do arquivo é derivada da palavra tópico). Ao contrário do AIML, que procura primeiramente encontrar o melhor casamento de padrão, o *ChatScript* tenta inicialmente encontrar o melhor casamento entre os tópicos e, em seguida, procura a melhor regra que está contida nesse tópico (MCNEAL; NEWYEAR, 2013).

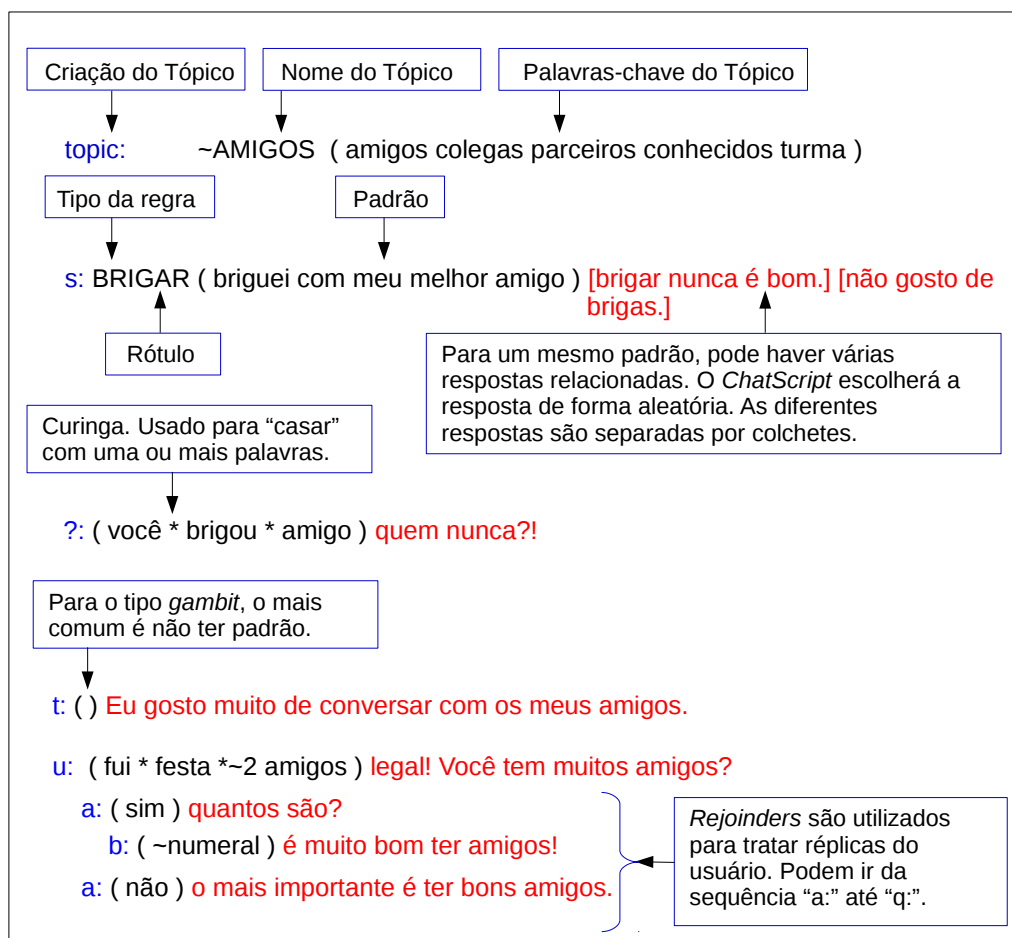
Na Figura 15 temos um exemplo de um arquivo *.top* em *ChatScript*. Nesse exemplo, declaramos um tópico sobre amigos e as regras que estão relacionadas sobre esse assunto. Como podemos observar na Figura 15, a palavra reservada *topic* é utilizada para criar um tópico. Em seguida, temos o nome do tópico, que deve ser precedido do acento gráfico til. Por convenção, os nomes dos tópicos devem sempre estar em letras maiúsculas. Após o nome do tópico, temos as palavras-chave que representam o tópico entre parênteses (amigos, colegas, parceiros, conhecidos, turma).

Ao lado do tipo da regra temos o rótulo, que é opcional. Todas as regras podem ter um rótulo, ele ajuda na depuração do sistema ou pode ser usado para referenciar outras regras. Após o rótulo, temos o padrão, que é delimitado por parênteses. Uma regra só pode ser executada quando seu padrão coincide com a mensagem informada pelo usuário. E, por fim, temos a resposta que o *chatbot* apresentará, caso a regra seja executada (WILCOX, 2016).

Numa regra, o primeiro elemento é o seu tipo, e este pode ser de cinco tipos diferentes. São eles:

- **s:** - afirmação - Tipo de regra usada para casar padrão com afirmações (*statements*)

³⁵ <<https://sourceforge.net/projects/chatscript/>>

Figura 15 – Exemplo Arquivo de Tópico no *ChatScript*.

Fonte: Autoria Própria.

feitas pelo usuário.

- **?:** - interrogação - Usada para casar padrão com entradas do usuário que são perguntas.
- **u:** - afirmação e interrogação - Esse tipo de regra pode tanto ser usada quando o usuário faz uma pergunta, quanto uma afirmação.
- **t:** - *gambit* - Essa regra é usada pelo *chatbot* para contar uma história, ou tentar conduzir a conversa com o usuário para um determinado assunto. Geralmente, esse tipo de regra é executado quando o *chatbot* não encontra um casamento de padrão para a frase informada pelo usuário. O *gambit* possui dois tipos: o “t:” que é executado sequencialmente em um tópico, e o tipo “r:”, executado de forma randômica.
- **a:** - *rejoinder* - Esse tipo de regra é usado, por exemplo, quando o usuário responde uma pergunta que o *chatbot* tenha feito. Quando o usuário responde uma pergunta, dependendo do contexto da conversa, o *chatbot* pode fazer outras perguntas em

seguida. Então, para essa situação, o mecanismo do *ChatScript* disponibiliza outros níveis de *rejoinders*, que pode ir da letra “a:” até “q:”.

Existe uma classificação para as regras segundo o seu tipo. Os tipos “s:”, “?:” e “u:”, são classificados como *responders*; os tipos “t:” e “r:” são classificados como *gambits*; e, por fim, os tipos “a:”, “b:” até “q:” são chamados de *rejoinders*.

Em relação a Figura 15, ainda podemos observar que na primeira regra (tipo u:) o padrão (briguei com meu melhor amigo) tem duas resposta associadas a ele. Todas as regras em *ChatScript* podem ter múltiplas respostas relacionadas, exceto as do tipo *gambit*. Para a regra do tipo *gambit* é opcional ter um padrão associado, o mais comum é não ter. Nas regras do tipo “?:” e “s:”, usamos o curinga (*) para representar uma ou mais palavras na frase do usuário.

Para criar um padrão, o *ChatScript* oferece uma gama de recursos, tornando maleável e poderoso o seu mecanismo de casamento de padrão. Citaremos a seguir alguns desses recursos (WILCOX, 2017b).

- **Curingas** - O *ChatScript* disponibiliza para o usuário vários tipos de curingas, que facilitam e tornam mais poderoso o casamento de padrão; alguns curingas são:
 - * (*Indefinite Wildcard*) Curinga indefinido - permite o casamento com qualquer número de palavras;
 - *n (*Precise Wildcard*) Curinga preciso - Onde “n” é um número positivo, permite o casamento com um número exato de palavras; e
 - *~n (*Range-restricted wildcard*) Curinga de alcance limitado - Permite o casamento com o intervalo de 0 (zero) até um número positivo “n” de palavras.
- **Limites de sentenças (< e >)** - Em alguns casos, para obter um casamento correto do padrão, precisaremos saber onde uma sentença começa e termina. Por exemplo, na regra: **?: (o que é uma ervilha) semente pequena e redonda, geralmente verde, contida em uma vagem.** Essa regra casa com as entradas do usuário: “O que é uma ervilha?”; ou “Por favor, me diga, o que é uma ervilha?”; ou ainda “O que é uma ervilha no canto da parede da sala?”. Essa última frase casa de forma inadequada com a regra. Para evitar tal problema, podemos usar o símbolo “>” no final do padrão, para garantir que ele esteja localizado no final da frase do usuário. Assim, reescrevendo a regra para a forma mais adequada, teríamos: **?: (o que é uma ervilha >) ...**
- **Casamento não ordenado (<< e >>)** - Quando criamos um padrão, é pressuposto que o casamento da entrada do usuário com esse padrão ocorra em sequência. Entretanto, podemos alterar esse comportamento utilizando os símbolos << e >>. Por exemplo, no padrão: **s: (eu amo gatos)** e queremos que o padrão case com a

entrada do usuário em qualquer ordem. Podemos modificar esse padrão para **s: (<< eu amo gatos >>)**. Assim, qualquer ordem em que o usuário escrever as palavras “eu amo gatos”, casará com esse padrão.

- **Escolhas []** - O *ChatScript* permite que definamos um conjunto de palavras alternativas para realizar o casamento de padrão. Por exemplo, no padrão: **?: (você joga [bola futebol vôlei basquete] todos os dias?)**. Esse padrão casará com qualquer entrada do usuário que contenha uma das palavras delimitadas pelos colchetes (evidente que a entrada do usuário deverá conter também as outras palavras fora dos colchetes).
- **Conceitos** - A forma mais prática e organizada de usar as escolhas (representado pelos colchetes []) dentro de um padrão é definindo conceitos. Os conceitos são conjuntos de palavras que podem representar sinônimos ou palavras afiliadas. Para definir um conceito, utilizamos a palavra reservada *concept*. Exemplo: **concept: ~esportes (bola futebol vôlei basquete)**. A criação de conceitos permite que os reusemos em outros padrões, além, é claro, de deixar o código mais organizado e compreensível.
- **Forma canônica** - O *ChatScript* permite que o usuário flexibilize os seus padrões. Ele permite que a entrada do usuário possa simultaneamente casar com os padrões existentes em sua base de conhecimento utilizando a palavra original, ou sua forma canônica. Para substantivos, a forma canônica é o singular. Exemplo, na regra: **?: (cachorro) Eu tenho um gato**. Essa regra casará igualmente às entradas do usuário: “Eu gosto de cachorro” ou “Eu gosto de cachorros”.
- **Palavras Opcionais { }** - Algumas palavras são opcionais numa entrada do usuário. Podemos utilizar esse recurso nos padrões e fazer com que certas, se fornecidas pelo usuário, possam casar o padrão existente. Assim, se o usuário fornecer ou não essas palavras, o padrão será executado. Exemplo, no padrão: **u: ({eu} gosto de cantar)**, se o usuário não fornecer na frase a palavra “eu”, a regra será executada normalmente.
- **Fatos** - O *ChatScript* suporta triplas, que são estruturas de dados conhecidas no *ChatScript* como fatos. Os fatos têm, normalmente, três campos de cadeias de textos (*strings*). Para criar uma fato, utilizamos a função *createfact*. Exemplo: *createfact* (Junior tem animal). Para recuperar as informações armazenadas em fatos, utilizamos a função *query*.

O fluxo de funcionamento típico do *ChatScript* obedece uma certa ordem de execução.

1. Quando o *ChatScript* recebe uma entrada do usuário, primeiramente tenta verificar se a entrada corresponde a algum padrão de um *rejoinder* que está pendente. Isso permite que o sistema teste diretamente as regras relacionadas à sua última resposta.
2. Se nenhuma resposta for gerada por algum *rejoinder*, então o sistema tentará casar a sentença do usuário com os *responders* (regras que começam com s:, ?: ou u:). Primeiramente tentará casar com o tópico ativo. Caso falhe, ele tentará casar a entrada do usuário com as palavras-chave dos demais tópicos (se houver). Se for encontrada alguma correspondência da entrada do usuário com as palavras-chave de outro tópico, então o sistema tentará casar a entrada do usuário com as regras pertencentes a esse novo tópico.
3. Se não forem encontrados *responders* para a entrada do usuário, então o *ChatScript* tentará executar um *gambit*. Primeiramente, a partir do tópico onde houve algum casamento de padrão com as palavras-chave do tópico. Se isso não for possível, então o sistema tentará executar um *gambit* do tópico que estava vigente anteriormente. Se mesmo assim falhar, então o sistema executará um *gambit* de forma aleatória de qualquer tópico.
4. Uma vez que a resposta é apresentada ao usuário, o trabalho é finalizado, voltando ao primeiro deste ciclo.

É importante notar que os passos de funcionamento do *ChatScript* descritos anteriormente ilustram o funcionamento típico do sistema. Uma vez que esse fluxo de controle é mais um tópico do *ChatScript* (arquivo .top), e pode ser modificado pelo *botmaster* da forma como ele desejar.

Embora o *ChatScript* tenha sido criado para a língua inglesa, ele pode ser usado para outras línguas (e.g., Espanhol, Alemão, Japonês). No entanto, adaptações precisam ser realizadas. Por exemplo, o dicionário possui apenas palavras na sua forma canônica, e o mecanismo do *ChatScript* é que realiza todas as flexões das palavras baseada na sua categoria gramatical. O corretor ortográfico também só funciona para a língua inglesa (WILCOX, 2017b).

Recentemente, a partir da versão 6.91 do *ChatScript*, foi disponibilizada uma forma de usar o recurso de *Stemming* e *POS-Tagging* externos à linguagem. Para isso, foram disponibilizadas duas opções para o usuário. A primeira é utilizar o *POS-Tagger Tree-Tagger*³⁶, que cobra pela licença comercial, e sua utilização para fins de pesquisas possui uma dificuldade, que é a criação de um arquivo para receber o resultado do seu processamento. A segunda opção é utilizar o recurso que o *ChatScript* oferece para usar qualquer

³⁶ <<http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/>>

POS-Tagger externo. Entretanto, o usuário precisa desenvolver um *script* para conseguir utilizar esse recurso.

Processamento de Linguagem Natural no *ChatScript*

Em relação ao tratamento de informações morfológicas, o *ChatScript* possui os seguintes recursos:

- Corretor ortográfico baseado na Distância de Levenshtein (ou Distância de Edição) (LEVENSHTEIN, 1966). Esse algoritmo é utilizado para corrigir palavras do mesmo tamanho com tolerância de mais ou menos um caractere.
- Base de correção de erros ortográficos, que pode ser utilizada para substituir palavras (ou sequência de palavras) predefinidas. Essa base é utilizada para corrigir a entrada do usuário ou torná-la mais fácil e adequada para o mecanismo de casamento de padrão.
- Etiketador gramatical (*Parts of Speech*), utilizado para identificar a classe gramatical das palavras de acordo com o papel que ela exerce na frase do usuário.
- Dicionário interno, que possui apenas a forma canônica das palavras. O *ChatScript* tem a capacidade de reconhecer as variações das palavras (e.g., conjugação dos verbos, afixos).

O *ChatScript* tem a capacidade de realizar o tratamento de informações sintáticas nas entradas do usuário, ou seja, ele pode determinar o papel das palavras de acordo com a sua função (e.g., sujeito, predicado, objeto direto). Como já citado, esse processo também é conhecido como *Parsing*.

O *Parsing* faz o tratamento na frase do usuário em duas etapas:

1. Verifica a frase do usuário com base nas regras cadastradas nas base de dados, que tem por objetivo eliminar diferentes significados das palavras. Exemplo, na frase: “*The man has slept.*” (tradução literal para o português, “O homem tem dormido.”), o verbo auxiliar “*has*” (ter) antecede o verbo “*slept*” (dormir) no *past participle* (particípio). Assim, o significado do verbo “ter” (que pode ter vários significados) foi determinado por sua função sintática. O objetivo dessa etapa é reduzir a ambiguidade encontrada nas frases (WILCOX, 2017b).
2. Procura determinar o papel de cada palavra numa frase sugerindo várias etiquetas gramaticais possíveis e alterando-as, caso algum erro tenha sido detectado. Essa etapa usa o algoritmo *Garden Path* (FONSECA; VIGÁRIO, 2011). Esse algoritmo funciona etiquetando as palavras da forma mais simples possível. Entretanto, quando o algoritmo alcança um certo ponto da frase em que foi percebido uma incoerência, então o algoritmo retrocede e corrige o erro encontrado. Por exemplo, na frase: “*While*

Mary was mending the sock fell off her lap.” (Enquanto Mary estava remendando a **meia** caiu do seu colo). Numa leitura rápida, tendemos a classificar a palavra “meia” como objeto direto do verbo “remendar”, mas depois podemos perceber que esse sintagma é, na verdade, o sujeito da oração “A meia caiu do seu colo”.

Em relação ao tratamento de informações semânticas, o *ChatScript* organiza todas as regras que são utilizadas para realizar o casamento de padrão em tópicos. Assim, as regras relacionadas ao tópico precisam ter um significado comum.

No que diz respeito ao tratamento de informações discursivas, o *ChatScript* possui uma base de substituição que pode ser utilizada para realizar a substituição de pronomes.

Em relação às informações pragmáticas, o *ChatScript* possui variáveis que armazenam dados sobre o usuário. Além dessas variáveis, a linguagem também possui recurso para criação de fatos, que são estruturas utilizadas para armazenar informações (já visto anteriormente).

2.3.3 AIML versus *ChatScript*

Ao realizar pesquisas sobre as linguagens AIML e *ChatScript*, encontramos alguns trabalhos nos quais os autores realizaram comparações entre as linguagens. Os resultados podem ser observados na Tabela 1.

Tabela 1 – Comparação entre o AIML e o *ChatScript*.

Autor	AIML	<i>ChatScript</i>
(SCHUETZLER, 2015)	Especificação dos diálogos usa sintaxe derivada do XML. Isso é útil para sua extensibilidade, porém difícil para realizar manutenção. A criação de um simples <i>chatbot</i> requer um grande número de aberturas e fechamentos de <i>tags</i> . A leitura do código se torna difícil.	Possui muitos recursos que tornam a criação de <i>chatbots</i> mais fácil e gerenciável. Possui uma sintaxe mais “limpa” que AIML porque não é baseada em XML, assim tornando-a mais simples de ler. Possui curingas como AIML, mas de uma maneira que possamos restringir o número de palavras com as quais deve casar.
(MCTEAR; CALLEJAS; GRIOL, 2016b)	No AIML 2.1 há uma facilidade para criar ontologias que o <i>chatbot</i> pode usar, e “raciocine” sobre o conhecimento. Possui o casamento de padrão baseado em palavras.	Tem a facilidade de usar ontologias a partir do Wordnet ^a . Casamento do padrão baseado no significado das palavras.
(WEERATUNGA et al., 2015)	Usa casamento de padrão. Simples e fácil de aprender, por isso é indicado para iniciantes. Baseado em XML, que não possui uma leitura amigável.	Casamento de padrão baseado em atos de fala. Usa casamento de padrão, fácil de criar e usa uma sintaxe visual simples.
(MCNEAL; NEWYEAR, 2013)	Simples. Fácil de aprender. Fácil de implementar. Bases de AIML com pré-autoria estão disponíveis. Possui casamento de padrão relativamente fraco. Pode ser necessário um grande tempo para fazer um bom <i>chatbot</i> e é difícil de realizar manutenção. Um grande número de categorias são necessárias para criar um <i>chatbot</i> robusto.	Casamento de padrão forte, poderoso, flexível e eficiente. Excelente documentação e suporte disponíveis. Desempenho comprovado em uso comercial. Mais difícil de aprender do que AIML. Ter experiência de programação é útil, mas não é necessário. Mais difícil de incorporar em uma página da <i>web</i> .
(BRADEŠKO; MLADENIĆ, 2012)	Sintaxe baseada em XML e consiste principalmente em regras de entrada (categorias) com a saída apropriada. O padrão descrito deve cobrir a entrada inteira. É possível usar curinga para casar com uma ou mais palavras. O verdadeiro poder do AIML reside na sua capacidade de usar recursão.	Sucessor da linguagem AIML. <i>ChatScript</i> é focada numa melhor sintaxe, que faz com que ela seja melhor para realizar manutenção. Ela corrige os problemas do casamento de padrão com nenhuma palavra e introduz várias funcionalidades adicionais, tais como: conceitos, lógica <i>and/or</i> , variáveis, triplas de fatos e funções.

^a <<https://wordnet.princeton.edu/>>**Fonte:** Autoria Própria.

Tabela 1 - Continuação da Comparação entre o AIML e o *ChatScript*.

Autor	AIML	<i>ChatScript</i>
(BELLOTTI et al., 2011)	Considerada de baixo nível. O uso de curingas e recursão implica na modificação da entrada do usuário, o que é muito problemático, particularmente se considerarmos a manutenção das regras criadas. A utilização de curingas é propensa a trazer resultados indesejados.	Foi concebida para favorecer a extração do significado da sentença e fornecer uma sintaxe que pode ser facilmente implementada, aumentando assim a facilidade da manutenção do código. <i>ChatScript</i> considera pontuação na análise do texto e tem uma sintaxe clara para diálogos aninhados. <i>ChatScript</i> suporta o casamento de padrão com as formas canônicas e originais dos verbos, e a formação de grupos de palavras em “conceitos”. Esses conceitos podem também explorar ontologias do WordNet. Isso permite que os padrões se aproximem do seu significado.
(WILCOX, 2011)	Uma vez que o casamento de padrão em AIML é tão primitivo e genérico, ele deve ter várias categorias de informações para realizar uma única tarefa. Se desejarmos que o sistema responda a uma palavra-chave sozinha ou com um prefixo, infix, ou sufixo, devemos ter quatro categorias para fazê-lo, uma para cada condição. Usa casamento de padrão simples, não faz aproximação com o significado das palavras. Realizar a depuração do código é difícil de fazer.	Possui uma sintaxe clara e não precisa de várias regras para executar uma única tarefa. Os curingas permitem restringir o número de palavras com as quais ele deve casar.

Fonte: Autoria Própria.

Como pode ser observado a partir das tabelas, o *ChatScript* apresenta melhores recursos para o desenvolvimento de *chatbots*. Com base nesse levantamento, decidimos adotar essa linguagem neste trabalho.

2.4 Considerações Finais

Neste capítulo, abordamos PLN, que é caracterizado como uma subárea da IA. Vimos os seus módulos de processamento, e onde PLN pode ser aplicado. Mostramos que os *chatbots* são ACs que surgiram a partir da proposição do Teste de Turing. Desde a proposição desse teste, vários *chatbots* foram criados e atualmente está acontecendo uma revolução em seu

uso. Vimos a evolução dos *chatbots* através de suas gerações e também as formas derivadas desse tipo de agente, que foram surgindo devido ao avanço de PLN e da IA. Apresentamos algumas tecnologias para o desenvolvimento de *chatbots*, e exploramos com mais detalhes duas delas, AIML e *ChatScript*. Por fim, mostramos uma tabela comparativa dessas duas tecnologias, e defendemos o uso do *ChatScript* por oferecer melhores recursos.

3 SISTEMAS DE APOIO AO TRATAMENTO DE TRANSTORNOS PSICOLÓGICOS

Descrevemos, neste capítulo, alguns dos principais sistemas automáticos para auxílio a pessoas que sofrem de transtornos psicológicos. Esses sistemas se baseiam em teorias da Psicologia, a fim de prover um auxílio efetivo aos pacientes. Algumas das teorias utilizadas na construção desses tipos de sistemas são: Terapia Cognitivo-Comportamental (TCC) (BECK, 1995), *Mindfulness* (PEPPING; O'DONOVAN; DAVIS, 2013), Terapia Racional Emotiva Comportamental (TREC) (ELLIS; ELLIS, 2002), Terapia Interpessoal (WEISSMAN; MARKOWITZ, 1994) e Dessensibilização e Reprocessamento Através de Movimentos Oculares (do inglês, *Eye Movement Desensitization and Reprocessing* (EMDR)) (SHAPIRO; SOLOMON, 1995).

Dentre os trabalhos da Psicologia para tratamento da depressão, destacamos a TCC (seção 3.1), que foi escolhida como base para o desenvolvimento desta pesquisa de mestrado. Essa teoria foi escolhida porque é de fácil compreensão por parte de leigos e tendo sido usada em sistemas variados de apoio psicológico (seções 3.2 e 3.3). Vale mencionar que nem todos os sistemas apresentados neste capítulo são baseados na TCC. Contudo, uma discussão mais abrangente sobre teorias e tipos de tratamento psicológico foge ao escopo deste trabalho. Por fim, temos a seção 3.4 na qual apresentamos uma tabela comparativa com algumas características de todos os sistemas vistos, e realizamos as considerações finais deste capítulo.

3.1 Terapia Cognitivo-Comportamental

A Terapia Cognitivo-Comportamental (TCC) (do inglês, *Cognitive Behavioural Therapy* (CBT)) é um tipo de psicoterapia que foi proposto pela primeira vez por Aaron T. Beck, na década de 1960. Inicialmente proposta para explicar os processos psicológicos da depressão, ela é também usada para tratar um grande número de desordens mentais (e.g., Transtorno Obsessivo-Compulsivo, Distúrbio de Ansiedade Generalizada, Transtorno do Pânico, Fobia Social, Abuso de Substâncias) (BECK, 1995). Beck propôs que os sintomas da depressão poderiam ser traduzidos como interpretações tendenciosas negativas de eventos do cotidiano, por exemplo, experiências da vida social. Esses eventos podem ser interpretados pelo indivíduo através de percepções negativas que ele tem de si mesmo, do mundo e do futuro (a tríade cognitiva) (KNAPP; BECK, 2008).

A TCC tem como objetivo a redução e a prevenção dos sintomas da depressão atra-

vés da identificação e alteração de pensamentos disfuncionais (BECK, 2005). Os termos “Terapia Cognitiva” e “Terapia Cognitivo-Comportamental” são usados com frequência como sinônimos para descrever psicoterapias baseadas no modelo cognitivo.

A forma como os indivíduos percebem e processam a realidade influenciará a maneira como eles se sentem e se comportam, esse é o princípio fundamental da TCC. A TCC alega que há pensamentos que ocorrem espontaneamente e são frutos de uma interpretação imediata de qualquer situação. Esses pensamentos são chamados de pensamentos automáticos. Eles são geralmente aceitos como plausíveis e verdadeiros. A maioria das pessoas não está consciente da presença dos pensamentos automáticos, a não ser que sejam treinadas para identificá-los e monitorá-los. De acordo com Beck, é igualmente possível perceber um pensamento, se concentrar nele e avaliá-lo, assim como é possível identificar e refletir sobre uma sensação como a dor (KNAPP; BECK, 2008). Quando os pensamentos trazem consequências negativas para a pessoa, eles são então chamados de *pensamentos automáticos negativos*.

Na raiz dessas interpretações automáticas negativas estão os pensamentos disfuncionais mais profundos, chamados de esquemas (também denominados crenças nucleares ou centrais) (ALMEIDA; NETO, 2003; KNAPP; BECK, 2008). Os esquemas são as bases sobre as quais se desenvolvem os pensamentos automáticos. Essas estruturas são relativamente duradouras, e são usadas para armazenar informações genéricas de estímulos, ideias ou experiências. A partir delas, são organizadas as novas informações, determinando como os fenômenos são percebidos e conceitualizados pelos indivíduos. Uma vez que uma determinada crença central se forma, ela pode influenciar o desenvolvimento de novas crenças e, se persistirem, são incorporadas à estrutura cognitiva duradoura. Crenças nucleares embutidas nessas estruturas cognitivas modelam o estilo de pensamento de um indivíduo e promovem erros cognitivos encontrados na psicopatologia. Na Tabela 2 temos exemplos de alguns erros de pensamento encontrados em diferentes transtornos emocionais.

Tabela 2 – Exemplos de Erros de Pensamento.

Erros de Pensamento:
Catastrofização ou Previsão do Futuro: Antecipar o futuro em termos negativos, sem levar em consideração a possibilidade de outros desfechos. Exemplos: “Meu namorado não atende o celular, deve estar com outra.”; “Eu sei que nunca terei sucesso na vida.”; “Que terrível, eu não tenho nenhuma roupa para vestir para a festa, isto é o fim.”.
Rotulação: Colocar um rótulo global e fixo em si mesmo e/ou nos outros. Exemplos: “Sou um fracasso.”; “Ele é um completo imbecil.”; “Quão burro eu sou.”.
Leitura Mental: Chegar a uma conclusão sem evidências adequadas, tentando presumir o que os outros estão pensando. Exemplos: “Eu sei que todos na escola pensam que sou burro.”; “Elas sentem pena de mim.”.
Polarização: Classificar a situação, a pessoa ou o acontecimento apenas em duas categorias, mutualmente exclusivas. Exemplos: “Deu tudo errado na festa.”; “Ninguém gosta de mim.”; “Tudo foi uma perda total.”; “Se eu não tirar sempre as notas máximas, serei um fracasso.”.
Ampliação/Minimização: Julgar a si mesmo, os outros e as situações, ampliando os aspectos negativos e/ou minimizando os aspectos positivos. Exemplos: “Minha nota foi 8 na prova. Isso demonstra o quanto fui ruim.”; “Tirei 10 no teste. Isso significa que foi muito fácil.”;
Raciocínio Emocional: Presumir que os sentimentos são fatos porque tem uma emoção envolvida. Exemplos: “Estou com muito medo, por isso deve ser realmente perigoso.”; “Tenho pavor de aviões, logo, voar deve ser perigoso.”.
Imperativos (“Deveria” e “Tenho que...”): Interpretar eventos em termos de como as coisas deveriam ser, ao invés de focar em como elas são. Exemplos: “Deveria ter dado mais atenção ao meu namorado, assim ele não teria me deixado.”; “Não devo cometer erros.”; “Eu deveria ter sido uma mãe melhor.”; “Eu tenho que ter controle sobre tudo.”.

Fonte: Baseado em Knapp e Beck (2008) e ZUGMAN (2011).

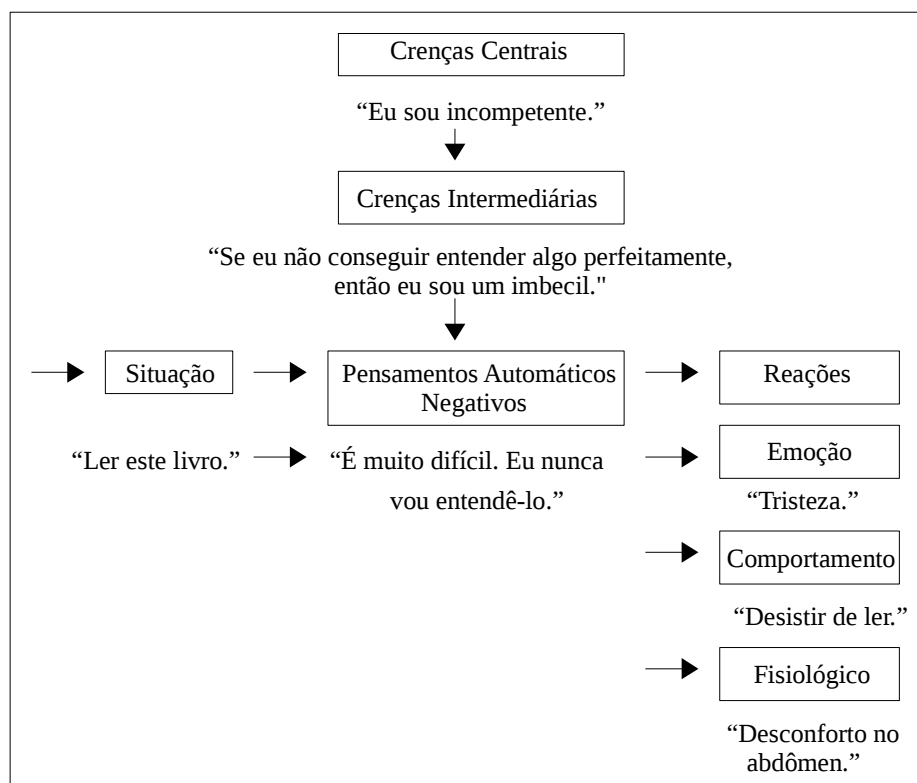
Os esquemas são adquiridos precocemente, durante a infância, agindo como “filtros” pelos quais as informações e experiências atuais são processadas (BECK, 1995). Essas crenças são moldadas por experiências pessoais e derivam da identificação do indivíduo com outras pessoas, e da percepção de suas atitudes. Os esquemas de indivíduos bem ajustados permitem avaliações realistas, ao passo que os de indivíduos mal ajustados levam a distorções da realidade, que, por sua vez, geram um transtorno psicológico.

Os esquemas influenciam a existência de uma classe intermediária de crenças que consiste em atitudes, regras e suposições. Por exemplo, um indivíduo tem a seguinte crença intermediária:

- **Atitude** - “É terrível ser incompetente.”;
- **Regras/Expectativas** - “Eu preciso trabalhar duro o tempo todo.”;
- **Suposição** - “Se eu trabalhar o máximo que posso, talvez eu consiga ser capaz de fazer algumas coisas que outras pessoas fazem facilmente”.

Essas crenças podem influenciar sua visão sobre uma situação, que por sua vez pode influenciar seu pensamento, seus sentimentos e seu comportamento. Todos esses elementos que são influenciados, podem ser resumidos em um modelo cognitivo. A Figura 16 ilustra o modelo cognitivo.

Figura 16 – Modelo Cognitivo.



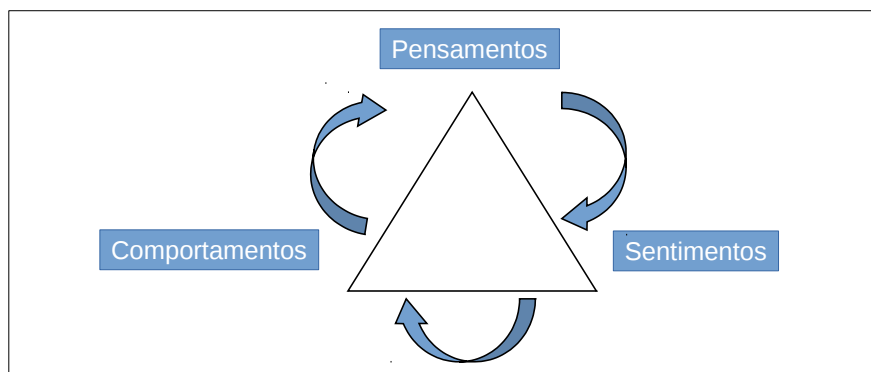
Fonte: Baseado em Beck (1995).

É importante salientar que a TCC não afirma que o pensamento disfuncional sozinho provoca o sofrimento emocional, mas sim é parte integrante desse problema. O distúrbio psicológico é acrescido por fatores genéticos, ambientais, familiares, culturais, da personalidade e de fatores do desenvolvimento (NEENAN; DRYDEN, 2014).

Os pensamentos, sentimentos, comportamentos, fatores ambientais e fisiológicos estão todos conectados. Cada um desses elementos é capaz de influenciar os outros em um ciclo interativo. A mudança em um desses elementos influencia a mudança nos demais (NEENAN; DRYDEN, 2014). O que pensamos influencia o que sentimos e a forma como nos comportamos. Os sentimentos influenciam o que pensamos e como agimos, e assim por diante. É como um ciclo. A Figura 17 resume e ilustra esse ciclo com os principais elementos trabalhados pela TCC.

A duração ideal de um tratamento baseado na TCC varia entre os indivíduos. Para a depressão leve a moderada, o uso da TCC de dezesseis a vinte sessões, ao longo de 3 a 4 meses, é o habitual (NICE, 2016).

Figura 17 – Ciclo dos Principais Elementos da TCC.



Fonte: Baseado em Hillman (2014).

A adolescência, período entre 12 e 18 anos de idade (REPÚBLICA, 1990), é sem dúvida uma das fases mais desafiadoras da vida. Nesse período de transição, o adolescente se depara com muitos desafios que podem causar distúrbios emocionais e, para vários deles, é aconselhável a busca por ajuda psicológica. Os adolescentes apresentam características de desenvolvimento específicos, em que o estabelecimento de interações com os pares assume uma relevância crescente. Essa é uma fase propícia ao surgimento de preocupações relacionadas com a avaliação e com a forma como o “eu” é percebido pelos outros (SUDHIR, 2015).

Muitos dos problemas emocionais têm início e pico durante essa fase da vida (KOBAK; MUNDT; KENNARD, 2015; SUDHIR, 2015). As intervenções psicológicas em adolescentes enfrentam desafios especiais, muitas vezes devido à natureza dos problemas. Muitos problemas vistos em anos posteriores, e administrados em terapia, podem estar presentes mesmo antes, mas sem serem considerados problemas. A depressão é um dos distúrbios psicológicos mais comuns na adolescência (NARDI et al., 2016).

Há um considerável debate sobre qual a idade mais adequada para participar da TCC. Essa discussão é apoiada, basicamente, pelo fato de a criança ou o adolescente talvez não ter o desenvolvimento cognitivo necessário para participar da TCC. É importante usar uma linguagem apropriada à idade, bem como outros recursos como jogos, fotos e figuras, e narrações de histórias (STALLARD, 2007).

Ajudar o adolescente a identificar e reconhecer o problema que ele enfrenta é importante para o sucesso da terapia. A formulação representa um entendimento compartilhado entre o terapeuta e o adolescente. Da perspectiva do adolescente, essa formulação é o meio pelo qual ele compreende e percebe suas dificuldades. Determinados pensamentos, comportamentos, sintomas e experiências que podem parecer desconectados são reunidos de uma maneira coerente e compreensível. Para o terapeuta, a formulação é utilizada para avaliar o início e o desenvolvimento da terapia, fornecendo o modelo que dirige e informa o conteúdo do programa de tratamento.

A formulação é específica para cada indivíduo, e é feita para conectar a teoria com o plano terapêutico. Um aspecto da formulação é que ela permanece aberta para novas informações e validações durante a terapia (STALLARD, 2007).

3.1.1 Depressão

A depressão é uma doença que aflige o mundo todo, com uma estimativa de 322 milhões de pessoas afetadas. É o que aponta o relatório da Organização Mundial da Saúde (OMS) divulgado no dia 23 de fevereiro de 2017 (BRASIL, 2017). No Brasil, a depressão atinge 11,5 milhões de pessoas. A depressão é caracterizada pelo sentimento de tristeza, perda de interesse, ausência de prazer, oscilações entre sentimentos de culpa e baixa autoestima, além de distúrbios do sono ou do apetite. Sintomas como sensação de cansaço e falta de concentração também podem ser apresentados (FIOCRUZ, 2017).

Vários fatores podem levar à depressão, como questões sociais, psicológicas e biológicas. É importante deixar claro que a depressão é diferente das variações de humor e das respostas emocionais de curta duração com que nos deparamos na vida cotidiana (BRASIL, 2017). Essa doença pode se tornar uma condição de saúde grave, especialmente quando é de longa duração e com intensidade moderada ou severa. Esse distúrbio causa um grande impacto na qualidade de vida do indivíduo, compromete as relações sociais e familiares e, no pior dos casos, pode levar ao suicídio. Cerca de 800 mil pessoas morrem por suicídio a cada ano no mundo. No Brasil, o suicídio é a terceira principal causa de morte entre os jovens (período de 15 a 29 anos de idade) (REPÚBLICA, 2013) (PSIQUIATRIA, 2014).

Embora existam tratamentos eficazes para depressão, menos da metade das pessoas com esse quadro no mundo recebe tais tratamentos. Os obstáculos ao tratamento incluem a falta de recursos, a falta de profissionais capacitados e o estigma social associado aos transtornos mentais (BRASIL, 2017). O estigma, particularmente em torno de transtornos mentais, faz com que muitas pessoas que estão pensando em tirar suas próprias vidas, ou que já tentaram suicídio, não procurem ajuda (BICKMORE et al., 2010).

O comportamento suicida constitui a tendência autodestrutiva mais extrema. É um comportamento não adaptativo, que está submetido a vários fatores, e pode se apresentar num gradiente de gravidade que varia da ideação suicida ao suicídio consumado. A adolescência é um período crítico a comportamentos agressivos. É uma fase de mudanças que envolve vários aspectos: sociais, familiares, físicos e afetivos. Essas mudanças, embora normais, fazem com que o jovem experimente níveis crescentes de ansiedade e angústia, assim aumentando o risco da ocorrência de problemas emocionais, entre os quais, sintomas depressivos e ideação suicida parecem estar entre os mais preocupantes (WERLANG; BORGES; FENSTERSEIFER, 2014).

Existem métodos eficazes para o tratamento da depressão. Profissionais de saúde podem oferecer tratamentos psicológicos, como ativação comportamental, TCC e psicoterapia interpessoal, ou medicamentos antidepressivos. Os medicamentos podem ser eficazes

para o tratamento da depressão moderada a grave, mas não são indicados para os casos mais brandos da doença. Os antidepressivos não devem ser usados para tratar a depressão em crianças, e não são recomendados como a primeira forma de intervenção da doença em adolescentes (BRASIL, 2017).

3.1.2 Terapia Cognitivo-Comportamental via Computador

A TCC também pode ser realizada através do computador. TCC via Computador (ou, do inglês, *Computerised Cognitive Behavioural Therapy* (CCBT), ou ainda *Internet-delivered Cognitive Behavioural Therapy* (ICBT)) é um termo genérico que é utilizado para se referir a vários métodos de uso da TCC. Ela pode ser disponibilizada através de um computador pessoal, através da Internet ou de telefone usando sistemas de Resposta de Voz Interativa (em inglês, *Interactive Voice Response* (IVR)) (NICE, 2013). A TCC via Computador tem um potencial para melhorar o acesso à terapia, que muitas vezes não está ao alcance de pessoas que necessitam de tratamento.

Em fevereiro de 2006, o NICE (*National Institute for Health and Care Excellence*, em português, Instituto Nacional para Saúde e Cuidados de Excelência), estabelecido na Inglaterra, mas com acordos para fornecer produtos e serviços a outros países do Reino Unido, recomenda o uso da TCC via Computador no NHS (*National Health Service*, em português, Serviço Nacional de Saúde) para pacientes com depressão leve a moderada como primeira opção de tratamento, ao invés de adotar inicialmente o uso de um antidepressivo.

A disponibilidade de sistemas de TCC permite uma maior flexibilidade no tratamento, especialmente para as pessoas que não querem ou que não são adequadas para terapia medicamentosa, ou ainda que não desejam interagir com um terapeuta (NICE, 2013). Algumas pesquisas identificaram vantagens relevantes para a comunicação mediada por computador em comparação com a interação humana face a face. Por exemplo: na promoção de um sentimento de anonimato e no aumento da auto-revelação (JOINSON, 2001).

A TCC via computador também pode ser usada para dar suporte às sessões de terapia. O uso da tecnologia pode variar de mínimo, em que é considerado um adjunto para a terapia, ou extenso, no qual o computador substitui completamente o terapeuta (EELLS et al., 2015). O uso de TCC via Computador inclui programas multimídia, realidade virtual, interfaces conversacionais e dispositivos portáteis.

Vários *softwares* para terapia baseados na TCC estão disponíveis na literatura. *Beating the Blues* (BTB, 2017) e *MoodGYM* (UNIVERSITY, 2001), são exemplos de *softwares* que usam a TCC via computador, e que não precisam de um terapeuta para acompanhar o paciente. Em contraste, o *Good Days Ahead* (WRIGHT; WRIGHT; BECK, 2017) foi concebido para servir como um adjuvante ao tratamento de tal modo que o programa de computador e o terapeuta formam uma equipe de trabalho. Em Richardson, Stallard e Velleman (2010) foi relatado um estudo com seis diferentes sistemas que usam TCC via computador, entre

eles estão o *MoodGYM* e o *Beating the Blues*. Os estudos relataram reduções nos sintomas clínicos e também melhorias em variáveis como comportamento, auto-estima e cognição. A satisfação dos pacientes com o tratamento foi de média a alta, embora os níveis de abandono e não conclusão são elevados.

Em nossas pesquisas encontramos alguns *chatbots* que usam a TCC com foco em transtornos psicológicos, como a depressão. Por exemplo, o Help4mood¹ é um ACI que usa a TCC para conversar com as pessoas que têm depressão grave. O *chatbot Overcoming Depression* ajuda o usuário a aprender técnicas da TCC através dos diálogos (COLBY, 1995). Nas seções seguintes, abordaremos mais sobre os *softwares* e ACs que podem ajudar pessoas com transtornos mentais.

Mas não podemos falar só de vantagens, existem também algumas limitações e barreiras no uso da TCC via computador. Por exemplo, não existe a ligação humana que é oferecida em uma sessão com o terapeuta, alguns estudos afirmam que essa ligação genuína com o terapeuta é importante para o sucesso do tratamento (EELLS et al., 2015); os programas de computador precisam ser empolgantes para cativar o cliente, pois a taxa de abandono do tratamento é alta; podem depender da adoção do terapeuta para uso da ferramenta; pode haver resistência dos terapeutas sobre a validade terapêutica do programa; havendo ainda problemas legais e éticos sobre os dados dos pacientes.

3.1.3 Ética e a Psicoterapia Mediada por Computador

A psicoterapia mediada pelo computador através da Internet vem sendo realizada nos Estados Unidos e na Europa de forma comercial, mas no Brasil ela só é permitida em caráter experimental (SILVA; KOTUJANSKY, 2017). De acordo com a resolução do Conselho Federal de Psicologia (CFP) N° 011/2012, alguns serviços psicológicos mediados por computador, como a orientação psicológica, podem ser oferecidos por profissionais de Psicologia, desde que o Código de Ética Profissional do Psicólogo não seja violado e que os serviços sejam pontuais (PSICOLOGIA, 2012).

Os serviços de Psicologia que podem ser realizados via computador são: processo prévio de seleção de pessoal; aplicação de testes devidamente regulamentados pela resolução pertinente; supervisão do trabalho de psicólogo, realizado de forma eventual ou complementar ao processo de sua formação profissional presencial; e atendimento eventual de cliente em trânsito e/ou de clientes que momentaneamente se encontram impossibilitados de comparecer ao atendimento presencial. Esses serviços são limitados ao número máximo de vinte 20 atendimentos (PSICOLOGIA, 2012).

A orientação psicológica online não é psicoterapia e não deve substituir o atendimento psicológico presencial. Ela tem por objetivo trabalhar questões pontuais dos pacientes. O atendimento psicoterapêutico mediado por computador no Brasil é somente permitido

¹ <<http://help4mood.info/site/default.aspx>>

em caráter exclusivamente experimental e é necessário uma autorização do CFP (PSICOLOGIA, 2012).

3.2 Sistemas Computacionais Baseados em TCC

Existem alguns sistemas disponíveis na Internet que empregam a TCC para tratar transtornos psicológicos. Apresentaremos três deles a seguir, que consideramos os mais importantes.

3.2.1 MoodGYM

MoodGYM é um programa interativo de auto-ajuda que está disponível de forma gratuita na Internet (CHRISTENSEN; GRIFFITHS, 2017). Foi criado pelo Centro de Pesquisa em Saúde Mental da Universidade Nacional da Austrália e é baseado na TCC para tratar a ansiedade e a depressão. Ele está disponível em seis idiomas, mas infelizmente não na língua portuguesa (YELLOWLEES; HILTY; MUCIC, 2016). Utiliza uma estrutura de cinco módulos. Os módulos foram criados para seguir uma ordem definida, e cobrem tópicos como: sentimentos, pensamentos, reestruturação, relaxamento e relacionamentos. Este formato estruturado é uma maneira de ensinar e garantir que os usuários recebam conteúdos relacionados às técnicas relevantes de mudança de comportamento (SCHUELLER; STILES-SHIELDS; YAROSH, 2017).

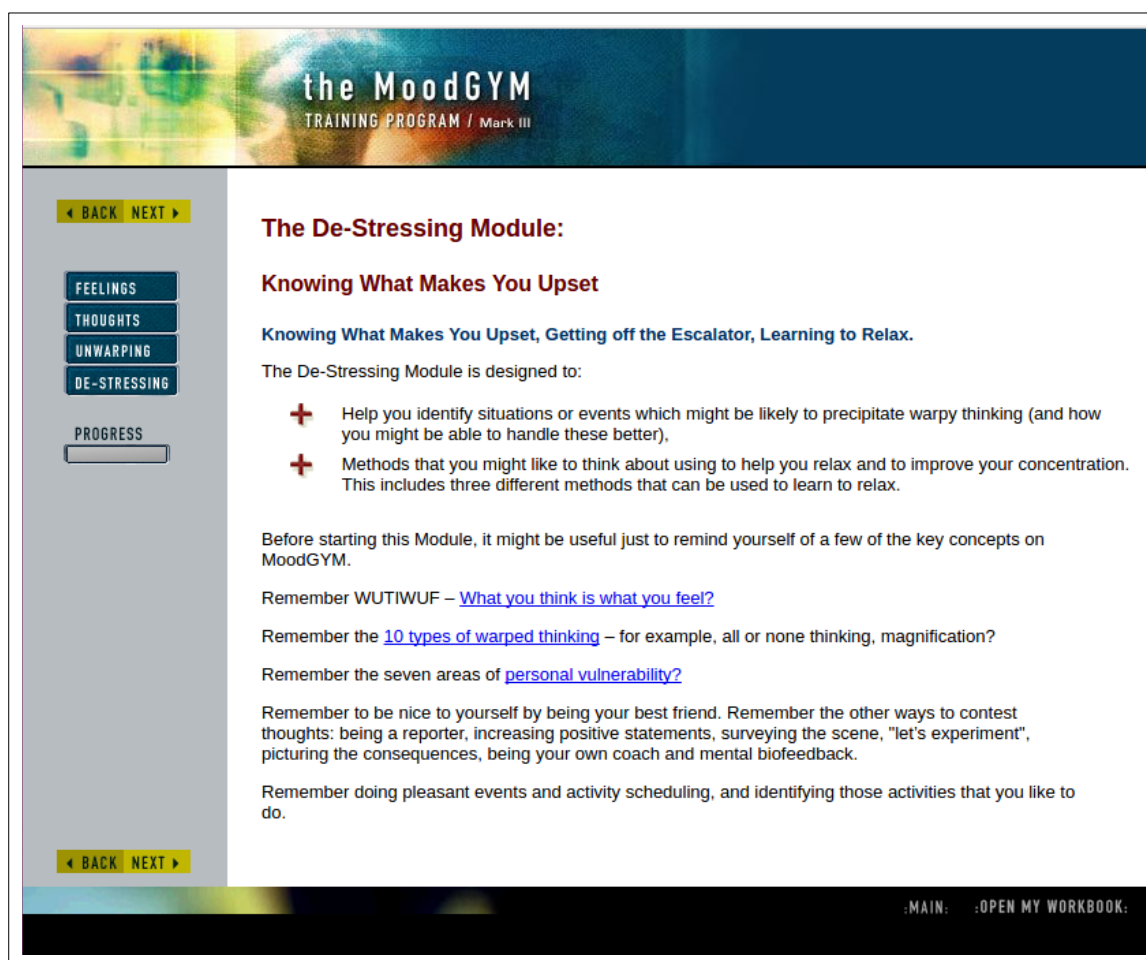
A Figura 18 mostra o exemplo de uma das telas do sistema MoodGYM.

O MoodGYM usa diagramas e exercícios para ensinar elementos da TCC, bem como técnicas de relaxamento e meditação. Tem se mostrado um tratamento eficaz para a redução de sintomas da depressão (CAI et al., 2015; CLARKE; KUOSMANEN; BARRY, 2015). Estudos evidenciaram que MoodGYM reduziu significativamente sintomas de depressão dos seus usuários.

MoodGYM é utilizado pelos usuários sem a ajuda de um terapeuta. Ele não personaliza a formulação para os usuários individualmente. Todas as pessoas que usam o programa parecem receber essencialmente a mesma intervenção. Isto é teoricamente problemático, sendo inconsistente com a ênfase nas formulações individuais, que é o cerne dos programas tradicionais de TCC.

Uma abordagem alternativa para o MoodGYM realizar a formulação de caso de forma individual para cada paciente seria usar dados clínicos, auditados de arquivos de pacientes anteriores para gerar grandes listas de pensamentos, crenças e comportamentos poucos úteis. Essas listas permitiriam que um programa de computador autônomo adaptasse respostas aos problemas individuais de cada usuário (HELGADÓTTIR et al., 2009).

Figura 18 – Tela do Sistema MoodGYM.



Fonte: (CHRISTENSEN; GRIFFITHS, 2017).

3.2.2 Beating the Blues

Beating the Blues é um sistema para o tratamento de pessoas com estresse, ansiedade e depressão (BTB, 2017). Baseado na TCC, esse sistema é composto de oito sessões que duram cerca de uma hora, e tenta ajudar o usuário a compreender a ligação existente entre o pensamento e como ele influencia os sentimentos e comportamentos. O sistema ensina estratégias e habilidades para a vida. Ele foi criado para ajudar o usuário, em um curto prazo, a enfrentar o futuro com confiança. Esse sistema está disponível na Internet, mas não é gratuito. Pode ser usado pelo usuário sozinho, como também comporta o acompanhamento de um terapeuta (BTB, 2017).

A primeira sessão do tratamento começa com a identificação de problemas, e busca o engajamento do usuário em atividades prazerosas. A segunda sessão usa as atividades prazerosas para introdução dos pensamentos automáticos. Cada sessão oferece conselhos e *feedback*. São utilizadas quatro personagens, uma mulher mais velha, um homem de meia-idade, um homem solteiro e uma mãe trabalhadora. Ao longo do tratamento essas

personagens são apresentadas através de quadros onde são fornecidas informações sobre a natureza da depressão e como tratar os sintomas usando estratégias e técnicas da TCC. Métodos interativos incorporados no programa permitem aos pacientes praticar os princípios e técnicas durante a sessão, bem como através de trabalhos de casa após a sessão (EELLS et al., 2014).

Na Figura 19 temos a tela inicial da página da Internet do *Beating the Blues*.

Figura 19 – Tela Inicial *Beating the Blues*.



Fonte: (BTB, 2017).

Desde fevereiro de 2006, o National Institute for Health and Care Excellence (NICE) recomenda o Beating the Blues como uma opção de tratamento para as pessoas diagnosticadas com depressão leve ou moderada no National Health Service (NHS) (CAI et al., 2015). O NICE também publicou estudos que demonstram a efetividade do uso do sistema para a melhoria da depressão (NICE, 2013).

3.2.3 Good Days Ahead

Good Days Ahead é um sistema interativo que ensina os indivíduos como aplicar a TCC para superar a depressão e a ansiedade. Os profissionais especializados também podem acessar o programa para monitorar o uso do sistema pelos seus clientes (WRIGHT; WRIGHT; BECK, 2017).

O sistema oferece uma variedade de mídias interativas, incluindo vídeos e uma série de exercícios práticos em cada sessão para ajudar os usuários a praticar os conceitos da TCC em suas próprias vidas (WRIGHT; WRIGHT; BECK, 2017).

Esse sistema foi projetado como um adjunto à terapia. Ele pode ser acessado através da Internet, e consiste em nove lições que abrangem os passos psico-educacionais e de ação prescritos através da TCC. As lições abrangem os princípios básicos da TCC, como:

- Identificar, reconhecer e modificar os pensamentos automáticos;
- A ligação entre o pensamento e a ação;
- Identificação e modificação de esquemas; e
- Reconhecer e corrigir erros cognitivos.

A Figura 20 mostra um exemplo de tela do sistema *Good Days Ahead*.

Figura 20 – Tela do Sistema *Good Days Ahead*.

The screenshot displays the 'Good Days Ahead' web application interface. At the top, there is a navigation bar with the logo 'good days ahead by empower' and links for 'Home', 'Learn', 'Practice' (highlighted), and 'Progress'. The user is logged in as 'Hi Kara' with links for 'Help' and 'Log Out'. On the left, a sidebar menu lists various practice steps: 'Start', 'Event', 'Automatic Thought', 'Thought Strength', 'Emotions', 'Cognitive Errors', 'Alternative Thoughts', 'Reevaluate', 'Action Plan', and 'Summary'. The main content area is titled 'Thought Change Record (TCR)'. It features a horizontal scale from 0 to 10. Under 'Cognitive Errors', the text reads 'Ignoring the Evidence Jumping to Conclusions'. Under 'Alternative Thoughts', it says 'He's just trying to help'. A 'Strength' bar is shown, filled to approximately 6 on the 0-10 scale. Below this, section '8 Action Plan' is titled 'Create Your Action Plan' with the prompt 'How might you act (or have acted) differently as a result of changing this automatic thought?'. A text input field contains the response 'I'll talk it out with him before it gets'. A 'Continue' button is located at the bottom right of the input area.

Fonte: (WRIGHT; WRIGHT; BECK, 2017).

O programa requer habilidades de digitação mínimas e foi feito para pessoas que estejam a partir do nono ano da escola. Os usuários recebem instruções de acesso do terapeuta e, uma vez registrados, podem acessar o programa. O sistema ainda encoraja

e demonstra o valor de fazer as tarefas de casa entre as sessões. Ele permite que os terapeutas acessem os exercícios de auto-ajuda e trabalhos de casa concluídos usando o programa, e mensura os níveis globais de depressão e ansiedade dos pacientes (EELLS et al., 2014).

Um estudo clínico randomizado foi realizado para verificar a eficácia do programa e a redução do tempo do terapeuta necessário para tratar o paciente. Psicólogos distribuíram aleatoriamente um total de 45 pessoas para a realização da TCC padrão, uso do *Good Days Ahead* ou uma lista de espera. Ambos os tratamentos consistiram em nove sessões durante 8 semanas. Para os pacientes que usaram o *Good Days Ahead*, o tempo do terapeuta foi reduzido pela metade após a primeira visita. Os testes mostraram que os pacientes tratados com o *Good Days Ahead* e a TCC padrão alcançaram significativa melhoria na gravidade da depressão quando comparados aos pacientes que estavam na lista de espera (EELLS et al., 2014).

3.3 Chatbots Aplicados à Transtornos Psicológicos

Na seção anterior vimos sistemas disponíveis na Internet que utilizam a TCC para ajudar usuários com transtornos psicológicos. Nesta seção, veremos alguns ACs (estamos utilizando o termo Agentes Conversacionais ACs aqui porque os sistemas apresentados nesta seção, não são necessariamente *chatbots*, podendo ser alguma de suas derivações) criados para apoiar os usuários com estresse, ansiedade e depressão. Os ACs apresentados aqui, não obrigatoriamente utilizam a TCC como base para tratar esses transtornos. Nas nossas pesquisas, além dos ACs descritos a seguir, encontramos também os seguintes ACs: Therachat², Joy³, Brisbot⁴ e Ahuna Bot⁵.

3.3.1 Overcoming Depression

Overcoming Depression é um *chatbot*, criado em 1991 por Kenneth M. Colby e seu filho Peter, que usa TCC para conversar com o usuário. Embora não nomeado como um *chatbot* pelo autor, esse sistema funciona em dois modos diferentes, o modo de texto e o de diálogo, que se alternam durante a conversa com o usuário. O modo de texto serve para dar instruções para o usuário. Ele provê fatos, conceitos e explicações sobre a depressão, juntamente com estratégias cognitivas da TCC para aliviar o distúrbio e prevenir sua recorrência. O modo de diálogo permite a conversa com o usuário, e o encoraja a expressar seus pensamentos e sentimentos. *Overcoming Depression* foi projetado para tratar pessoas com depressão leve a moderada. O autor deixa claro que o sistema não foi feito para

² <<https://therachat.io/>>

³ <<http://www.hellojoy.ai/>>

⁴ <<http://brisbot.com/>>

⁵ <<http://ahunabot.com/>>

simular um psicoterapeuta, como o *chatbot* ELIZA, mas sim para prover uma nova forma de ajuda terapêutica. Os modos de texto e diálogo do *chatbot* são empregados ao longo de sete sessões (COLBY, 1995).

Na Figura 21, temos um trecho de uma conversa com o *chatbot* *Overcoming Depression* traduzida para o português.

Figura 21 – Trecho de Conversa com o *chatbot* *Overcoming Depression*.

PROGRAM: O que, em particular, o deprime?
 VOCÊ: Manhãs. Não consigo sair da cama de manhã.
 PROGRAM: Deve ser difícil sair da cama quando não há um objetivo.
 VOCÊ: Verdade. Eu odeio a ideia de ir trabalhar.
 PROGRAM: O que, em particular, você não gosta no seu trabalho?
 VOCÊ: Eu tenho um trabalho sem futuro. Não sei como posso mudar isso.
 PROGRAM: O que faz você sentir que não há nada o que fazer sobre essa situação?
 VOCÊ: Eu me sinto impotente para mudar qualquer coisa.
 PROGRAM: Alguns dos seus sentimentos de desesperança provêm de um equívoco de que não há nada que você possa fazer para melhorar sua condição.

Fonte: Traduzido de Colby (1999).

O *chatbot* funciona através de casamento de padrão, e faz associações das entradas do usuário com os conceitos existente na base de conhecimento para que a conversa possa fluir. O sistema ainda faz um tratamento no texto para retirar palavras não importantes como, artigos, pontuações e vários advérbios (COLBY, 1999).

Existem duas versões do sistema, uma versão para o usuário doméstico e outra para profissionais de saúde mental, ambas pagas. Muitas pessoas com depressão usaram tanto a versão doméstica quanto a profissional. Os métodos de tratamento podem ser avaliados em termos de segurança, eficácia, eficiência, economia, durabilidade dos resultados e aceitabilidade. Em estudos empregado pelo autor, o *chatbot* recebeu uma taxa de satisfação de 96%, em uma amostra de 142 usuários. Não existem dados de melhoria da recaída de pacientes depois de usar o programa. Em uma terapia convencional com um terapeuta e medicação, a taxa de recaída é de 20% a 30% um ano depois que o tratamento termina. Segundo o autor, depois desse tempo a pessoa que usou o programa pode ainda continuar usando (COLBY, 1995).

3.3.2 TeenChat

TeenChat é um *chatbot* criado na língua chinesa, cujo objetivo é ajudar o adolescente a aliviar o estresse. O estresse pode acarretar problemas físicos e psicológicos, incluindo ansiedade, depressão e até mesmo o suicídio. O *chatbot* interage com os adolescentes como se fosse um amigo virtual (HUANG et al., 2015). A Figura 22 traz um exemplo da tela do TeenChat.

Figura 22 – Tela do *Chatbot* TeenChat.

Fonte: (HUANG et al., 2015).

Não é descrito no artigo se é utilizada alguma teoria da psicologia para embasar os diálogos do *chatbot*, apenas é relatado a aplicação de um questionário chamado *Cohen's Perceived Stress Scale* (CPSS-14) para detectar o estresse do adolescente (COHEN; KAMARCK; MERMELSTEIN, 1983). O TeenChat funciona construindo uma Árvore de Dependência Linguística para analisar a correlação das palavras da entrada do usuário e sua base de conhecimento. Para a construção dessa árvore, é usada uma ferramenta chamada de LTP (*Language Technology Platform*, em português, Plataforma Tecnológica de Idiomas) (CHE; LI; LIU, 2010). Durante o diálogo, o *chatbot* procura em sua base de conhecimento se existe uma classificação cadastrada para o tema de predominância do estresse do adolescente, que pode ser, por exemplo: os estudos, as relações interpessoais e a auto-estima. Se houver uma correspondência, o TeenChat conduzirá a conversa com foco no tema dessa classificação. Caso não seja encontrada uma classificação adequada, é acessada uma base online chamada de Baidu Knows⁶, ou o repositório existente do *chatbot* Simsimi⁷ para tentar achar uma resposta apropriada. Se for encontrado um bom grau de similaridade para a entrada do usuário, então a resposta é recuperada e apresentada ao usuário. Se mesmo assim não for encontrado algo para dizer para o usuário, então uma resposta motivacional aleatória é fornecida, ou é feito um tipo de brincadeira.

Foi realizado um experimento por um mês para avaliar o *chatbot*. Nesse experimento,

⁶ <<https://zhidao.baidu.com/>>

⁷ <<http://www.simsimi.com/>>

as respostas apresentadas pelo *chatbot* obtiveram uma precisão de 78,34% e uma cobertura de 76,72%. Os criadores do TeenChat pretendem, futuramente, usar recursos das redes sociais para melhorar a precisão das respostas. Um problema existente no TeenChat é relativo à privacidade das conversas com o usuário (HUANG et al., 2015).

3.3.3 Help4Mood

Help4Mood é um sistema desenvolvido por um consórcio formado entre a Escócia, a Espanha, a Romênia e a Itália (BURTON et al., 2012). Esse sistema é interativo e possui, entre os seus componentes, um ACI que é utilizado como o meio de comunicação com o usuário. O Help4Mood foi criado para ajudar no acompanhamento de pacientes que estão em tratamento de depressão grave. Ele é usado para monitorar o humor, os pensamentos negativos, as atividades diárias e a fala dos usuários. Através desses monitoramentos, o sistema pode gerar informações tanto para o paciente quanto para o profissional que o acompanha. Além disso, esse sistema encoraja os pacientes a refletir sobre os pensamentos negativos usando elementos da TCC. As atividades diárias e a fala do usuário são monitoradas através de um acelerômetro (que pode ser utilizado no pulso ou no bolso) e análise acústica da fala, respectivamente (BURTON et al., 2016).

Na análise acústica da fala, o ACI pede para que o usuário fale algumas coisas, por exemplo: contar de um até dez e depois de dez até um. A fala das pessoas com depressão frequentemente mostra sinais claros de sua condição, como: entonação, fala lenta e pausas longas (WOLTERS et al., 2015). Na Figura 23, temos o exemplo de uma das telas do ACI Help4Mood.

O Help4Mood consiste em três componentes principais (WOLTERS et al., 2012):

1. Um Sistema de Monitoramento Pessoal (do inglês, *Personal Monitoring System (PMS)*) que rastreia as atividades e o sono;
2. Um ACI que interage com o paciente, aplica questionários, coleta dados psicomotores e de humor, oferece exercícios e usa elementos da TCC para ensinar ao usuário como melhorar. O sistema usa recursos de NLG para gerar as sentenças que serão faladas pelo agente. Além disso, esse ACI é capaz de dialogar em três idiomas: Inglês, Romeno ou Espanhol;
3. Um Sistema de Suporte à Decisão (do inglês, *Decision Support System (DSS)*) que interpreta os dados recebidos, planeja as sessões do ACI com o usuário, e resume os resultados para ele e o profissional que o está acompanhando. O DSS pode gerar sessões curtas, médias e longas, e o usuário seleciona uma dessas opções cada vez que usa o sistema.

Os usuários podem configurar ainda a aparência do sistema, isso inclui o estilo de interação (fala e/ou texto) e a aparência do ACI. O Help4Mood oferece quatro tipos de

Figura 23 – Exemplo de Tela do Help4Mood.

Hi John,
how are you feeling today?

From my reports it seems
that you did not sleep very
well last night!

I would like you to answer
to a simple questionnaire!

Over the last 2 weeks, how often have you been bothered by any of the following problems?

	Not to all	Several days	More than half the days	Nearly every day
1. Little interest or pleasure in doing things				
2. Feeling down, depressed, or hopeless				
3. Trouble falling or staying asleep, or sleeping too much				
4. Feeling tired or having little energy				
5. Poor appetite or overeating				
6. Feeling bad about yourself, or that you are a failure, or have let yourself or your family down				
7. Trouble concentrating on things, such as reading the newspaper or watching television				
8. Moving or speaking so slowly that other people could have noticed. Or the opposite, being so fidgety or restless that you have been moving around a lot more than usual				
9. Thoughts that you would be better off dead, or of hurting yourself in some way				

Confirm

Reminders Reports Customization Log out help 4 mood

Fonte: (HELP4MOOD, 2017).

avatares, dois masculinos e dois femininos. Cada avatar está disponível em dois estilos de vestuário, formal ou informal, para combinar com o estilo de interação desejado (WOLTERS et al., 2012).

Como o Help4Mood é um adjunto do tratamento psicológico, ele não está disponível livremente para os usuários. O Help4Mood está disponível para *desktop*, mas há planos para disponibilizá-lo através de *tablets* (BURTON et al., 2016).

Foi realizado um estudo com 28 participantes ao longo de 4 semanas, 14 com terapia normal e 14 com Help4Mood. Três participantes do grupo que estavam utilizando o Help4Mood foram desclassificados por alguns critérios adotados ou não completaram a pesquisa. Os resultados globais da pesquisa apenas indicaram uma pequena melhoria na média dos escores BDI-2 (questionário, criado por Aaron T. Beck, com 21 itens de múltipla escolha para medir a severidade da depressão)⁸ (BURTON et al., 2016).

O Help4Mood apenas utiliza o reconhecimento de voz para realizar a análise acústica da fala, ele não utiliza o reconhecimento como recurso de comunicação entre usuário e o sistema. Na verdade, o usuário se comunica com o ACI através de texto e formulários.

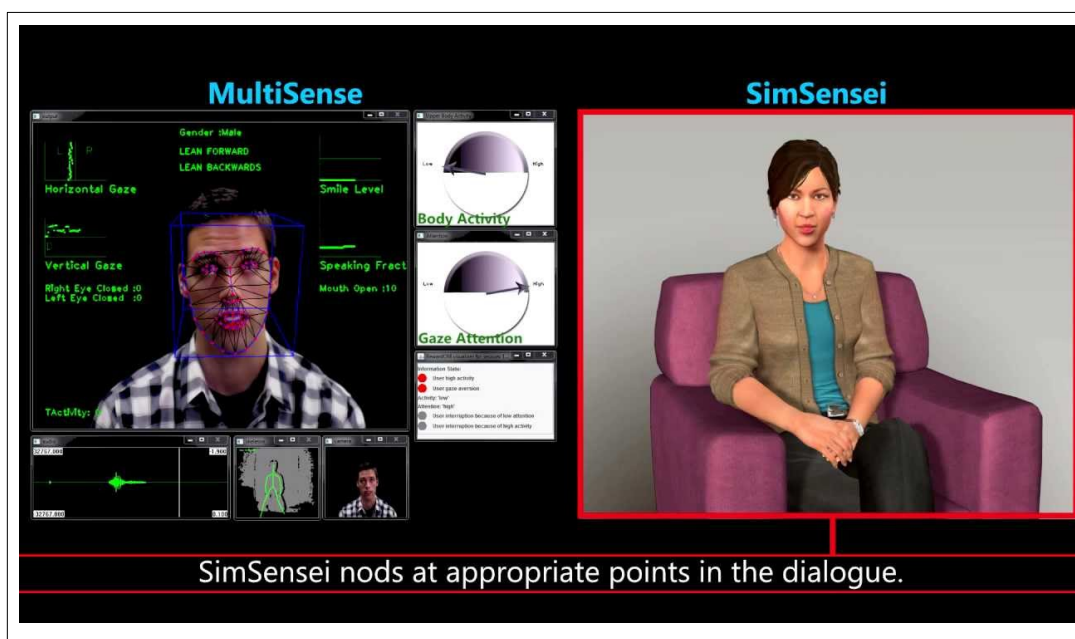
⁸ <https://pt.wikipedia.org/wiki/Escala_de_depress~ao_de_Beck>

3.3.4 Ellie

Ellie é um Humano Virtual desenvolvido pelo Instituto para Tecnologias Criativas (do inglês, *Institute for Creative Technologies (ICT)*) da Universidade do Sul da Califórnia (RIZZO; MORENCY, 2011). Ellie tem a capacidade de realizar entrevistas semi-estruturadas com o usuário com o objetivo de perceber automaticamente indicadores de angústia que são definidos através de comportamentos verbais e não-verbais correlacionados com depressão, ansiedade ou transtorno de estresse pós-traumático (DEVAULT et al., 2014). Ellie foi criada através do Simsensei Kiosk (DEVAULT et al., 2014), um *framework* para realização de entrevistas através de humanos virtuais, e projetado para criar uma interação face a face. Esse Humano Virtual usa técnicas de PLN e Aprendizagem de Máquina para processar a entrada do usuário.

O objetivo do *framework* Simsensei Kiosk é disponibilizar uma ferramenta de apoio à decisão clínica, fornecendo aos profissionais de saúde medidas objetivas do usuário. Os indicadores percebidos por Ellie podem ajudar o profissional de saúde a fazer um diagnóstico do paciente mais acurado. Quando Ellie conversa com o usuário por um longo período, os indicadores podem ser comparados com os dados da semana ou do mês anterior, permitindo que os profissionais de saúde detectem se uma mudança está ocorrendo (DEVAULT et al., 2014). Alguns resultados obtidos sugerem que os humanos virtuais podem reduzir o estresse e o medo associados à percepção de serem julgados e, assim, diminuir as barreiras emocionais à divulgação de informações. A Figura 24 mostra um exemplo de uma interação de Ellie com um usuário.

Figura 24 – Exemplo de Interação de Ellie.



Fonte: (ROBINSON, 2015).

Como base para o sistema de percepção, o Simsensei Kiosk utiliza o *framework* Multi-Sense (STRATOU; MORENCY, 2017), que é um sistema flexível para detecção multimodal (áudio e vídeo) em tempo real. Ellie tem a capacidade de analisar os comportamentos não-verbais do usuário através de uma câmera. Como exemplo de um comportamento não-verbal, podemos citar a intensidade do sorriso. O sorriso é um comportamento importante que tem sido vinculado a investigações de depressão. Já como fatores associados à fala, podemos citar como exemplos: a diminuição do tom da voz e longas pausas silenciosas, bem como diferenças em certas frequências lexicais, incluindo o uso de pronomes de primeira pessoa e palavras negativas (DEVAULT et al., 2014).

Um dos criadores de Ellie, o professor Louis-Philippe Morency, diz que esse Humano Virtual não substitui uma terapia com um profissional real. Ele é uma ferramenta de apoio à decisão projetada para coletar informações. O diagnóstico e as decisões de tratamento ainda são feitos por profissionais humanos (ROBINSON, 2015).

Foram realizados três estudos com um total de 351 participantes. Desses, 120 foram tratados com psicólogos reais, 140 com a utilização de Ellie controlada por humanos e 91 com Ellie de forma autônoma. No final dos estudos, foi submetido um questionário de usabilidade do sistema com várias perguntas para os participantes que utilizaram a versão controlada e a versão autônoma de Ellie. Exemplos de perguntas realizadas: *Eu me senti confortável em compartilhar informações com Ellie?*; *Eu compartilhei várias informações pessoais com Ellie?*; *Eu recomendaria Ellie a um amigo?*. A versão controlada de Ellie teve um total de 80,71% de satisfação, e a versão autônoma, 75,43% (DEVAULT et al., 2014).

Ellie possui apenas um pequeno conjunto de perguntas e respostas, e foram reportados pelos participantes da pesquisa alguns erros de expressões corporais não condizentes com a fala. Não é informado nos artigos se é utilizada alguma teoria da psicologia para estruturar os diálogos de Ellie (DEVAULT et al., 2014).

3.3.5 Wysa

O *chatbot* Wysa⁹ (*Wise young smile assistant*) tem a forma de um pinguim, e pode conversar sobre os sentimentos, orientar sobre meditação ou exercícios, como a realização de alongamentos. Wysa pode ensinar técnicas de meditação *Mindfulness*, técnicas da TCC e de pensamentos positivos. A seleção do conteúdo apresentado ao usuário é determinada durante as conversas (MCINTYRE, 2016).

Wysa é capaz de avaliar a melhoria do usuário ao conversar com ele, e quando o usuário faz as tarefas solicitadas. No final de cada semana, o *chatbot* faz um resumo das emoções e hábitos do usuário. Se Wysa considerar que o usuário está se sentindo triste, o *chatbot* poderá executar um teste para detecção de depressão. Se o teste indicar que o usuário está

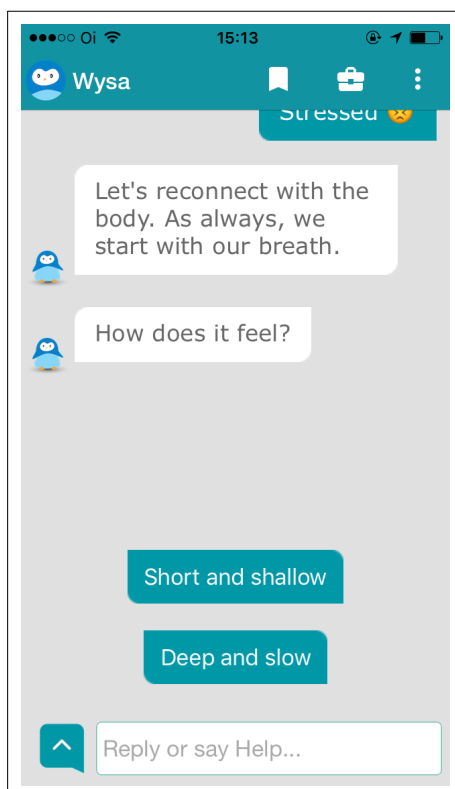
⁹ <<http://www.wysa.io/>>

deprimido, Wysa recomendará que ele consulte um profissional especializado. O *chatbot* interage através de linguagem natural, de figuras e de vídeos, que podem mostrar exercícios de relaxamento, ou técnicas de como fazer meditação (KAR, 2017).

Wysa foi desenvolvida pela Touchkin¹⁰ e está disponível gratuitamente nas plataformas de dispositivos móveis, como: IOS¹¹, Android¹², Messenger, e futuramente para o Slack¹³ (TOUCHKIN, 2016).

A Figura 25 mostra uma tela do *chatbot* Wysa da versão para a plataforma IOS.

Figura 25 – Exemplo Tela do *Chatbot* Wysa Versão IOS.



Fonte: (TOUCHKIN, 2016).

Wysa possui um vocabulário limitado. Para superar um pouco essa limitação, o *chatbot* usa vários formulários para que o usuário responda às perguntas, como também sugere algumas palavras-chave para que o usuário possa acionar algumas funcionalidades.

As falas do *chatbot* são rápidas, e como a tela de alguns dispositivos móveis são pequenas, a mensagem logo desaparece da tela.

Wysa não reconhece repetições de frases escritas pelo usuário. Cumprimos mais de uma vez o *chatbot*, e ele respondeu como se a sessão tivesse acabado de iniciar.

¹⁰ <<http://www.touchkin.com/>>

¹¹ <<https://www.apple.com/br/ios/ios-10/>>

¹² <https://www.android.com/intl/pt-BR_br/>

¹³ <<https://slack.com/>>

3.4 Considerações Finais

A Tabela 3 apresenta comparações feitas entre algumas características dos sistemas e dos *chatbots* vistos neste capítulo com relação ao *chatbot* Beck desenvolvido neste trabalho de mestrado. Embora o *chatbot* Beck seja apresentado com detalhes somente no próximo capítulo, já revelamos aqui alguns dos seus recursos. As características que não conseguimos identificar nos sistemas e ACs vistos neste capítulo, através dos documentos pesquisados, foram representadas com o sinal de interrogação.

Na Tabela 3 temos as seguintes colunas:

- **Programa** - Onde relacionamos todos os sistemas e ACs vistos;
- **Tipo** - Identifica o tipo do programa, se é um *Chatbot*, um Sistema, um ACI ou um Humano Virtual;
- **TCC** - Indica se o programa usa ou não a TCC;
- **Memória** - Caracteriza se o sistema é capaz de lembrar informações das conversas do usuário ou, no caso de sistemas, os módulos que o usuário já realizou;
- **Gratuito** - Descreve se o programa é gratuito ou não;
- **Idiomas** - Informa os idiomas em que estão disponíveis; e
- **PLN** - Identifica algumas técnicas de PLN utilizadas em seu funcionamento. As técnicas listadas são as que foram descritas em artigos sobre o sistema.

Neste capítulo, apresentamos a TCC que é um tipo de psicoterapia que ajuda as pessoas a entender, como os pensamentos automáticos podem influenciar as emoções e os comportamentos. Vários estudos comprovaram a eficácia dessa terapia no tratamento da depressão. Vimos que a adolescência é uma fase da vida na qual o indivíduo pode se deparar com muitos desafios, e estes podem causar distúrbios emocionais. A TCC é particularmente interessante, porque ela pode ser utilizada através do computador, conforme visto através dos sistemas e *chatbots* apresentados neste capítulo. Os sistemas e *chatbots* abordados, foram criados para apoiar o tratamento de transtornos psicológicos (e.g., ansiedade, estresse, depressão). Finalizamos o capítulo com uma tabela comparativa de alguns recursos oferecidos pelos sistemas e *chatbots* apresentados e comparamos com os recursos do *chatbot* Beck desenvolvido neste trabalho.

Tabela 3 – Tabela Comparativa dos Sistemas e dos *Chatbots*.

Programa	Tipo	TCC	Memória	Gratuito	Idiomas	PLN
MoodGym	Sistema	Sim	Sim	Não	Inglês, Norueguês, Holandês, Chinês, Finlandês e Alemão	Não
Beating the Blues	Sistema	Sim	Sim	Não	Inglês	Não
Good Days Ahead	Sistema	Sim	Sim	Não	Inglês	Não
Over. Depression	<i>Chatbot</i>	Sim	?	Não	Inglês	<i>Stopword</i> , Rule-based
TeenChat	<i>Chatbot</i>	Não	Sim	?	Chinês	Árvore de Dependên- cia Linguística
Help4Mood	ACI	Sim	Sim	Não	Inglês, Romeno e Espanhol	STT, TTS e NLG
Ellie	H. Virtual	Não	Sim	Não	Inglês	STT, TTS e NLU
Wysa	<i>Chatbot</i>	Sim	Não	Sim	Inglês	?
Beck	<i>Chatbot</i>	Sim	Sim	Sim	Português	STT, TTS, Correção Ortográ- fica, <i>Stemming</i> e <i>POS- Tagging</i>

Fonte: Autoria Própria

4 BECK: UM CHATBOT PARA APOIAR ADOLESCENTES COM DEPRESSÃO

Dedicamos este capítulo para apresentar o trabalho desenvolvido nesta pesquisa de mestrado. Esse capítulo está organizado em 7 seções. Na seção 4.1, realizamos uma explanação geral de todos os passos realizados para o desenvolvimento do *chatbot* Beck; na seção 4.2, abordamos sobre os diálogos implementados no *chatbot* Beck; na seção 4.3 apresentamos a arquitetura geral do sistema e explicamos sobre os recursos utilizados para a construção de Beck; na seção 4.4 mostramos como Beck funciona; na seção 4.5 apresentamos a interface implementada, que permite a interação com o usuário; na seção 4.6 descrevemos alguns testes realizados durante o desenvolvimento do *chatbot*; e, por fim, a seção 4.7 traz uma breve conclusão deste capítulo.

4.1 Descrição Geral do Trabalho Realizado

Desenvolvemos, neste trabalho, um *chatbot* para conversar com adolescentes de 15 à 18 anos de idade que têm depressão, ou que gostariam de adquirir conhecimentos para evitá-la. É importante dizer que os diálogos elaborados para Beck não são exclusivamente para adolescentes com quadro depressivo. Como já mencionado (seção 3.1), a TCC pode ser aplicada para vários transtornos psicológicos. Dessa forma, os diálogos realizados por Beck podem ser utilizados para conversar com pessoas com outros tipos de transtornos.

Os diálogos elaborados para o *chatbot* foram baseados na TCC e estruturados a partir da TAC. Esse *chatbot*, batizado de Beck, busca mostrar aos usuários que os pensamentos automáticos negativos que eles têm podem influenciar os seus sentimentos e os seus comportamentos. Embora o *chatbot* Beck use a TCC para conversar com os adolescentes, ele é um sistema de autoajuda e não de psicoterapia. Em vista disso, foi decidido juntamente com uma psicóloga que Beck não usará o termo TCC durante suas conversas com os adolescentes. Beck não aplica nenhum teste psicológico e não faz diagnósticos dos usuários acerca dos seus problemas psicológicos. Beck, atualmente, dispõe de uma única sessão de conversa com o adolescente. Porém, a base de diálogos pode ser estendida.

Beck, um *chatbot* feminino, foi implementada através de um servidor *Web*, e está disponível na Internet, através do endereço ***beck-bot.cin.ufpe.br***, visando facilitar o seu acesso. Para o desenvolvimento de Beck, utilizamos o *ChatScript*, uma linguagem de desenvolvimento de *chatbot* baseada em PLN. O *ChatScript* permite a criação de *chatbots* para qualquer língua humana (e.g., inglês, alemão, japonês, espanhol), no entanto, alguns dos recursos que a linguagem oferece só funciona para o inglês (e.g., dicionário, corretor ortográfico, *POS-Tagging*). Para utilizar alguns desses recursos na língua portuguesa,

precisamos realizar algumas adaptações. Como exemplo, o acoplamento de ferramentas externas de Correção Ortográfica, de *Stemming* e de *POS-Tagging*.

Para guardar algumas informações dos usuários, integramos o *ChatScript* com um Banco de dados. Assim, buscando facilitar o armazenamento e a recuperação de informações.

Beck é capaz de se comunicar com o usuário através de mensagem de texto e voz (síntese e reconhecimento de voz), porém o recurso de voz só está disponível para o Navegador *Web* Google Chrome¹.

Nas seções a seguir, veremos com mais detalhes todo o trabalho realizado para o desenvolvimento do *chatbot* Beck, desde a elaboração dos diálogos até os testes realizados após implementação do *chatbot*.

4.2 Dialogando com Beck

A primeira etapa para criação de Beck foi a elaboração dos diálogos. Esses diálogos têm o objetivo de abordar a TCC de uma forma simples e adequada à idade dos adolescentes. Esses diálogos foram criados a partir de uma pesquisa em artigos e livros sobre a TCC (CLARKE; LEWINSOHN; HOPS, 1990; BECK, 1995; STALLARD, 2007; KNAPP; BECK, 2008; ZUGMAN, 2011; COLLINS-DONNELLY, 2014; NEENAN; DRYDEN, 2014) e posteriormente foram validados com uma psicóloga.

Veremos, nesta seção, os objetivos dos diálogos elaborados para Beck e como eles foram estruturados a partir da TAC. Mostraremos também, cada tópico de diálogo criado, e exemplos de conversação desses tópicos, realizados entre Beck e o usuário.

4.2.1 Objetivos do Diálogo

Preliminarmente, com o objetivo de modelar os diálogos entre Beck e o usuário, foi criado um *script* de diálogos para que o *chatbot* possa conversar com os adolescentes utilizando a TCC. Os diálogos modelados têm os seguintes objetivos:

1. **Usar os princípios da TCC para explicar para os adolescentes sobre a relação do pensamento, da emoção e do comportamento;**
 - Falar dos pensamentos automáticos negativos e como eles podem impactar as emoções e os comportamentos;
 - Explicar que os pensamentos podem influenciar as emoções e os comportamentos;
 - Explicar para o adolescente qual destes elementos é mais fácil de controlar: pensamento, emoção ou comportamento;

¹ <<https://www.google.com/chrome/browser/desktop/index.html>>

- Exibir histórias de personagens para ajudar o adolescente a compreender melhor sobre os pensamentos automáticos negativos; e
 - Mostrar para o adolescente que os pensamentos podem e devem ser desafiados através da busca de evidências que os sustentem.
2. **Incentivar o adolescente a realizar atividades prazerosas (e.g., conversar com amigos, ir ao cinema, jogar futebol);**
- Identificar se o adolescente tem amigos e, caso não tenha, dar dicas de como fazer amigos;
 - Verificar quais são as atividades prazerosas que o adolescente gosta de fazer; e
 - Estimular o adolescente a se engajar em atividades prazerosas e, caso ele não goste de nenhuma atividade, sugerir algumas para ele.
3. **Apresentar para o adolescente alguns erros de pensamento que podemos ter (e.g., catastrofização, raciocínio emocional, rotulação), conforme visto no capítulo 3;**
- Verificar quais pensamentos negativos o adolescente apresenta e quais sentimentos e comportamentos são fruto desses pensamentos; e
 - Mostrar para o adolescente, baseado nos pensamentos descritos por ele, qual é ou quais são os tipos de erros de pensamento que ele apresenta.
4. **Verificar o humor do adolescente a cada conversa.**
- Mostrar para o adolescente, a cada conversa, um formulário para que ele escolha o seu estado de humor; e
 - Buscar avaliar se o adolescente tem apresentado alguma melhora de sessão a sessão, mostrando para ele os resultados.

Os objetivos descritos acima estão implementados na base de conhecimento de Beck através do *script* de diálogos criado. Esse *script* está no Apêndice A deste documento.

4.2.2 Aquisição do Perfil do Usuário

Antes de iniciar a interagir com Beck, o adolescente terá que fazer um cadastro no sistema, a fim de que ele possa ser automaticamente identificado em futuras conversas. No cadastro, são fornecidas algumas informações, por exemplo: nome (pelo qual gostaria de ser chamado), data de nascimento, sexo, e-mail e a cidade onde reside. A Figura 26 mostra uma das telas desse cadastro.

Beck se apresentará para o adolescente com o intuito de que ele faça o mesmo. Nas aberturas das conversações é comum se iniciar por uma apresentação do *chatbot*, seguido

Figura 26 – Tela de Cadastro de Usuário.

A interface de cadastro de usuário é exibida em uma tela com fundo verde escuro. No topo, há uma barra de progresso com três etapas numeradas: 1 (DADOS DE ACESSO), 2 (DADOS PESSOAIS) e 3 (LOCALIZAÇÃO). A etapa 2 está selecionada. O formulário principal, com fundo branco, contém o título 'Dados Pessoais' em verde. Abaixo dele, o texto 'Complete os dados abaixo' é exibido. O formulário possui três campos de entrada: 'Nome pelo qual gostaria de ser chamado', 'Data de Nascimento: dd/mm/aaaa' e '-- Informe o sexo --' com uma seta verde para baixo. Na base do formulário, há dois botões verdes: 'VOLTAR' e 'PROXIMO'.

Fonte: Autoria própria.

de um pedido de identificação do usuário, assim como acontece nas ligações telefônicas, onde aquele que recebe o telefonema costuma perguntar quem está falando antes de iniciar a conversa. Quando duas pessoas não se conhecem, a apresentação costuma ser um pouco mais prolongada. A Figura 27 mostra a apresentação de Beck para um usuário novo. Essa apresentação só é realizada uma vez. No turno (1) Beck diz para o usuário que antes de iniciar a conversa, gostaria de se apresentar para ele. A partir do turno (2) até o turno (4), Beck faz sua apresentação. No turno (5) ela fala para o usuário que agora gostaria de conversar com ele para conhecê-lo melhor, e no turno (6) ela avisa que fará perguntas e pergunta ao adolescente se ele está de acordo.

Figura 27 – Trecho da Apresentação de Beck para o Usuário.

- (1) Beck: Antes de começarmos, eu gostaria de me apresentar pra você.
- (2) Beck: Meu nome é Beck.
- (3) Beck: Moro em um computador na Internet, e nas horas vagas eu gosto de conversar com as pessoas.
- (4) Beck: Eu fui criada para tentar ajudar as pessoas a enfrentar e vencer algumas dificuldades da vida.
- (5) Beck: Agora eu gostaria de conversar com você para o conhecer melhor.
- (6) Beck: Vou fazer algumas perguntas. Tudo bem?

Fonte: Autoria própria.

A apresentação do adolescente será guiada por Beck. Desta forma, ela fará perguntas a fim de reunir algumas informações sobre o usuário. Seguindo a ideia dos pares adjacentes

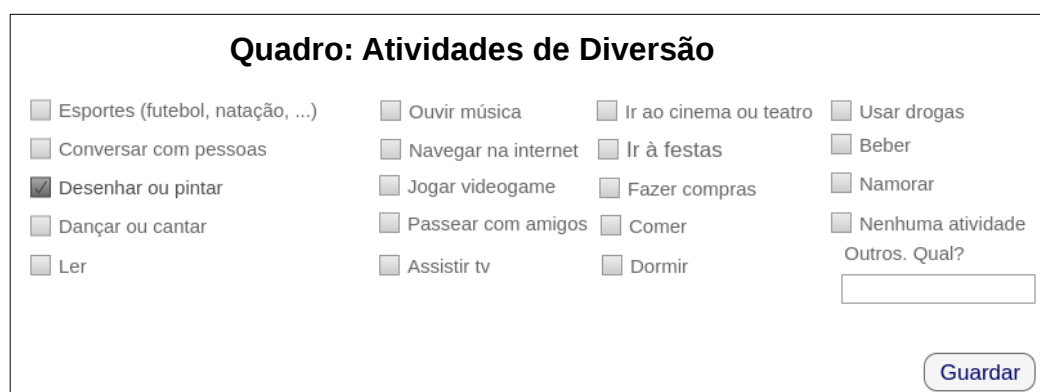
da TAC, numa pergunta, a preferência é pela resposta, mesmo que essa seja negativa. Algumas perguntas podem ser respondidas pelo usuário em linguagem natural (exemplos: *Com quem você mora?*, *Você tem namorado(a)?*). Enquanto outras podem ser respondidas através de formulários com múltiplas opções (exemplo: *O que você gosta de fazer para se divertir?*).

A Figura 28 mostra o formulário de atividades de diversão que é usado para que o adolescente escolha, dentre as opções, qual ou quais ele mais gosta de realizar. É importante destacar que o formulário contém:

1. Atividades prazerosas - Exemplos: dançar, atividades físicas em geral, ir ao cinema ou teatro;
2. Atividades que dependendo da frequência e intensidade podem ser consideradas neutras ou negativas - Exemplos: comer, jogar videogame, navegar na internet, dormir;
3. Atividades consideradas prejudiciais para a fase da adolescência (ou também para qualquer outra fase da vida) - Exemplos: beber e usar drogas.

O formulário de atividades também oferece o campo *outros*, para que o adolescente fique livre para informar outra atividade que não esteja listada.

Figura 28 – Quadro de Atividades de Diversão.



O formulário, intitulado "Quadro: Atividades de Diversão", apresenta uma grade de 4x4 com checkboxes para selecionar atividades. A atividade "Desenhar ou pintar" está selecionada. Abaixo da grade, há um campo de texto rotulado "Outros. Qual?" e um botão "Guardar".

Quadro: Atividades de Diversão			
☐ Esportes (futebol, natação, ...)	☐ Ouvir música	☐ Ir ao cinema ou teatro	☐ Usar drogas
☐ Conversar com pessoas	☐ Navegar na internet	☐ Ir à festas	☐ Beber
☒ Desenhar ou pintar	☐ Jogar videogame	☐ Fazer compras	☐ Namorar
☐ Dançar ou cantar	☐ Passear com amigos	☐ Comer	☐ Nenhuma atividade
☐ Ler	☐ Assistir tv	☐ Dormir	Outros. Qual?

Guardar

Fonte: Autoria própria.

A maioria das perguntas feitas para o adolescente tem o objetivo de conhecê-lo melhor e também de detectar fatores de risco que podem causar problemas emocionais nos indivíduos dessa faixa etária. Alguns dos fatores que podem tornar a criança e o adolescente mais vulneráveis a sofrerem problemas psicológicos são (AVANCI et al., 2007; BECK, 1995; PSQUIATRIA, 2014):

- Individuais - Sexo, idade, características psicológicas (como baixa auto-estima e falta de autoconfiança);

- Familiares - História de problemas de saúde mental, especialmente materna, problemas de álcool/drogas, violência física, psicológica e sexual, violência entre os pais, perdas por morte, separação dos pais;
- Biológicos - Fatores genéticos, aparência física, doença incapacitante; e
- Socioculturais - Pobreza, violência no contexto social, relacionamento com os pares.

Como são muitos fatores de risco associados, e infelizmente neste trabalho de mestrado não podemos tratar todos, Beck fará perguntas para detectar alguns desses fatores. Por exemplo: no turno (1) da conversação ilustrada pela Figura 29, Beck pergunta ao adolescente com quem ele mora. Se o adolescente responder que não mora com os pais ou que mora com apenas um deles, como no caso do turno (2), onde foi respondido que o adolescente mora com a mãe, Beck perguntará para ele onde está o pai, conforme turno (3). Beck guardará a resposta do adolescente para futuros diálogos.

Figura 29 – Pergunta de Beck sobre os Pais do Adolescente.

(1) Beck: Com quem você mora?
(2) Junior: minha mãe
(3) Beck: Onde está seu pai?
(4) Junior: meus pais são separados, ele mora na casa dele
(5) Beck: Entendo.

Fonte: Autoria própria.

Como visto, as apresentações entre Beck e o usuário só ocorrem uma única vez. Nos demais acessos que o usuário fizer ao sistema, Beck o cumprimentará e pedirá para que ele responda como está o seu humor, antes de iniciar ou continuar o tópico da conversação.

Terminados os cumprimentos e as apresentações, Beck dirá para o adolescente que agora já o conhece um pouco e está feliz por estar construindo um vínculo com ele, conforme podemos observar no turno (4) do trecho da conversação representado na Figura 30. É importante dizer, como pode ser observado também no turno (4) da figura, que Beck consegue tratar corretamente o artigo “o” ou “a”, que faz o papel de um pronome pessoal, quando está conversando com um usuário do sexo masculino ou feminino. Esta informação é recuperada a partir do cadastro do usuário.

4.2.3 Organização do Diálogo

Como já mencionado, os diálogos criados para Beck foram estruturados com base na TAC. A base de diálogos foi organizada em três seções, Abertura, Desenvolvimento e Fechamento, seguindo o modelo de organização global apresentado pela teoria da conver-

Figura 30 – Fim da Aquisição de Perfil.

(1) Beck: Você tem namorado(a)?
(2) Junior: sim
(3) Beck: Certo.
(4) Beck: Agora eu já o conheço um pouco e fico feliz de estarmos construindo um vínculo.

Fonte: Autoria própria.

sação. Em relação à organização local da conversação, Beck executa os diálogos através de turnos.

Abertura

Na seção de abertura, Beck dá as boas-vindas ao usuário, já o identificando com o nome que ele escolheu usar e, em seguida, se apresenta para o usuário conforme visto na seção de aquisição de perfil.

Em Neves, Barros e Hodges (2006), foram realizadas algumas análises nos *logs* de conversas de alguns *chatbots* com os usuários. Foi constatado que, após a seção de abertura da conversação, já na seção de desenvolvimento, alguns usuários repetiam cumprimentos para verificar a reação do *chatbot*. Conforme relatado pelos autores, a maioria dos *chatbots* só conseguiu detectar a repetição do cumprimento quando ele era realizado imediatamente após a frase atual do usuário.

Para evitar essa situação relatada em Neves, Barros e Hodges (2006), e para preservar o modelo de organização global da conversação implantado em Beck, foi criada uma variável de controle no mecanismo do *chatbot* para que essa situação não ocorra. Assim, quando a conversa com o usuário já passou da seção de abertura, e o usuário cumprimentar Beck durante a seção de desenvolvimento ou fechamento, ela perguntará ao usuário se já não o cumprimentou antes. Além de lembrar para o usuário que os cumprimentos já foram realizados, Beck também pode controlar a repetição desses cumprimentos, alertando o usuário que, caso ele repita várias vezes o cumprimento, ela se negará a continuar falando com ele.

Na Figura 31 podemos ver o trecho de uma conversa de Beck com o usuário, na qual ele repete os cumprimentos. No turno (1), Beck faz uma pergunta ao usuário. No turno (2), o usuário a cumprimenta novamente. No turno (3), Beck questiona se já não cumprimentou o usuário antes, no turno (4) o usuário insiste e faz um novo cumprimento. No turno (5), Beck afirma para o usuário que ele está sendo repetitivo. Se o usuário continuar insistindo em cumprimentá-la, Beck informará que não poderá continuar a conversa, enquanto ele não mudar o seu comportamento.

Quando o usuário acessa Beck pela segunda vez em diante, ela o cumprimenta e pede para que o usuário responda o formulário do humor, a fim de acompanhar como ele está

Figura 31 – Controle da Organização Global da Conversa.

- (1) Beck: Com quem você mora?
- (2) Junior: olá.
- (3) Beck: Já não nos cumprimentamos antes?
- (4) Junior: oi
- (5) Beck: Você está sendo um um pouco repetitivo.

Fonte: Autoria própria.

se sentindo. Veremos o formulário do humor na seção seguinte.

Desenvolvimento

Após a seção de abertura, Beck iniciará uma conversação com o adolescente. Os tópicos abordados por ela envolvem os conceitos da TCC. Beck possui um comportamento proativo, ela tomará a postura de conduzir a conversa.

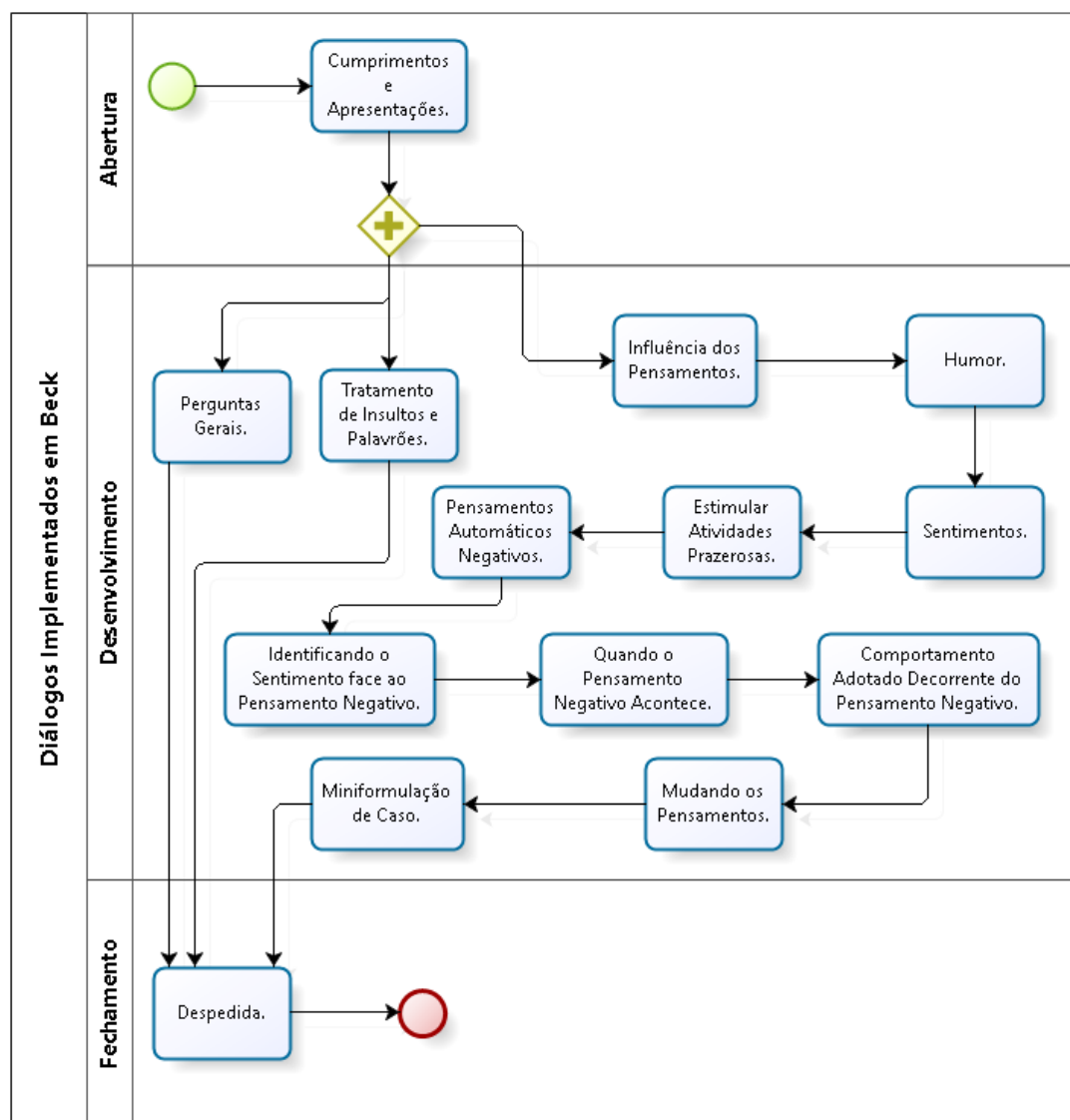
Os tópicos relacionados nesta seção são sequenciais, exceto o de perguntas gerais e o de tratamento de insultos e palavrões. Então Beck, ao concluir um tópico, já inicia o próximo. A Figura 32 mostra o fluxograma dos tópicos de conversação abordados por Beck.

Para o mecanismo de conversação implementado em Beck vale a máxima: “fala um de cada vez”. Assim, quando Beck estiver com a palavra, o adolescente não poderá falar, até que haja novamente a troca de turno e seja a vez dele falar. Para isso, foi implementado um bloqueio no campo de mensagem e no microfone da tela principal de Beck, conforme mostra a Figura 33.

Os 12 tópicos atualmente abordados por Beck são descritos a seguir.

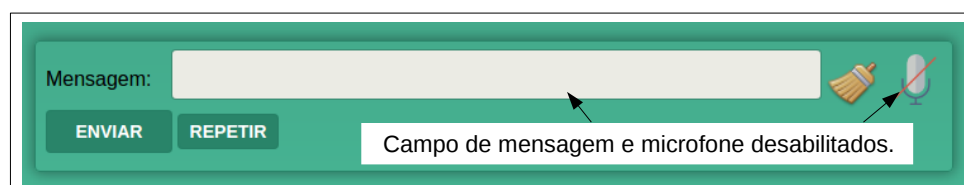
1. **Influência dos Pensamentos** - Dentro desse tópico, Beck falará sobre os pensamentos e como eles podem influenciar os sentimentos e os comportamentos adotados pelo indivíduo. Do turno (1) até o (3) da Figura 34, temos um trecho sobre esse tópico.
2. **Humor** - Esse tópico foi criado para verificar o humor do adolescente durante a conversa com Beck. O objetivo é verificar se o adolescente tem tido alguma melhora a cada conversa realizada. A Figura 35 mostra o quadro do humor com cinco opções, são elas: muito triste, triste, indiferente, feliz e muito feliz. O adolescente é instruído a saber o que cada figura representa ao apontar a seta do mouse sobre ela.
3. **Sentimentos** - Esse tópico busca explanar sobre como os sentimentos influenciam os comportamentos adotados pela pessoa. No decorrer desse tópico também é perguntado para o adolescente se ele possui amigos. Um dos problemas que permeia a

Figura 32 – Diálogos Implementados em Beck.



Fonte: Autoria própria.

Figura 33 – Bloqueio do Campo de Mensagem e Microfone.



Fonte: Autoria própria.

adolescência é a interação social com os pares. A Figura 36 mostra um trecho da conversa sobre o tópico sentimentos. Do turno (1) ao (5) Beck, fala sobre sentimentos, e do turno (6) ao (8), ela fala sobre amigos.

Figura 34 – Diálogo Sobre a Influência dos Pensamentos.

- (1) Beck: Você sabia que os pensamentos que as pessoas têm podem afetar os seus sentimentos e os seus comportamentos?!
- (2) Beck: Vou dar um exemplo: quando temos pensamentos pessimistas, geralmente nos sentimos tristes.
- (3) Beck: Veja que os sentimentos também podem afetar os pensamentos e os comportamentos.

Fonte: Autoria própria.

Figura 35 – Quadro do Humor.

Como você está se sentindo agora?

				
☐	☐	☐	☐	☐

Fonte: Autoria própria.

Figura 36 – Conversa sobre os Sentimentos.

- (1) Beck: Vamos falar sobre alguns sentimentos...
- (2) Beck: Você sabia que quando nos sentimos bem, temos mais vontade de criar e descobrir coisas, conhecer e nos relacionarmos com as pessoas?!
- (3) Beck: E o contrário também é verdadeiro.
- (4) Beck: Quando nos sentimos mal, temos menor probabilidade de fazer as coisas que gostamos.
- (5) Beck: E então começamos a duvidar da nossa capacidade... Como, por exemplo, da nossa capacidade de fazer amigos.
- (6) Beck: Por falar em amigos... Você tem amigos?
- (7) Junior: sim, a Maria e o João
- (8) Beck: Maria e João, são os seus melhores amigos?

Fonte: Autoria própria.

4. **Estimular Atividades Prazerosas** - Nesse tópico, serão discutidas as atividades que o adolescente disse que gostava de realizar durante a seção de Aquisição de Perfil. As atividades escolhidas previamente por ele são então listadas e, em seguida, é perguntado para ele qual das atividades ele tem realizado com maior frequência durante a semana. O objetivo é estimular as atividades prazerosas. Portanto, se a atividade escolhida pelo adolescente é feita com baixa ou nenhuma frequência, ele é então estimulado a fazê-la mais vezes. Beck negociará prazos para a realização da atividade, a fim de que o adolescente faça um compromisso com ela e cumpra o prazo combinado. Caso não tenha sido selecionada nenhuma atividade durante a seção de Aquisição de Perfil, então Beck sugerirá alguma atividade para o adolescente fazer.

A Figura 37 mostra um trecho de diálogo sobre esse tópico.

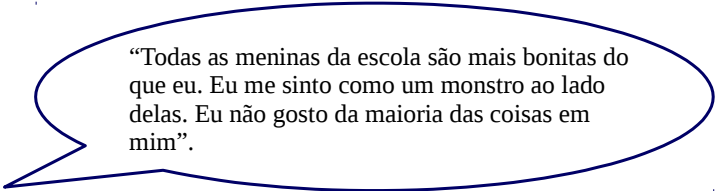
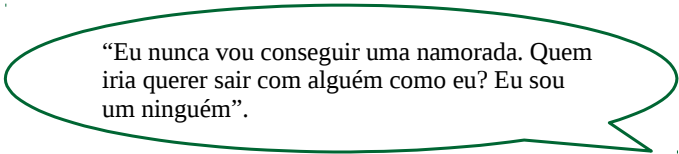
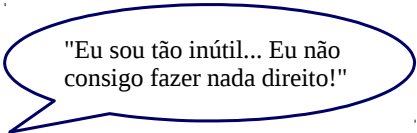
Figura 37 – Conversa sobre Estimular Atividades prazerosas.

Beck: Essas são algumas das atividades que você disse que gostava de realizar: **futebol e assistir tv** .
Beck: Qual dessas atividades você tem realizado mais frequentemente?
Junior: **futebol**
Beck: Certo.
Beck: Qual a frequência com que você tem realizado esta atividade?
Junior: **uma vez por mês**
Beck: Vamos tentar realizá-la pelo menos uma vez nesta semana?
Junior: **talvez**
Beck: Não fique em dúvida. Tenho certeza de que você vai conseguir.
Beck: Vamos fazer um compromisso então?
Beck: O que você acha de eu lhe perguntar daqui a uns 2 dias se você conseguiu realizar essa atividade. Tudo bem?
Junior: **hummm**
Beck: Que tal 3 dias?

Fonte: Autoria própria.

5. **Pensamentos Automáticos Negativos** - Nessa fase da conversação, são apresentadas três personagens para o adolescente, e os seus respectivos pensamentos negativos. Os personagens têm o objetivo de ilustrar para o adolescente que tipos de pensamentos negativos que as pessoas podem ter. Durante essa fase, é perguntado para o adolescente se ele tem algum pensamento negativo. Em caso afirmativo, Beck pergunta qual seria esse pensamento. Como explanado na seção 3.1, os pensamentos automáticos negativos podem ser categorizados em alguns erros de pensamento. De acordo com o pensamento exposto pelo adolescente, Beck tentará categorizá-lo em algum desses tipos. A Figura 38 mostra exemplos dos pensamentos automáticos negativos das personagens Maria, João e a Joana.
6. **Identificando o sentimento face ao pensamento negativo** - Nesse tópico, são explorados os sentimentos que o adolescente pode sentir quando tem algum pensamento negativo. É exibido um formulário com várias opções a fim de que ele identifique os sentimentos face aos pensamentos negativos. A Figura 39 mostra o formulário com algumas opções de sentimentos que o adolescente pode vivenciar.
7. **Quando o pensamento negativo acontece** - No decorrer desse tópico são utilizadas novamente as personagens apresentadas durante o tópico dos pensamentos automáticos negativos, mas desta vez, elas são usadas para ilustrar as situações vivenciadas que fazem com que elas tenham os pensamentos negativos. Depois, é perguntado para o adolescente se ele conseguiu perceber qual é a situação que faz

Figura 38 – Os Pensamentos Negativos da Maria, do João e da Joana.

Identificando os Pensamentos Automáticos Negativos	
Veja o exemplo do pensamento negativo que a Maria tinha:	
 “Todas as meninas da escola são mais bonitas do que eu. Eu me sinto como um monstro ao lado delas. Eu não gosto da maioria das coisas em mim”.	
Observe o pensamento negativo do João:	
 “Eu nunca vou conseguir uma namorada. Quem iria querer sair com alguém como eu? Eu sou um ninguém”.	
E a Joana, veja o pensamento negativo dela:	
 "Eu sou tão inútil... Eu não consigo fazer nada direito!"	

Fonte: Autoria própria.

Figura 39 – Quadro o que eu sinto com frequência.

O que eu sinto com frequência?			
☐ Angústia	☐ Irritação	☐ Tristeza	☐ Sentimento de que perdi peso
☐ Ansiedade	☐ Insônia	☐ Receio	☐ Dúvidas sobre mim mesmo
☐ Cansaço	☐ Medo	☐ Sentimento de Culpa	☐ Vontade de comer
☐ Chateação	☐ Preocupação	☐ Sentimento de fracasso	☐ Vontade de chorar
☐ Ciúmes	☐ Vergonha	☐ Sentimento de rejeição	Outros. Qual?
			Guardar

Fonte: Autoria própria.

com que ele tenha pensamentos negativos. A Figura 40 mostra as situações vivenciadas pelas personagens.

- Comportamento adotado decorrente do pensamento negativo** - Nesse tópico, Beck conversa com o adolescente para saber o que ele faz quando tem algum pensamento negativo. É apresentado um formulário para que ele identifique o comportamento que ele pode adotar. A Figura 41 mostra o formulário com algumas opções de comportamentos, além do campo “outros” para que ele informe outro comportamento que não esteja listado no formulário.

Figura 40 – Situações Vivenciadas pelas Personagens que Desencadeavam os Pensamentos Negativos.

Quando o pensamento negativo acontece		
A Maria, que se achava um monstro, tinha esses pensamentos quando olhava os cabelos das amigas e os comparava com os dela.	O João sempre pensava que nunca iria conseguir namorar quando via as postagens do Facebook do seu amigo Carlos com a namorada Jéssica.	A Joana pensava que não fazia nada direito quando estava na aula de dança, já que não conseguia fazer todos os passos.

Fonte: Autoria própria.

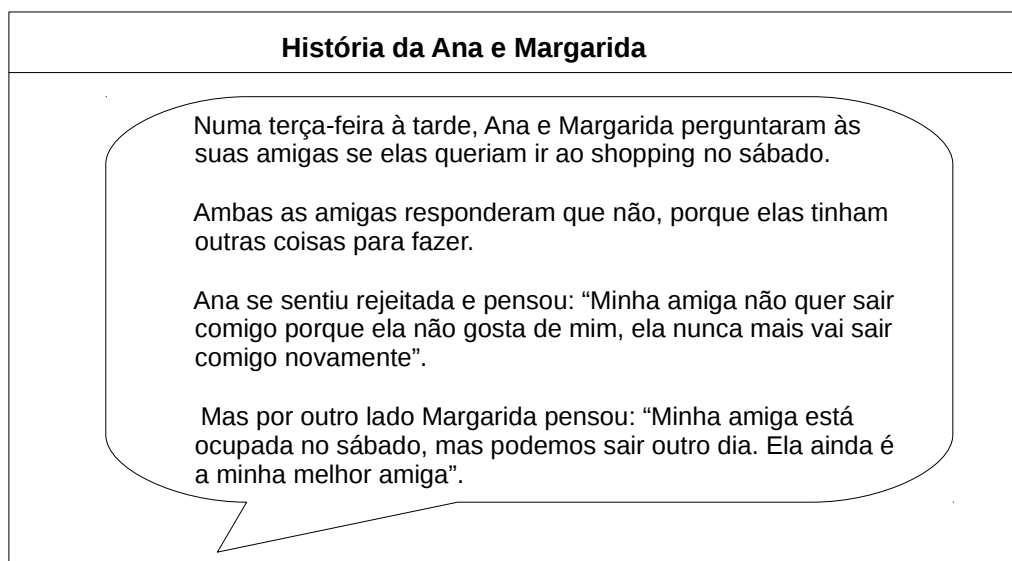
Figura 41 – Quadro o que eu faço com frequência.

O que eu faço com frequência?	
☐ Brigo com as pessoas	☐ Faço bastante exercício físico
☐ Como muito	☐ Nada
☐ Choro bastante	Outro. Qual?
☐ Me isolo das pessoas	
☐ Machuco a mim mesmo	
Guardar	

Fonte: Autoria própria.

9. **Mudando os pensamentos negativos** - Nessa fase do diálogo, são apresentadas três outras personagens e suas histórias. O intuito desse tópico é mostrar para os adolescentes que os pensamentos negativos acontecem, geralmente, porque temos uma reação exagerada em determinadas situações. A melhor forma para validar um pensamento é buscar evidências de que ele está correto. Na Figura 42 é exibida a história de duas dessas personagens, Ana e Margarida, sobre as diferentes reações que ambas tiveram em relação a um mesmo evento. Depois de exibir essa história, Beck faz algumas perguntas ao adolescente e, em seguida, mostra a história da terceira personagem.
10. **Miniformulação de Caso** - Nesse tópico são recuperadas e reunidas as informações compartilhadas pelo adolescente durante os tópicos anteriores. É criada então uma miniformulação de caso que contém: (I) a situação vivenciada, (II) os pensamentos automáticos negativos decorrentes dessa situação, (III) os sentimentos resultantes

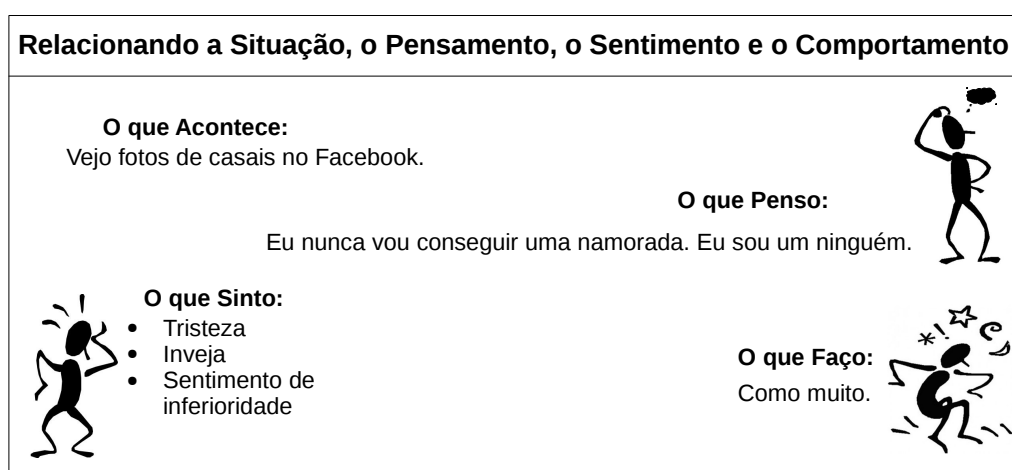
Figura 42 – A História da Ana e Margarida.



Fonte: Autoria própria.

desses pensamentos e (IV) o comportamento adotado pelo adolescente. O objetivo da formulação de caso é mostrar para o adolescente que os pensamentos, sentimentos e comportamentos estão todos conectados, e que, para mudar, o primeiro passo que devemos dar é reconhecer aquilo que não nos faz bem. A Figura 43 ilustra uma situação ocorrida, os pensamentos negativos que surgiram decorrentes da situação, os sentimentos resultantes e o comportamento praticado.

Figura 43 – Miniformulação de Caso do Usuário.



Fonte: Autoria própria.

- Perguntas Gerais** - Em Perkins (2016), foi relatada uma pesquisa para saber quais as perguntas mais realizadas pelo usuários ao conversar com um *chatbot*. Foram elencadas as dez perguntas mais frequentes. Algumas dessas perguntas são: *Saber a*

idade do chatbot; Se ele é real; Que línguas pode falar; Onde mora. Essas perguntas e algumas outras mais foram incluídas na base de conhecimento de Beck.

12. **Tratamento de Insultos e Palavrões** - Uma situação que pode acontecer quando alguns usuários estão conversando com *chatbots* é realização de insultos e uso de palavrões. Em Beck, foi implementado um tratamento para esses casos. Se Beck for insultada pelo adolescente ou se ele usar palavrões durante a conversa, ela automaticamente pedirá que o usuário mude seu comportamento para que a conversa possa continuar. Caso o adolescente insista com esse comportamento, Beck pode encerrar a conversação, avisando-o previamente.

A Figura 44 ilustra uma situação onde Beck é insultada. No turno (2), o usuário faz um insulto a Beck; no turno (3), ela pergunta o motivo do insulto; no turno (4), o adolescente continua com os insultos; no turno (5) Beck percebe novamente que foi insultada e pede para o adolescente mudar o comportamento. No turno (6), o adolescente diz que não vai parar. No turno (7), Beck insiste para ele parar de realizar os insultos. Beck se comportará sempre tentando voltar à conversa, mas se o adolescente insistir em fazer insultos ou usar palavrões, ela o avisará que a conversa será encerrada.

Figura 44 – Tratamento de Insultos.

(1) Beck: Seja bem-vindo Junior! Tudo bem com você?
(2) Junior: #%\$#@-
(3) Beck: Por que você está me insultando?
(4) Junior: #!&\$%*@&
(5) Beck: Você me insultou novamente! Peço para você parar com esse comportamento. Tudo bem?
(6) Junior: não
(7) Beck: Eu insisto, por favor, pare com esse comportamento. Vamos continuar conversando... Está tudo bem com você?

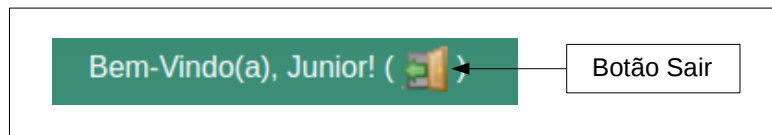
Fonte: Autoria própria.

Fechamento

O fechamento é a fase de encerramento do diálogo. Nos fechamentos, geralmente, quem iniciou a conversa é que sinaliza que deseja terminá-la. Quando o usuário informa que quer terminar a conversa, através da linguagem natural, Beck pergunta para o usuário se ele precisa ir mesmo embora. Se houver a confirmação do usuário, então Beck se despede e finaliza a seção de conversação. Note que, Beck também pode encerrar a conversa com o usuário quando os tópicos programados para o dia forem finalizados. Ainda existe a opção de terminar a conversa através do botão *Sair*, que fica na tela principal de Beck.

Ao clicar nesse botão, a conversa é imediatamente finalizada com o usuário. A Figura 45 mostra o botão *Sair* da tela principal de Beck.

Figura 45 – Botão Sair.



Fonte: Autoria própria.

A Figura 46 ilustra o encerramento de uma conversa do usuário com Beck. No turno (1), o usuário sinaliza que precisa finalizar a conversa. No turno (2), Beck pergunta se o usuário tem que ir naquele momento mesmo. No turno (3), o usuário confirma, e nos turnos (4) e (5), Beck se despede do usuário.

Figura 46 – Fechamento da Conversação.

- (1) Junior: **preciso ir embora**
- (2) Beck: **Tem que ir agora mesmo?**
- (3) Junior: **infelizmente tenho**
- (4) Beck: **Tudo bem, depois conversamos mais!**
- (5) Beck: **Tchau Junior!**

Fonte: Autoria própria.

4.3 Arquitetura do Sistema

O *chatbot* Beck foi construído para a plataforma *Web*, e está disponível na Internet no endereço **beck-bot.cin.ufpe.br**. Beck funciona através de troca de mensagens em texto e/ou voz. Foram utilizadas algumas ferramentas e algoritmos no processo de sua criação. Todos os recursos utilizados são de código aberto.

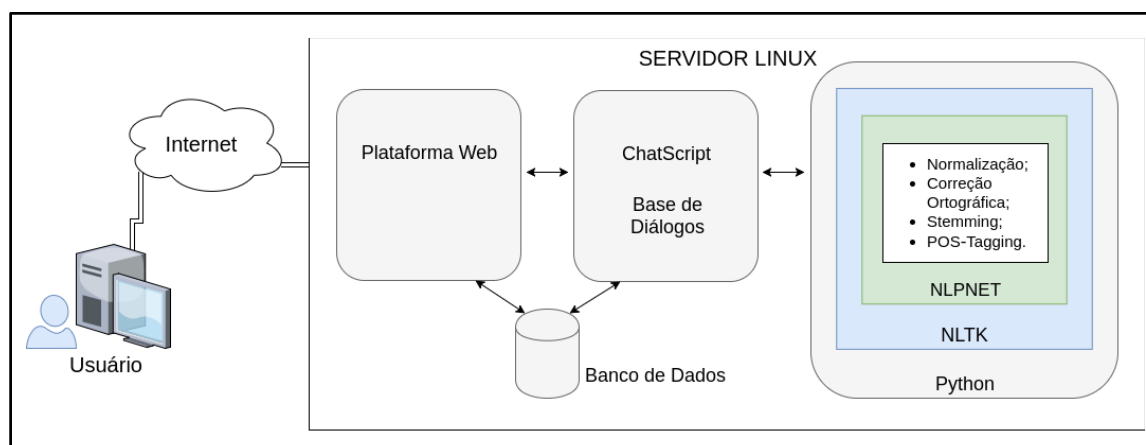
A plataforma *Web*, para o *chatbot* Beck, foi implementada através de um conjunto de ferramentas e linguagens que permitem que os usuários possam acessar e conversar com Beck através de um Navegador *Web*. Essa plataforma se comunica com um Banco de Dados para guardar algumas informações sobre o usuário. A plataforma também se comunica com o *ChatScript*, linguagem usada para criar a base de conhecimento de Beck.

Antes do *ChatScript* realizar a interpretação da mensagem do usuário, são realizados os processos de normalização do texto, correção ortográfica, *Stemming* e *POS-Tagging*. Para isso, o *ChatScript* se comunica com o NLPNET² através de um *script* desenvolvido

² <<http://nilc.icmc.usp.br/nlpnet/index.html>>

na linguagem Python³. O NLPNET é uma biblioteca do Python, e funciona através do *Natural Language Toolkit (NLTK)*⁴, sendo responsável pelo processo de *POS-Tagging*. Aproveitamos essa biblioteca e embutimos nela a função para normalização das sentenças do usuário, a ferramenta para correção ortográfica e o algoritmo de *Stemming*. A Figura 47 mostra a arquitetura montada para o sistema, implementada em um Servidor Linux, que possibilita o acesso dos usuários ao *chatbot* Beck.

Figura 47 – Arquitetura Geral do Sistema.



Fonte: Autoria Própria.

A seguir, apresentaremos brevemente as ferramentas e os algoritmos usados para criação de Beck, como também das adaptações realizadas em alguns desses recursos. No Apêndice B, temos o processo de instalação de todos os elementos utilizados para a criação do *chatbot*.

4.3.1 Plataforma Web

Como já mencionado, Beck está disponível para a plataforma *Web* e, para que isso fosse possível, utilizamos alguns recursos e linguagens específicos para essa plataforma. Os recursos utilizados foram: Linguagem PHP, Servidor Apache HTTP Server⁵, HTML⁶, CSS⁷ e *Javascript*⁸.

Para acessar o *chatbot* Beck, o usuário precisará de uma conexão com a Internet e ter instalado no computador um Navegador *Web*, como o Google Chrome. Somente a partir da versão 58 desse navegador é que Beck poderá se comunicar através do Reconhecimento e Síntese de voz. Para os demais navegadores, Beck só funciona em modo texto.

³ <<https://www.python.org/>>

⁴ <<http://www.nltk.org/>>

⁵ <<http://httpd.apache.org/docs/current/>>

⁶ <<https://www.w3schools.com/html/default.asp>>

⁷ <<https://www.w3schools.com/css/>>

⁸ <<https://www.w3schools.com/js/default.asp>>

Para permitir a comunicação através de voz, utilizamos a interface de programação em *JavaScript* chamada *Web Speech API*. Essa interface permite integrar reconhecimento e síntese de voz a páginas *Web* (SHIRES; WENNBORG, 2012). Antes de executar a síntese de voz utilizando essa API, é necessário realizar um tratamento no texto, pois há uma limitação do número de caracteres das palavras que podem ser sintetizados. De acordo com Shires e Wennborg (2012), o limite é de 32.767 caracteres, mas infelizmente esse número não está correto, dado que tivemos problemas com um número bem menor de caracteres. Através de pesquisas realizadas, alguns usuários falam em um tempo limitado em 60 segundos de síntese de voz (GOOGLE, 2016).

Para corrigir esse problema, implementamos um *script* de quebra automática do texto a cada 150 caracteres. Essa quebra leva em consideração a estrutura da sentença (sinais de pontuação). Outro tratamento realizado no texto que incluímos junto a esse *script*, foi a retirada de *tags* HTML que são usadas para formatar algumas respostas de Beck exibidas na tela para o usuário. Se essas *tags* permanecerem junto ao texto no momento do envio para o sintetizador de fala, elas serão pronunciadas juntamente com o texto.

A Figura 48 mostra o código que realiza esses tratamentos. Na linha (1) da figura é declarada a função de tratamento de texto. Na linha (3), é declarada a função *stripHTML* para retirada de *tags* HTML. Na linha (4), é aplicada a função *stripHTML* no texto. Na linha (5), é declarada a variável *tamMaximoTexto* para limitar o número máximo de caracteres que pode ser sintetizado. Na linha (6), é criada uma expressão regular para quebra automática do texto. Na linha (9), é declarada uma variável para guardar as sentenças após a quebra automática. Da linha (10) à (13), é aplicada a expressão regular para realização da quebra do texto. Na linha (14), é declarado um contador para controlar a síntese de voz para cada segmento do texto após a quebra automática. Da linha (15) à linha (27), é declarada e executada a função *loopArrayTexto* para chamar o sintetizador de voz para cada segmento do texto.

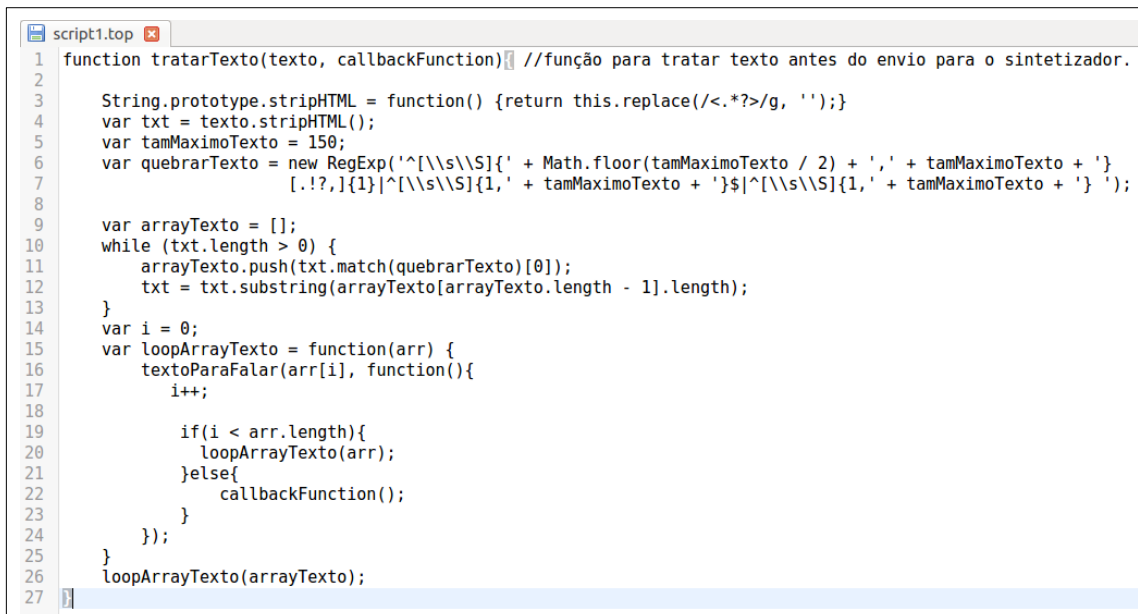
4.3.2 Banco de Dados

Para armazenar alguns dados do usuário, como os dados cadastrais, e algumas informações capturadas durante as conversas com Beck, utilizamos um banco de dados PostgreSQL⁹.

A modelagem do banco de dados montada para Beck pode ser vista na Figura 49. Nessa Figura, podemos ver a estrutura e as relações entre as tabelas que compõem o banco de dados do sistema. A tabela *users* é utilizada para armazenar os dados cadastrais dos usuários, permitindo assim a validação dos seus acessos. A tabela *elementos* é usada para guardar a resposta de qual dos elementos principais da TCC (pensamento, sentimento e comportamento) o usuário acredita ser mais fácil de ser controlado. As tabelas *uf* e *municipio* guardam uma lista dos estados e municípios brasileiros. A tabela *humor* é usada para armazenar o humor do adolescente, que é verificado a cada dia de conversa. A

⁹ <<https://www.postgresql.org/>>

Figura 48 – Tratamento do Texto para a Síntese de Voz.

A screenshot of a code editor window titled 'script1.top'. The code is in JavaScript and defines a function 'tratarTexto' that processes text for voice synthesis. The code includes comments in Portuguese and uses various JavaScript methods like 'stripHTML', 'match', 'substring', and 'push'. It also uses a regular expression to split text into segments of a maximum length of 150 characters. The code is as follows:

```
1 function tratarTexto(texto, callbackFunction) //função para tratar texto antes do envio para o sintetizador.
2
3     String.prototype.stripHTML = function() {return this.replace(/<.*?>/g, '');}
4     var txt = texto.stripHTML();
5     var tamMaximoTexto = 150;
6     var quebrarTexto = new RegExp('^([\\s\\S]{1, ' + Math.floor(tamMaximoTexto / 2) + ', ' + tamMaximoTexto + '}'
7         + '[!?,]{1})^([\\s\\S]{1, ' + tamMaximoTexto + '}' + '$|^([\\s\\S]{1, ' + tamMaximoTexto + '}' + ' '));
8
9     var arrayTexto = [];
10    while (txt.length > 0) {
11        arrayTexto.push(txt.match(quebrarTexto)[0]);
12        txt = txt.substring(arrayTexto[arrayTexto.length - 1].length);
13    }
14    var i = 0;
15    var loopArrayTexto = function(arr) {
16        textoParaFalar(arr[i], function(){
17            i++;
18
19            if(i < arr.length){
20                loopArrayTexto(arr);
21            }else{
22                callbackFunction();
23            }
24        });
25    }
26    loopArrayTexto(arrayTexto);
27
```

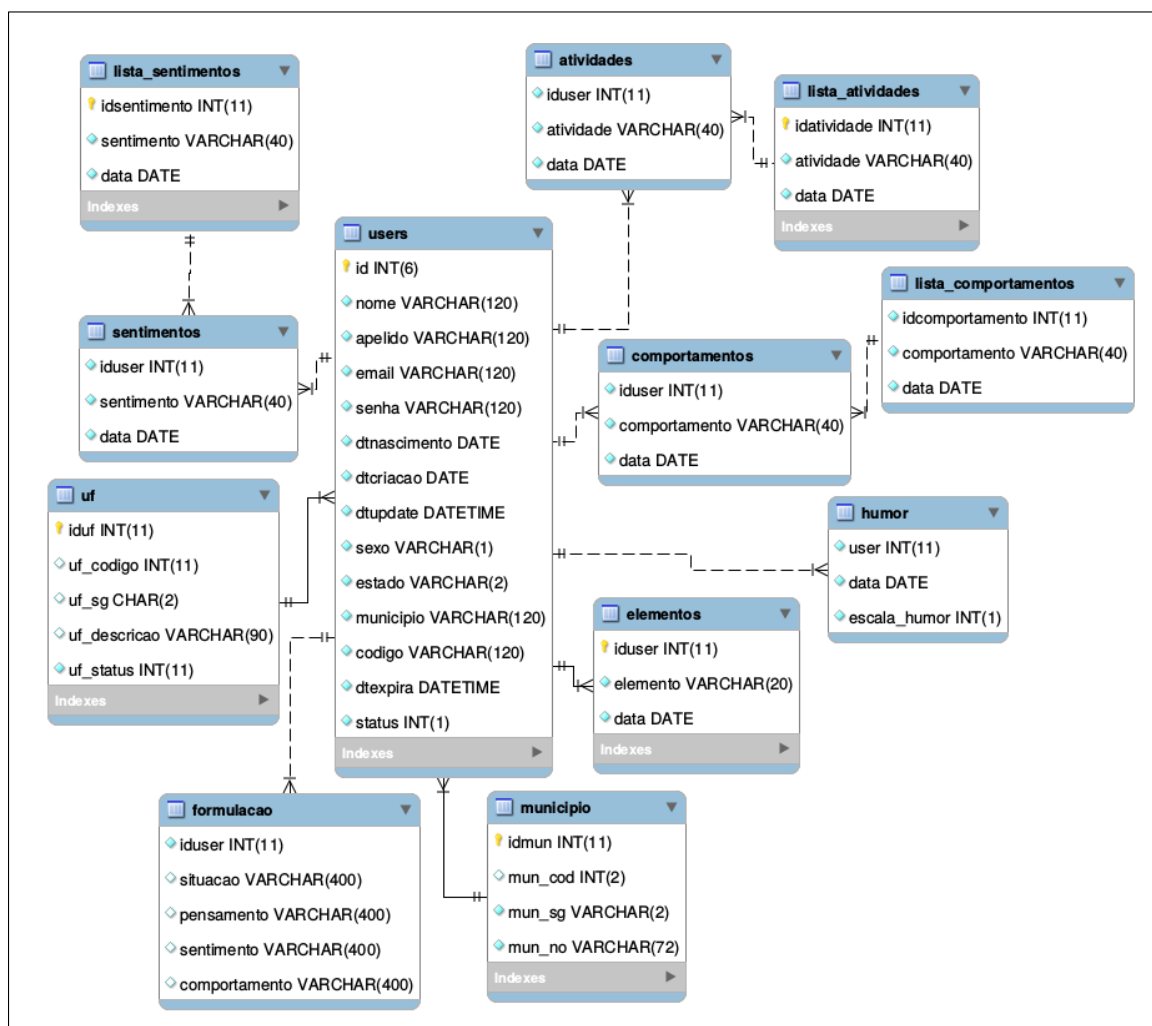
Fonte: Autoria Própria.

tabela *lista_atividades* guarda todas as atividades de diversão cadastradas no sistema, e a tabela *atividades* armazena as atividades que o adolescente informou que gosta de fazer durante as conversas com Beck. A tabela *lista_sentimentos*, armazena todas as opções de sentimentos que são apresentadas para o usuário para que ele escolha, entre os elementos da lista, quais sentimentos ele possui. A tabela *sentimentos*, registra os sentimentos escolhidos pelo usuário durante o diálogo. A tabela *lista_comportamentos*, guarda uma lista de opções de comportamentos que são exibidos para o usuário. A tabela *comportamentos*, guarda os comportamentos selecionados pelo usuário durante a conversação com Beck. A tabela *formulação*, é utilizada para armazenar a situação vivenciada pelo adolescente, o pensamento automático negativo, o sentimento e o comportamento, informados pelo usuário durante a conversação com Beck.

4.3.3 ChatScript - Base de Diálogos

Para construção da Base de Diálogos do *chatbot* utilizamos o *ChatScript*. A linguagem *ChatScript* é o principal componente da arquitetura criada para Beck. Essa linguagem oferece vários recursos, mas infelizmente alguns deles apenas para o inglês (por exemplo: WordNet, correção ortográfica e *POS-Tagging*). Assim, utilizamos algumas ferramentas externas para poder continuar utilizando alguns desses recursos no *ChatScript*.

Visando simplificar o funcionamento do sistema, centralizamos o uso desses recursos externos dentro do NLPNET. Para tornar possível a comunicação do *ChatScript* com o NLPNET foi desenvolvido um *script* em Python conforme pode ser observado na Figura

Figura 49 – Modelagem do Banco de Dados para o *Chatbot* Beck.

Fonte: Autoria Própria.

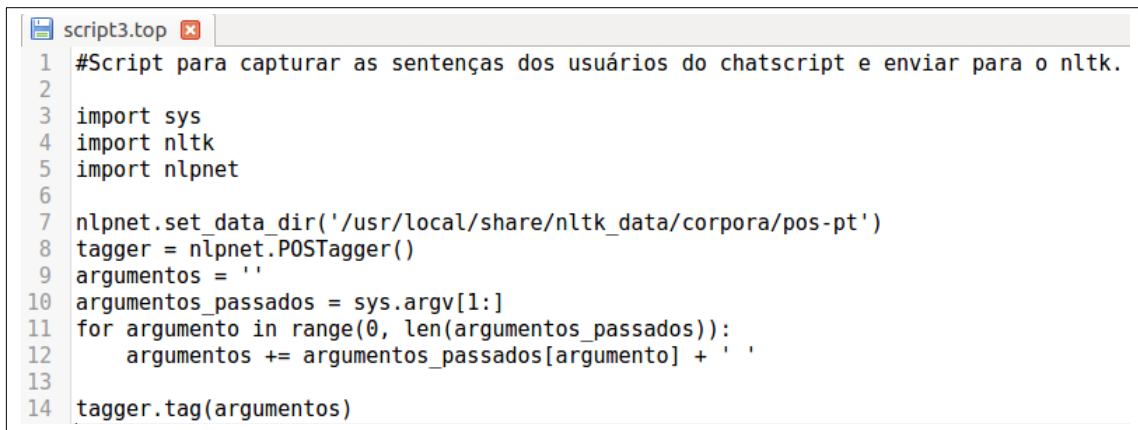
50. Na Figura, da linha (3) até a (5) temos a importação de algumas bibliotecas. Na linha (7), é definido o local dos corpora¹⁰ que serão usados para determinar a classe gramatical das palavras. Na linha(8), é criada uma instância da classe *POS-Tagger* do NLPNET. Na linha (9), é declarada uma variável para guardar a frase de entrada do usuário. Na linha (10), é capturada essa entrada do usuário recebida do *ChatScript*. Nas linhas (11) e (12), é realizado um processo para armazenar a frase do usuário em uma variável do tipo *string*. E na linha (14), é executado o método de *POS-Tagging*.

4.3.4 NLPNET

O NLPNET é uma biblioteca do Python utilizada para realizar tarefas de PLN baseado em redes neurais. Atualmente, essa biblioteca pode realizar *POS-Tagging* e Etiquetagem

¹⁰ Conjunto de documentos em textos de uma língua, e que servem como base para realização de análises.

Figura 50 – Script Python para Comunicação com o NLPNET.



```

1 #Script para capturar as sentenças dos usuários do chatscript e enviar para o nltk.
2
3 import sys
4 import nltk
5 import nlpnet
6
7 nlpnet.set_data_dir('/usr/local/share/nltk_data/corpora/pos-pt')
8 tagger = nlpnet.POSTagger()
9 argumentos = ''
10 argumentos_passados = sys.argv[1:]
11 for argumento in range(0, len(argumentos_passados)):
12     argumentos += argumentos_passados[argumento] + ' '
13
14 tagger.tag(argumentos)

```

Fonte: Autoria Própria.

de Papel Semântico. Embora seja uma biblioteca do Python, ela depende do NLTK, do Punkt Tokenizer Models¹¹ e do NumPy¹² (FONSECA; ROSA, 2013a).

Como já mencionado, os processos de normalização de textos, verificação ortográfica, *Stemming* e *POS-Tagging* são executados a partir da biblioteca NLPNET. A seguir veremos um pouco mais sobre os recursos embutidos nessa biblioteca.

Normalização do Texto - Antes do texto ser submetido ao verificador ortográfico, é realizada uma normalização. Essa normalização tem o objetivo de retirar e separar símbolos e sinais de pontuações do texto para que não haja problema na busca por palavras no dicionário. A Figura 51 mostra o trecho do código embutido no NLPNET para normalização do texto. Da linha (3) até a linha (9) do código, é criado um *laço* para verificar cada caractere das palavras, em busca dos símbolos (“,”, “:”, “?”, “!”, “%”, “*”, “'”, “\”, “;”, “<”, “>” e “&”). Se houver, então é retirado o símbolo. Na linha (12), é criada uma *lista* com as palavras do texto. Na linha (13), é declarada uma variável do tipo *lista*. Da linha (14) até a linha (22), é criado um novo *laço*, mas este é apenas para separar símbolos e sinais de pontuação das palavras nos finais das frases.

Verificador Ortográfico - Para realizar a correção ortográfica, utilizamos o *Hunspell*. Ele é um verificador ortográfico e analisador morfológico, projetado para línguas com rica morfologia e com complexa composição de palavras ou codificação de caracteres. Esse programa funciona juntamente com um dicionário disponível em várias línguas. O *Hunspell* é utilizado por muitos *softwares*, entre eles LibreOffice¹³, OpenOffice.org¹⁴ e Mozilla Firefox¹⁵ (NÉMETH, 2017).

Quando uma palavra é submetida ao *Hunspell*, ele consulta o seu dicionário para ve-

¹¹ <<http://www.nltk.org/api/nltk.tokenize.html>>

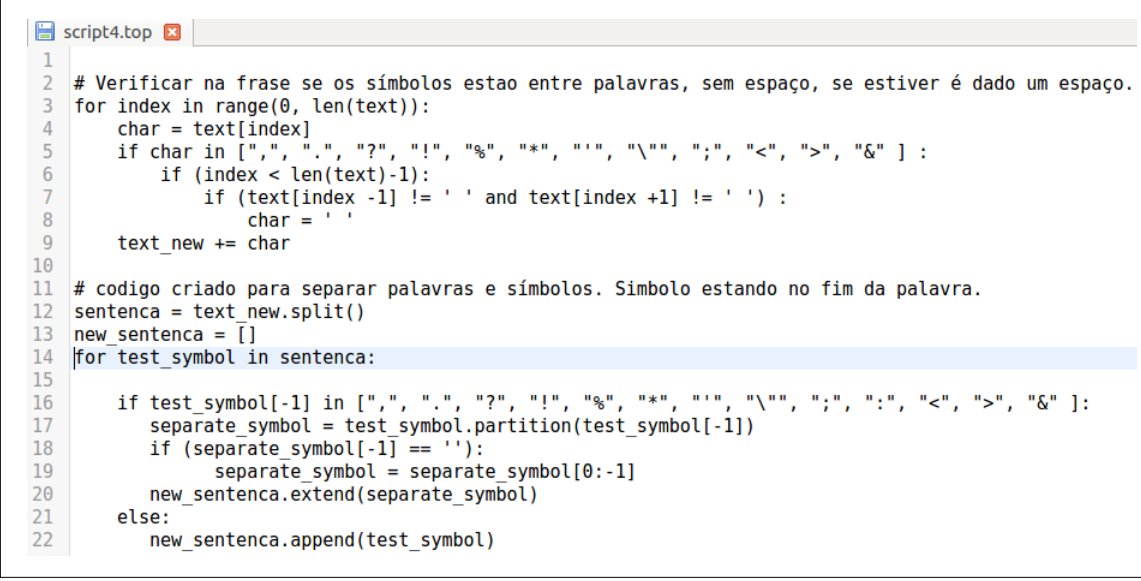
¹² <<http://www.numpy.org/>>

¹³ <<http://www.libreoffice.org/>>

¹⁴ <<http://www.openoffice.org/>>

¹⁵ <<https://www.mozilla.org/pt-BR/firefox/new/>>

Figura 51 – Código de Normalização do Texto do Usuário.



```

1
2 # Verificar na frase se os símbolos estão entre palavras, sem espaço, se estiver é dado um espaço.
3 for index in range(0, len(text)):
4     char = text[index]
5     if char in [",", ".", "?", "!", "%", "*", "'", "\"", ";", "<", ">", "&"] :
6         if (index < len(text)-1):
7             if (text[index - 1] != ' ' and text[index + 1] != ' ') :
8                 char = ' '
9         text_new += char
10
11 # código criado para separar palavras e símbolos. Símbolo estando no fim da palavra.
12 sentenca = text_new.split()
13 new_sentenca = []
14 for test_symbol in sentenca:
15
16     if test_symbol[-1] in [",", ".", "?", "!", "%", "*", "'", "\"", ";", ":", "<", ">", "&"] :
17         separate_symbol = test_symbol.partition(test_symbol[-1])
18         if (separate_symbol[-1] == ''):
19             separate_symbol = separate_symbol[0:-1]
20         new_sentenca.extend(separate_symbol)
21     else:
22         new_sentenca.append(test_symbol)

```

Fonte: Autoria Própria.

rificar a existência da palavra. Caso não exista, ele poderá sugerir uma lista de palavras que mais se assemelham à palavra procurada. Para melhorar a precisão dessa sugestão, implementamos junto aos comandos do Hunspell o algoritmo de Distância de Levenshtein (LEVENSHTEIN, 1966). Esse algoritmo funciona verificando o menor número de operações necessárias para transformar uma palavra em outra. As operações podem ser de substituição, inserção ou deleção de caracteres para realizar essa transformação. A Figura 52 mostra o trecho do código com o uso da ferramenta Hunspell e o algoritmo de Distância de Levenshtein implementados dentro da biblioteca NPLNET.

Na linha (2) da figura, é declarado um *laço* para verificar cada palavra da frase do usuário e buscar se ela existe no dicionário. Na linha (3), cada palavra é codificada com o mesmo padrão unicode¹⁶ do dicionário. Na linha (4), é “perguntado” ao dicionário se a palavra existe. Se existir, ela é então adicionada à frase final, se ela não existir, então o algoritmo busca no dicionário sugestões de palavras similares, linha (7). Na linha (8), é verificado se há sugestões, se não houver sugestões, então a palavra é adicionada novamente na frase final, sem modificações, linha (24). Se houver sugestões, é criado um novo *laço* para verificar cada sugestão, linha (9). Cada sugestão é submetida à função de distância de Levenshtein, linha (11). Da linha 13 à (20) é verificado qual das palavras possui o número menor de operações para ser transformada na palavra original. A palavra que tiver sofrido o menor número de operações substitui então a palavra original que não foi encontrada no dicionário.

Da linha (28) até a (42), podemos ver o algoritmo de Levenshtein implementado através da função *minimumEditDistance* (ROSETTACODE.ORG, 2017). Na linha (29), é verificado

¹⁶ <<https://pt.wikipedia.org/wiki/Unicode>>

se a primeira palavra é maior do que a segunda, se for, na linha (30) elas são então trocadas de ordem. Na linha (31), é declarada uma *lista* com o tamanho da primeira palavra. Na linha (32), é declarado um *laço* para varrer cada caractere da segunda palavra. Na linha (33) é criado uma nova *lista* *newDistances*. Na linha (34), é declarado um *laço* para varrer cada caractere da primeira palavra. Da linha (35) até a linha (42), é comparado cada caractere da primeira palavra com todos os caracteres da segunda palavra, para caracteres iguais é guardado o seu índice na *lista*, para caracteres diferentes é somado um ponto na *lista*. No final, é retornado o último valor da *lista* *distance* que está em ordem decrescente. Esse valor representa o número menor de operações para transformação de uma palavra em outra.

Figura 52 – Trecho do Código do Hunspell e do Algoritmo de Distância de Levenshtein.

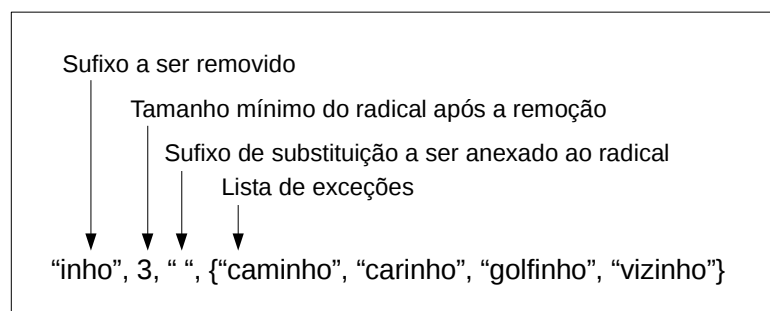
```
script5.top x
1 #Laço para verificar cada palavra da sentença e submetê-la ao Hunspell
2 for palavra in new_seneca:
3     palavra = palavra.encode(dic.get_dic_encoding())
4     if (dic.spell(palavra)):
5         frase.append(palavra.decode(dic.get_dic_encoding()))
6     else:
7         sugestoes = dic.suggest(palavra)
8         if (sugestoes):
9             for index in range(0, len(sugestoes)):
10
11                 resultLeven = self.minimumEditDistance(palavra,sugestoes[index])
12
13                 if (index == 0):
14                     compareResultLeven = resultLeven
15                     auxPalavra = sugestoes[index]
16                 else:
17
18                     if (resultLeven < compareResultLeven):
19                         compareResultLeven = resultLeven
20                         auxPalavra = sugestoes[index]
21
22                 frase.append(auxPalavra.decode(dic.get_dic_encoding()))
23     else:
24         frase.append(palavra.decode(dic.get_dic_encoding()))
25
26 #função para calcular a distancia de Levenshtein para verificar entre as palavras
27 #sugeridas do Hunspell qual mais se assemelha com a palavra digitada pelo usuário
28 def minimumEditDistance(self,s1,s2):
29     if len(s1) > len(s2):
30         s1,s2 = s2,s1
31     distances = range(len(s1) + 1)
32     for index2,char2 in enumerate(s2):
33         newDistances = [index2+1]
34         for index1,char1 in enumerate(s1):
35             if char1 == char2:
36                 newDistances.append(distances[index1])
37             else:
38                 newDistances.append(1 + min((distances[index1],
39                                                 distances[index1+1],
40                                                 newDistances[-1])))
41
42         distances = newDistances
43     return distances[-1]
```

Fonte: Autoria Própria.

Stemming - Para realizar o *stemming* nas palavras, utilizamos o RSLP Stemmer (Removedor de Sufixos da Língua Portuguesa) (ORENGO; HUYCK, 2001b). O RSLP Stemmer é um algoritmo implementado na linguagem de programação “C” e está disponível através

do NLTK. Esse algoritmo possui um conjunto de regras que são aplicadas de acordo com a palavra analisada (ORENGO; HUYCK, 2001b). A Figura 53 ilustra uma regra implementada pelo algoritmo RSLP Stemmer.

Figura 53 – Regra do Algoritmo RSLP Stemmer.



Fonte: Traduzido de Orengo e Huyck (2001a).

POS-Tagging - No *chatbot* Beck, utilizamos a funcionalidade do *POS-Tagging* da biblioteca NLPNET. A Figura 54 mostra um exemplo de uso do NLPNET através da linha de comando do Python. Na linha (1) é importada a biblioteca. Na linha (3) é definido o local onde se encontra os corpora a ser utilizado. Na linha (4), é criada uma instância da classe *POS-Tagger*. Na linha (5) é executado o método *tag* para etiquetar as palavras com suas classes gramaticais para a frase “O rato roeu a roupa do rei de Roma.”. Das linhas (6) à (8) podemos ver o resultado da execução do método *tag*.

Figura 54 – Exemplo do Uso do NLPNET.

```

script6.top
1 >>> import nlpnet
2 >>>
3 >>> nlpnet.set_data_dir('/usr/local/share/nltk_data/corpora/pos-pt')
4 >>> tagger = nlpnet.POSTagger()
5 >>> tagger.tag('O rato roeu a roupa do rei de Roma.')
6 >>> [(('O', 'ART'), ('rato', 'N'), ('roeu', 'V'), ('a', 'ART'),
7 >>> ('roupa', 'N'), ('do', 'PREP+ART'), ('rei', 'N'), ('de', 'PREP'),
8 >>> ('Roma', 'NPROPN'), ('.', 'PU'))]
  
```

Fonte: Baseado em Fonseca e Rosa (2013b).

4.4 Funcionamento do Sistema

O funcionamento do *chatbot* Beck, para um melhor entendimento, pode ser explanado em fases. Essas fases contemplam desde a captura da entrada do usuário até o fornecimento de uma resposta. O pré-processamento captura a entrada do usuário e realiza uma preparação do texto para o posterior processamento. A fase de processamento busca na base de

conhecimento do *chatbot* a regra mais apropriada de acordo com a entrada do usuário. E por fim, a fase de pós-processamento realiza os tratamentos finais para apresentar a resposta ao usuário. Essas fases são detalhadas nas seções a seguir.

4.4.1 Pré-Processamento

Inicialmente, o sistema recebe a entrada do usuário, que pode ser através de texto e/ou voz. Em seguida, são realizados os processos de normalização do texto e correção ortográfica, aplicação do algoritmo de *Stemming*, identificação das categorias gramaticais (*POS-Tagging*) e, se existir, associação de conceitos às palavras. A seguir falaremos um pouco sobre cada um desses processos.

- **Reconhecimento de Voz** - Se a entrada do usuário for através de voz, o sistema aciona o módulo de Reconhecimento de Voz para realizar a transformação dessa entrada de voz para texto.
- **Normalização e Correção Ortográfica** - Esse processo é responsável por separar os sinais de pontuação e outros símbolos das palavras, e realizar a correção ortográfica das palavras não encontrada no dicionário.
- **Redução da palavra ao Radical** - Através de um algoritmo de *Stemming*, a palavra é reduzida a um radical. Esse radical não é necessariamente a raiz da palavra na língua portuguesa. O radical aqui, pode ser interpretado como a parte suficiente da palavra que pode identificar outras palavras que possui essa parte em comum. Exemplo, a palavra “incerteza” possui o radical “cert” na língua portuguesa. No entanto, o algoritmo de *Stemming* usado em Beck fornece como resultado o radical “incert”. O algoritmo de *Stemming* é essencial para construção das regras de casamento de padrão no *ChatScript*, pois permite a construção de um número bem menor de regras para casar com a entrada do usuário.
- **Identificação das Categorias Gramaticais** - O *POS-Tagging* é responsável por etiquetar cada palavra de acordo com a categoria gramatical a que ela pertence. É importante lembrar que uma palavra pode pertencer a mais de uma categoria gramatical. Por exemplo: a palavra *melhor* em: *O mel é melhor do que o açúcar para a saúde.*; ou *O melhor é você chegar cedo.* Um bom etiquetador gramatical deve verificar a categoria correta da palavra de acordo com o papel que ela exerce na frase.
- **Conceitos** - Conceitos são conjuntos de palavras que podem representar sinônimos ou palavras afiliadas. Esse processo anexa às palavras da sentença do usuário o conceito a que ela pertence. Por exemplo, a palavra “mãe” pode estar associada ao conceito “família”. Uma palavra pode estar associada a mais de um conceito. A

categoria gramatical também é representada através de um conceito que é associado a palavra. Foi desenvolvida uma base com 50 conceitos e aproximadamente 3.000 palavras e expressões associadas para o *chatbot* Beck. Essa base de conceitos foi desenvolvida, em parte, com base no tesauro disponibilizado pela página da Internet *Thesaurus da Língua Portuguesa do Brasil* (JORGE, 2004).

4.4.2 Processamento

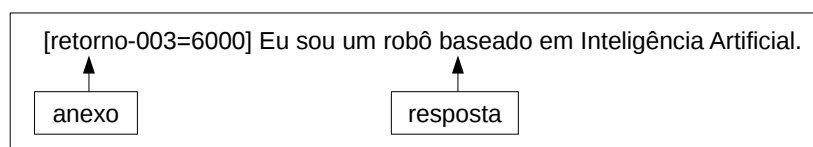
Nessa fase, o sistema procura na base de diálogos a regra que melhor casa com a entrada do usuário. O casamento de padrão pode ser realizado com a palavra original, com a sua forma canônica ou com os conceitos associados a ela. O princípio básico de funcionamento dessa fase é “varrer” a frase do usuário em busca de palavras-chave. A entrada do usuário pode casar com vários padrões existentes na base, porém esse processo só finalizará quando for gerada uma resposta. Caso não seja encontrado nenhum padrão que case com a entrada do usuário, o sistema poderá emitir um *gambit* (visto na seção 2.3.2), e caso não exista *gambits*, o sistema poderá recuperar a resposta padrão que deve estar cadastrada previamente na base de diálogos.

Após selecionar o padrão mais adequado para a entrada do usuário, o sistema deve recuperar uma das resposta que está relacionada a esse padrão. Para um mesmo padrão pode existir uma ou mais respostas associadas.

Para cada resposta recuperada, pode existir um adendo a ela chamado de OOB (Out-Of-Band, ou em português “Fora da Banda”). O recurso OOB é utilizado para enviar informações extras numa resposta. Esse recurso é bastante interessante para enviar comandos de gestos e expressões corporais quando se está trabalhando com um ACI. No nosso caso, esse recurso é utilizado para informar ao sistema que ele precisa retornar o turno do diálogo para Beck porque será necessário dizer algo mais para o usuário.

A Figura 55 mostra um exemplo de uma resposta com um anexo. Esse anexo representa o comando que o sistema deve retornar o turno para o *chatbot* pois ele precisará dizer algo mais para o usuário (o número 003 é o código identificador desse retorno e o 6000 representa o tempo em milissegundos que determina em quanto tempo o turno deve ser retornado).

Figura 55 – Resposta para o Usuário com um Anexo (OOB).



Fonte: Autoria Própria.

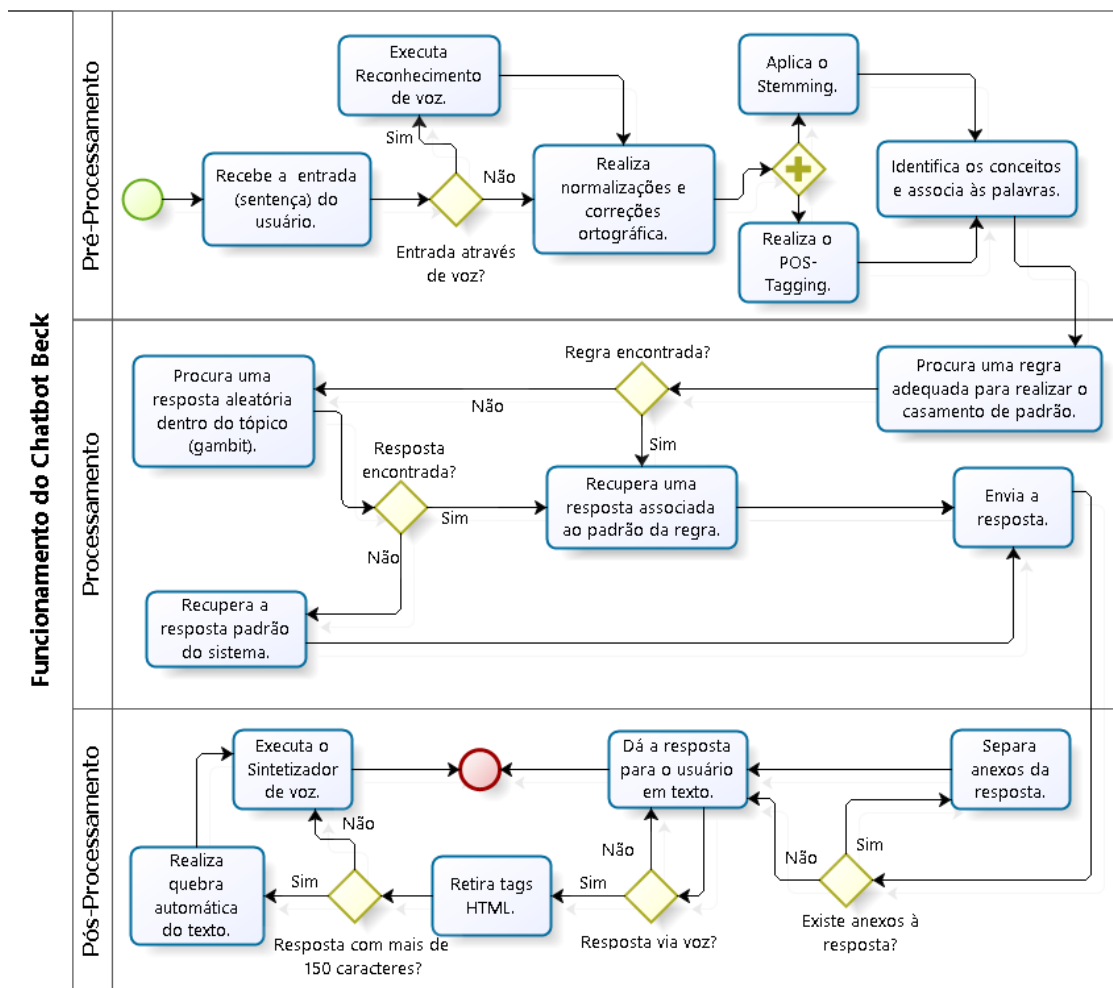
4.4.3 Pós-Processamento

Antes da resposta recuperada da base de conhecimento ser exibida para o usuário, o sistema realiza previamente a separação de anexos (OOB) para então exibi-la.

Depois da exibição da resposta em texto, o sistema verificará se ela deve ser gerada também via voz. Se afirmativo, a sentença é então tratada para retirada das *tags* HTML. Depois é verificada a quantidade de caracteres da resposta, se este ultrapassar a quantidade de 150 caracteres, então a sentença deve ser quebrada automaticamente antes de ser enviada para o sintetizador de fala.

A Figura 56 mostra essas três fases do funcionamento de Beck.

Figura 56 – Fases do Funcionamento do Sistema.



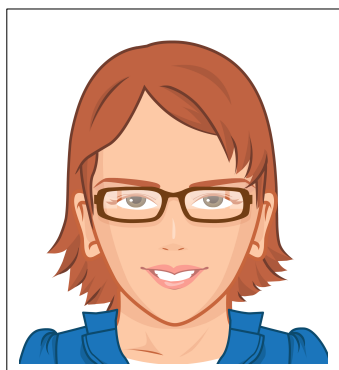
Fonte: Autoria Própria.

4.5 Interface com o Usuário

A interface, meio pela qual o usuário se comunica com o *chatbot*, foi criada buscando maximizar a interatividade e a experiência do usuário. Para se comunicar com o adolescente, Beck utiliza, além de texto e/ou voz, formulários, quadros e figuras.

O *chatbot* Beck é representado através de uma imagem feminina, como pode ser observado na Figura 57. A imagem feminina é muitas vezes descrita com qualidades como: gentil, útil e comunicativa. Em relação à imagem masculina, as qualidades geralmente são representadas pelas palavras: agressivo, vigoroso, competente e independente (BRAHNAM; ANGELI, 2012). Através de um estudo realizado, foi verificado que os *chatbots* com uma imagem feminina tiveram um número maior de conversas em relação àqueles representados por uma figura masculina. Entretanto, conversas com conteúdo sexual e uso de palavrões foram evidenciadas com uma maior frequência em *chatbot* com representação feminina (BRAHNAM; ANGELI, 2012).

Figura 57 – Representação do *Chatbot* Beck.



Fonte: Autoria Própria.

A tela inicial de Beck pode ser visualizada através da Figura 58. Nessa tela, podemos ver as opções de acesso ao sistema. Para acessar o *chatbot* Beck, se o usuário já possui um cadastro, basta preencher o e-mail e a senha informados no registro. Caso seja o primeiro acesso, será necessário realizar um cadastro no sistema. Ainda na tela inicial, o usuário tem disponível a opção *Esqueci minha senha*, que deve ser utilizada quando o usuário não lembrar mais a senha de acesso.

Por questão de segurança, os usuários devem ser identificados unicamente no sistema através do seu e-mail e senha de acesso. A senha cadastrada é criptografada no banco de dados, permitindo que apenas o usuário saiba e acesse a sua conta. Para criptografar a senha do usuário, usamos a função *cript* da linguagem PHP (PHP.NET, 2017). Os dados compartilhados pelo usuário durante a conversa com Beck são confidenciais e não são divulgados a ninguém.

Figura 58 – Tela Inicial de Beck.



Acesso ao Chatbot Beck

Informe seu e-mail

Senha

ACESSAR

[Criar Conta](#) [Esqueci minha senha](#)

Fonte: Autoria própria.

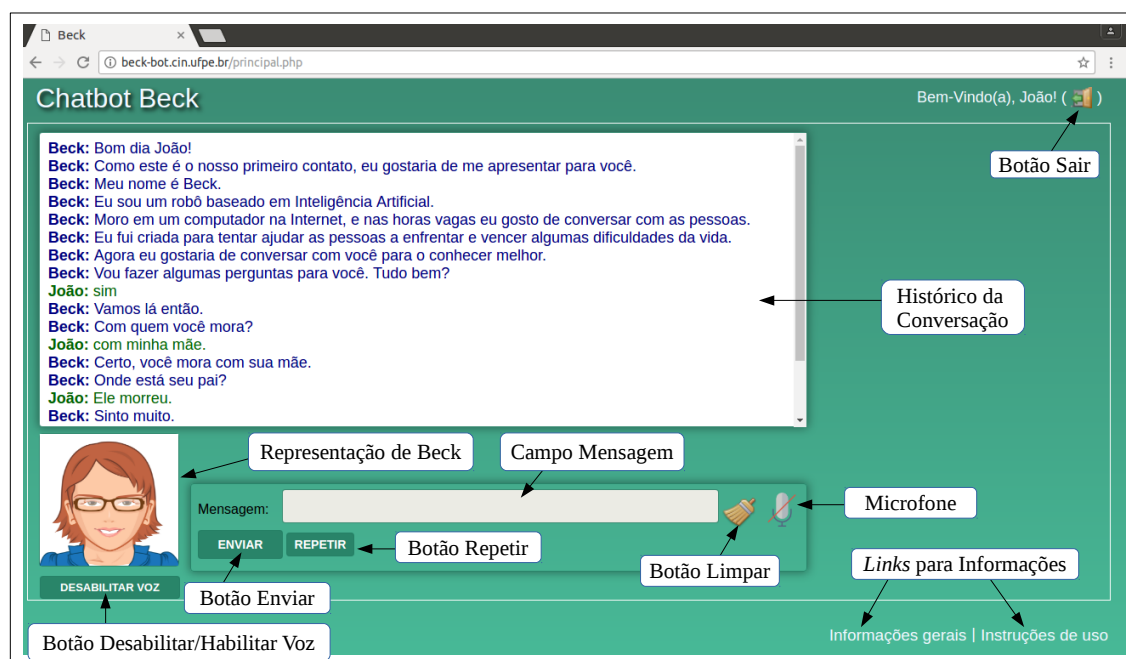
Após a autenticação no sistema, será exibida a tela principal de Beck. Na Figura 59 podemos ver essa tela.

Na tela principal (Figura 59), são disponibilizadas as seguintes opções:

- **Mensagem** - Utilizado pelo usuário para escrever uma mensagem;
- **Enviar** - Transmite a mensagem para Beck;
- **Repetir** - Repete a última pergunta realizada por Beck;
- **Limpar** - “Limpa” o campo mensagem;
- **Microfone** - Usado pelo usuário para se comunicar com Beck através do Reconhecimento de voz;
- **Desabilitar/Habilitar voz** - Desabilita e habilita a síntese e o reconhecimento de voz de Beck;
- **Informações gerais** - Exibe uma tela com informações gerais sobre Beck;
- **Instruções de uso** - Disponibiliza informações para que o usuário possa tirar eventuais dúvidas sobre como se comunicar com Beck; e
- **Sair** - Utilizado para sair do sistema. O usuário também pode encerrar a conversa com Beck usando palavras ou expressões de despedida, como: tchau, até mais, entre outras.

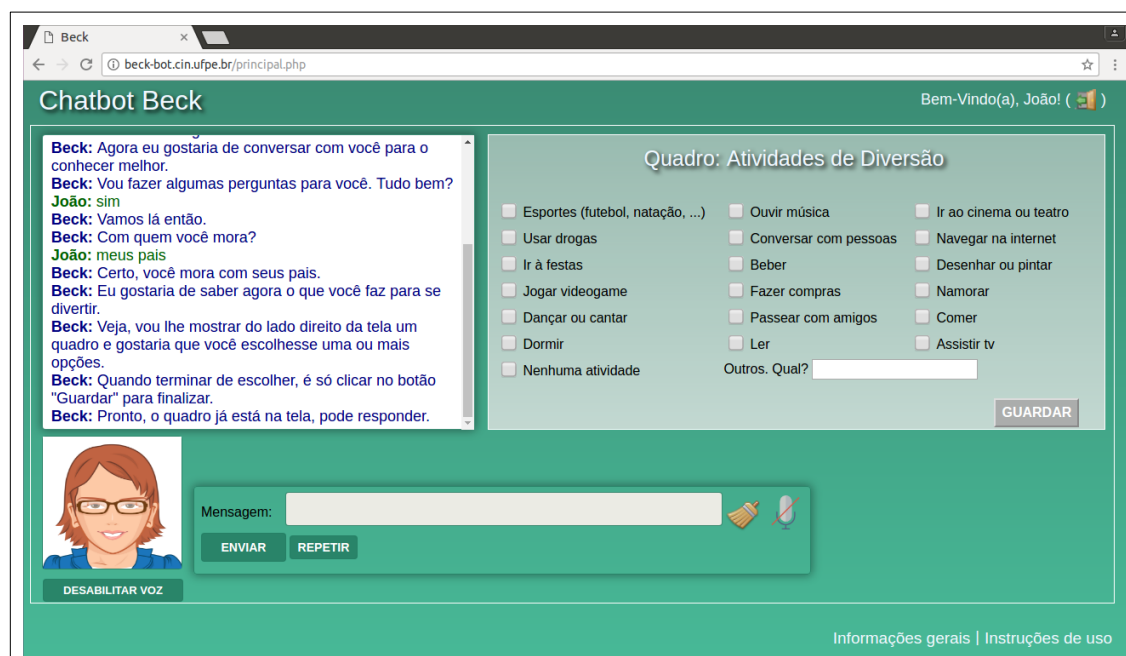
Na Figura 60, temos o exemplo de uma conversa realizada com Beck, na qual ela mostra um dos formulários para adolescente.

Figura 59 – Tela Principal de Beck.



Fonte: Autoria própria.

Figura 60 – Exemplo de Conversa com Beck.



Fonte: Autoria própria.

4.6 Testes do Sistema

Durante o desenvolvimento de Beck, foram realizados testes para validar cada tópico de diálogo implementado. Na medida em que cada tópico era finalizado, testes unitários eram realizados. O objetivo dos testes unitários é identificar erros de lógica e de implementação de cada módulo do sistema, separadamente (BARBOSA et al., 2000). Uma das vantagens desse teste é a possibilidade de realizar uma análise mais criteriosa e direcionada de uma unidade independente do restante do sistema. A Figura 61 mostra um exemplo de um caso de teste unitário realizado em Beck. Esse caso de teste foi feito para verificar se o usuário respondesse afirmativamente ao questionamento de Beck, então ela poderia iniciar as perguntas.

Figura 61 – Teste de Unitário - Início das Perguntas.

Entradas:
Condição Inicial: O <i>chatbot</i> Beck ter perguntado ao usuário se ele pode responder algumas perguntas.
Passos: O usuário responde positivamente usando alguma palavra que denote afirmação.
Saída:
Resultado esperado: Beck inicia os questionamentos perguntando ao usuário com quem ele mora.

Fonte: Autoria própria.

Como vimos, para a construção de Beck, foram utilizadas diferentes ferramentas e algoritmos. Todos os recursos utilizados no *chatbot* estão integrados, e a comunicação entre eles é realizada através de interfaces dos programas. Por exemplo, a comunicação da plataforma *Web* com o *ChatScript* é realizada através de *sockets*¹⁷. Para evitar erros de comunicação entre cada módulo de funcionamento de Beck, foram realizados testes de integração do sistema. Esse tipo de teste visa procurar falhas associadas às interfaces dos módulos do sistema, quando esses são integrados constituindo toda a estrutura do *software* (NETO, 2007). A Figura 62 mostra o caso de teste de integração realizado para verificar a comunicação do banco de dados com o *ChatScript* em uma das conversas realizadas por Beck.

Por fim, foram realizados testes de sistema para verificar o funcionamento do *chatbot* como um todo. Esse teste visa buscar falhas no sistema por meio da sua utilização, como se fosse um usuário final. A Figura 63 mostra um caso de teste de sistema utilizado para verificar o acesso do usuário ao *chatbot*.

¹⁷ <http://php.net/manual/pt_BR/function.fsockopen.php>

Figura 62 – Teste de Integração - Recuperação de Informações do Banco no *ChatScript*.

Entradas:
Condição Inicial: Atividades prazerosas que o usuário escolheu gravadas consistentemente no banco de dados.
Passos: Recuperar essas atividades no <i>ChatScript</i> através de uma <i>query</i> .
Saída:
Resultado esperado: O <i>chatterbot</i> consegue perguntar qual das atividades escolhidas pelo adolescente ele tem feito frequentemente.

Fonte: Autoria própria.

Figura 63 – Teste de Sistema - Acesso do Usuário ao *Chatbot*.

Entradas:
Condição Inicial: Conexão com a Internet; Cadastro do usuário efetivado e consistente no banco de dados.
Passos: Realizar o <i>login</i> de acesso no sistema.
Saída:
Resultado esperado: Beck dá as boas-vindas ao usuário usando o seu nome recuperado do banco de dados.

Fonte: Autoria própria.

A partir da perspectiva utilizada para realização dos testes, podemos classificá-la em duas técnicas de testes.

1. Técnica Estrutural (ou teste de Caixa-Branca) - Avalia o comportamento interno do *software*. Nessa técnica, o avaliador tem acesso ao código fonte do sistema e os casos de teste são elaborados analisando o código. O objetivo é testar todas as possibilidades de funcionamento dos componentes do *software*; e
2. Teste Funcional (ou teste de Caixa-Preta) - Nesse tipo de teste, o componente de *software* a ser testado é abordado como se fosse uma caixa-preta, ou seja, não se considera o comportamento interno do componente (NETO, 2007).

Os testes unitários e de integração realizados em Beck foram realizados a partir de uma perspectiva estrutural (caixa-branca), e o teste de sistema foi realizado a partir de

uma perspectiva funcional (caixa-preta).

4.7 Considerações Finais

Neste capítulo, apresentamos o *chatbot* Beck que foi construído com base na TCC. Vimos, que o *chatbot* Beck é um sistema de autoajuda, e seu principal aspecto é utilizar a linguagem natural para conversar com os adolescentes. Os diálogos elaborados para Beck foram estruturados a partir da TAC. Essa teoria evidencia estruturas e padrões existentes nas conversações na prática social. Os tópicos da TCC abordados por Beck têm o objetivo de mostrar para o adolescente que os pensamentos automáticos negativos podem influenciar as suas emoções e os seus comportamentos. Vimos também, os algoritmos e as ferramentas utilizadas para a construção de Beck, e a principal delas é a *ChatScript*. No entanto, foram realizadas adaptações para o uso do *ChatScript* na língua portuguesa. E, por fim, apresentamos os testes realizados em Beck.

5 AVALIAÇÃO E RESULTADOS

Neste capítulo, apresentamos e discutimos o *Survey* realizado para avaliar o desempenho e a utilidade do *chatbot* Beck após um teste de conversação realizado com adolescentes. Na seção 5.1, explanamos a metodologia adotada para realização do *Survey* com os adolescentes. Na seção 5.2, descrevemos o teste realizado e discutimos os resultados obtidos através do *Survey*. Por fim, a seção 5.3 traz as considerações finais sobre o que foi visto neste capítulo.

5.1 Metodologia

Nesta seção, descrevemos a metodologia adotada para avaliar o desempenho e a utilidade do *chatbot* Beck.

Para avaliar Beck com os adolescentes, definimos a **Questão de Pesquisa** a seguir.

Questão de Pesquisa

O desempenho e a utilidade de Beck são satisfatórios em uma conversação com adolescentes?

Para responder a essa questão, elaboramos um *Survey* que foi aplicado após um teste de conversação com Beck.

Survey é um método utilizado para descrever, comparar ou explicar conhecimentos, atitudes e comportamentos (PFLEEGER; KITCHENHAM, 2001). Através de um *Survey*, podemos coletar dados de indivíduos que respondem a uma série de questões sobre comportamentos e opiniões, muitas vezes sob a forma de um questionário. Esse é um dos métodos de pesquisa mais utilizados, buscando garantir aos indivíduos um alto nível de anonimato para que eles se sintam livres para expressar suas ideias e opiniões pessoais, sem o risco de serem identificados (LITTLE et al., 2012).

O *Survey* foi elaborado com 5 questões, são elas:

1. *Você considera que foi interessante conversar com Beck?*
2. *Você aprendeu alguma coisa com essa conversa?*
3. *Beck conseguiu entender as suas perguntas e respostas?*
4. *Você compreendeu que os seus pensamentos podem influenciar o que você sente e a forma como você se comporta?*
5. *Você conversaria novamente com Beck?*

Para os adolescentes responderem às questões, utilizamos a escala *Likert* (LIKERT, 1932) com um nível de gradação de cinco pontos sobre o grau de conformidade. Assim, utilizamos os valores de 1 a 5 para cada resposta, sendo: (1) Discordo Totalmente, (2) Discordo, (3) Não Tenho Certeza, (4) Concordo e (5) Concordo Totalmente.

Para possibilitar a aquisição de um perfil mínimo do adolescente, além de perguntarmos o sexo e a idade, fizemos também a seguinte pergunta: *Você já tinha conversado com um sistema parecido com Beck?*. Essa pergunta foi realizada para tentarmos verificar se o fato de o adolescente já ter tido uma experiência com *chatbots* influenciaria o resultado do *Survey*. Além disso, também incluímos no *Survey* um campo para comentários, para que eles fornecessem suas opiniões sobre a conversa realizada com Beck.

5.2 Realização do Teste de Conversação e do *Survey*

Nesta seção, explanamos como o teste de conversação foi realizado, e discutimos os resultados obtidos com o *Survey*.

Para um melhor entendimento sobre como foi realizado o teste de conversação e o *Survey*, dividimos a explicação em etapas.

Etapas 1: Adequação da Interface de Beck para Realização do Teste

Para garantir o sigilo e o anonimato dos adolescentes na realização do teste, criamos uma nova tela de cadastro para Beck e a substituímos pela anterior (a tela de cadastro anterior pode ser observada no capítulo 4). Essa tela foi criada para solicitar apenas o sexo, a idade e um nome fictício do adolescente, que já é sugerido pelo sistema no momento da escolha do sexo. Se o adolescente selecionar a opção *Masculino*, o sistema já sugere o nome *João*. Se for escolhido o sexo *Feminino*, é sugerido *Joana*. O adolescente pode alterar o nome sugerido, se assim o desejar. A Figura 64 mostra a tela inicial de Beck criada para a realização do teste.

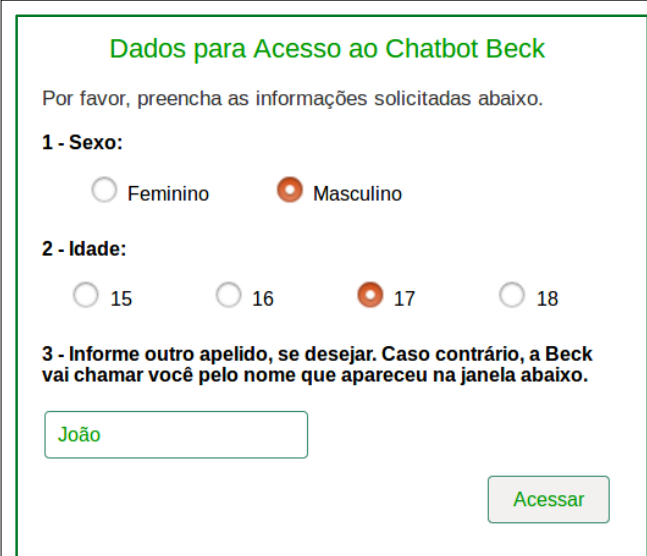
O *chatbot* também foi alterado para inclusão do *Survey* aplicado no final do teste de conversação. Além das 5 questões elaboradas para o *Survey*, incluímos também no *Survey* a pergunta se o adolescente já tinha conversado com um sistema parecido com Beck, citada anteriormente.

Etapas 2: Solicitação de Autorização da Escola e Identificação do Ambiente Computacional para Realização do Teste

O teste de conversação de Beck e o *Survey* foram realizados com adolescentes entre 15 e 18 anos de idade, estudantes de duas Escolas Públicas do Estado de Pernambuco.

Pedimos autorização aos responsáveis pelas escolas para realização do Teste. Enviamos a cada escola uma *Solicitação para Realização de Teste de Software* para explicar sobre o

Figura 64 – Tela Inicial de Beck Criada para Realização do Teste.



Dados para Acesso ao Chatbot Beck

Por favor, preencha as informações solicitadas abaixo.

1 - Sexo:

☐ Feminino ☒ Masculino

2 - Idade:

☐ 15 ☐ 16 ☒ 17 ☐ 18

3 - Informe outro apelido, se desejar. Caso contrário, a Beck vai chamar você pelo nome que apareceu na janela abaixo.

Fonte: Autoria Própria.

projeto e as questões do *Survey* que seria realizado com os adolescentes. Essa solicitação pode ser vista no Apêndice C deste trabalho.

Informamos às escolas que o *chatbot* foi desenvolvido para ser utilizado em computadores com acesso à Internet, através de Navegadores *Web*. Portanto, antes de realizar o teste com os adolescentes, precisaríamos saber se os laboratórios das escolas atendiam a esses requisitos.

Fizemos uma visita prévia a uma das escolas para verificar os computadores. Não foi necessário visitar a outra escola, pois ela participa de um projeto patrocinado pela Secretaria de Educação Estadual chamado *Pernambucoders*¹, assim a escola possui computadores em um bom estado de uso e com acesso à Internet.

Foi identificado que as escolas não usavam computadores do tipo *Desktop*, e sim um modelo de *Netbook*. Foi-nos fornecido um equipamento igual aos que as escolas utilizavam. Esse equipamento foi identificado como um *Netbook* modelo MOBO 5950 da POSITIVO². O equipamento atende a todos os requisitos para o funcionamento de Beck, exceto a resolução do vídeo que é mais baixa, tendo o impacto na exibição correta de alguns elementos da interface de Beck. Assim, foi necessário alterar os códigos responsáveis pela exibição da interface de Beck para atender também a esse modelo de equipamento.

Etapa 3: Realização do Teste

O Teste foi realizado com 38 adolescentes, nos dias 7, 8 e 12 de junho deste ano.

¹ <<http://www.educacao.pe.gov.br/portal/?pag=&cat=37&art=3195>>

² <http://www.positivoinformatica.com.br/uploads/files/Catalogos/Ficha_Tecnica_Positivo_Mobo_5950.pdf>

Antes de iniciar o teste, foi explicado para eles sobre o projeto, o tempo de realização do teste (em média 30min), a aplicação do questionário no final da conversação, e sobre o anonimato. Um total de 6 adolescentes desistiram do teste durante a sua realização.

Como eram, em média, 10 alunos realizando o teste ao mesmo tempo, foi pedido para eles que não usassem o recurso do microfone do computador (Reconhecimento de voz) para se comunicar com Beck, e para os que tinham fone de ouvido, solicitamos que usassem, diminuíssem o volume das caixas de som, ou mesmo desabilitassem a voz de Beck (Síntese de voz). Essa decisão foi tomada para preservar o sigilo de cada adolescente na troca de informações com Beck e para manter um ambiente mais tranquilo para todos.

No decorrer do teste, notamos lentidão no funcionamento de Beck em ambas as escolas, e em alguns equipamentos Beck parou de funcionar (possivelmente devido a problemas com a Internet). Após utilizar a opção *Recarregar a página* do Navegador Web, o *chatbot* voltou a funcionar.

5.2.1 Resultados

Um total efetivo de 32 adolescentes completou o teste e respondeu ao *Survey*. Desse total, 13 são do sexo masculino e 19 do feminino. O tempo médio para realização do teste e do *Survey* foi de 47 minutos.

Em relação à pergunta: *Você já tinha conversado com um sistema parecido com Beck?*, um total 84,37% dos adolescentes nunca haviam conversado com um sistema parecido com Beck, enquanto 15,63% responderam positivamente.

Apresentamos agora os resultados das questões realizadas no *Survey*.

Questão 1: *Você considera que foi interessante conversar com Beck?*

Um total de 56,25% dos adolescentes concordaram que foi interessante conversar com Beck, e 40,63% concordaram totalmente com essa questão. Um percentual de 3,13% respondeu não ter certeza se foi interessante conversar com o *chatbot*. Assim, podemos concluir que 96,88% dos adolescentes concordaram que foi interessante conversar com Beck.

Questão 2: *Você aprendeu alguma coisa com essa conversa?*

O percentual de 37,50% dos adolescentes concordaram que aprenderam algo conversando com Beck, 46,88% concordaram totalmente com essa questão, 9,38% não têm certeza se aprenderam algo, e 6,25% discordaram com essa questão. Portanto, concluímos que 84,38% dos adolescentes concordaram que aprenderam com Beck. Nenhum dos adolescentes discordou totalmente com a questão 2.

Questão 3: *Beck conseguiu entender as suas perguntas e respostas?*

Do total de adolescentes, 50% concordaram que Beck conseguiu entender suas perguntas e respostas, 31,25% concordaram totalmente que Beck entendeu suas perguntas e

respostas, 15,63% não têm certeza que Beck conseguiu entender e 3,13% discordaram que Beck entendeu suas perguntas e respostas. Portanto, 81,25% dos adolescentes concordaram que Beck conseguiu entender suas perguntas e respostas, e apenas 3,13% discordaram. Nenhum dos adolescentes discordou totalmente com a questão 3.

Questão 4: *Você compreendeu que os seus pensamentos podem influenciar o que você sente e a forma como você se comporta?*

Um total de 68,75% dos adolescentes concordaram que compreenderam que os pensamentos podem influenciar o sentimento e o comportamento e 31,25% dos adolescentes concordaram totalmente com essa questão. Portanto, 100% dos adolescentes concordaram com a questão 4.

Questão 5: *Você conversaria novamente com Beck?*

Um percentual de 46,88% dos adolescentes concordaram que conversariam com Beck novamente, 37,50% dos adolescentes concordaram totalmente com essa questão, 6,25% não têm certeza se conversariam com Beck novamente, 6,25% discordaram dessa pergunta e 3,13% discordaram totalmente que repetiriam essa atividade. Assim, podemos concluir que 84,38% dos adolescentes concordaram que conversariam com Beck em uma nova oportunidade e apenas 9,38% não conversariam novamente. Acreditamos que esse resultado é muito positivo, pois trata-se de um tipo de interação novo para muitos deles, e que aborda um tema em geral esquecido pelos adultos (depressão na adolescência).

Na Tabela 4, podemos ver os resultados das 5 perguntas realizadas no *Survey*.

Por uma questão de espaço, só colocamos as iniciais das opções de respostas no cabeçalho das colunas. Assim, **DT** significa **Discordo Totalmente**, **D** significa **Discordo**, **NTC** quer dizer **Não Tenho Certeza**, **C** representa **Concordo** e **CT** significa **Concordo Totalmente**. Para cada coluna de resposta, podemos ver o percentual de adolescentes que as responderam. Na última linha da tabela, temos uma média calculada dos percentuais das respostas pelo número de questões realizadas.

Tabela 4 – Resultado das Questões Realizadas no *Survey*.

Questão	DT	D	NTC	C	CT
1	0%	0%	3,13%	56,25%	40,63%
2	0%	6,25%	9,38%	37,5%	46,88%
3	0%	3,13%	15,63%	50%	31,25%
4	0%	0%	0%	68,75%	31,25%
5	3,13%	6,25%	6,25%	46,88%	37,5%
Média	0,63%	3,13%	6,88%	51,88%	37,50%

Fonte: Autoria Própria

A **Questão de Pesquisa** (*O desempenho e a utilidade de Beck são satisfatórios em uma conversa com adolescentes?*) que norteou o *Survey* realizado com os adolescentes, foi dividida em dois elementos principais, o desempenho e a utilidade. Em relação a esses elementos, temos os seguintes resultados conforme evidencia a Tabela 5.

Tabela 5 – Resultado em Relação ao Desempenho e a Utilidade de Beck.

Elemento	D	NTC	C
Desempenho	3,91%	10,16%	85,94%
Utilidade	3,13%	4,69%	92,19%

Fonte: Autoria Própria

Para calcular o desempenho de Beck, somamos os resultados das respostas da questão 1 (*Você considera que foi interessante conversar com Beck?*), da questão 3 (*Beck conseguiu entender as suas perguntas e respostas?*) e da questão 5 (*Você conversaria novamente com Beck?*), e calculamos a média. Agrupamos as respostas *Discordo Totalmente* e *Discordo* no resultado *Discordo*. A resposta *Não tenho Certeza* permaneceu com a mesma nomenclatura de resultado e, *Concordo* e *Concordo Totalmente* em *Concordo*.

Conforme mostra a Tabela 5, 85,94% dos adolescentes concordaram que Beck teve um desempenho satisfatório, contra 3,91% dos adolescentes que discordaram. O elemento “desempenho” foi traduzido em relação ao interesse dos adolescentes em conversar com Beck, e nos erros de interpretação das perguntas e do processamento das respostas cometidos por Beck.

Para calcular a utilidade de Beck realizamos o processo similar ao cálculo do resultado do desempenho. Somamos os resultados das respostas da questão 2 (*Você aprendeu alguma coisa com essa conversa?*) e da questão 4 (*Você compreendeu que os seus pensamentos podem influenciar o que você sente e a forma como você se comporta?*) do *Survey*, e calculamos a média.

De acordo com a Tabela 5, 92,19% dos adolescentes concordaram que Beck foi útil para eles, e 3,13% dos adolescentes discordaram da sua utilidade. O elemento “utilidade” foi traduzido em relação ao aprendizado adquirido pelos adolescentes.

5.2.2 Discussão

Acreditamos que o ideal seria realizar a avaliação de Beck com adolescentes diagnosticados com depressão. Contudo, devido às dificuldades em identificar esse público e ainda para evitar possíveis problemas éticos, optamos por realizá-la com adolescentes que estivessem na faixa etária do nosso público-alvo (independentemente de terem sido ou não diagnosticados com depressão).

A realização do teste de conversa de Beck com vários adolescentes em uma mesma sala pode ter impactado na veracidade das informações trocadas com o *chatbot*. Em alguns

momentos, com a surpresa dos adolescentes em conversar com Beck, eles dialogaram entre si, comentando sobre as perguntas e respostas de Beck.

Um fator que prejudicou a realização do teste de conversação foi a lentidão das falas e do processamento das respostas por Beck. Essa lentidão foi causada pela baixa velocidade da Internet. Alguns adolescentes que desistiram do teste alegaram que o sistema estava muito lento, e que, não poderiam esperar muito. Mesmo aqueles que fizeram todo o teste e o *Survey*, reclamaram bastante da lentidão apresentada durante a interação com o *chatbot*.

Em uma das escolas, a perda da conexão com a Internet foi contínua. Consequentemente, houve interrupção do teste em vários momentos. Assim, os adolescentes tinham que constantemente *Recarregar* o Navegador *Web* para prosseguir conversando com Beck.

Na outra escola, houve uma maior desistência, 4 no total, pois os alunos estavam na semana de provas. Desse modo, o horário em que eles permaneciam na escola estava reduzido. No primeiro dia do teste, realizamos a avaliação de Beck após a prova. Nesse dia houve uma maior desistência (3 adolescentes). No segundo dia, o teste foi realizado antes da prova, havendo apenas uma desistência. A quantidade total de adolescentes envolvidos no teste, incluindo os desistentes, foi de 38 adolescentes.

Realizamos uma análise parcial no *log* de registro das conversas para verificar os erros de perguntas e respostas cometidos por Beck. Mostramos alguns deles na Tabela 6, onde incluímos comentários explicativos sobre o problema identificado.

Uma base de diálogos precisa ser criteriosamente construída para minimizar os erros de interpretação do *chatbot*. Quando o objetivo do *chatbot* é apenas conduzir uma conversa fluente, sem uma real necessidade de conversar sobre um tópico específico e adquirir informações a partir do diálogo, podemos utilizar truques para manter a conversa sem interrupções. Um truque simples, usado por muitos *chatbots*, é mudar o foco da conversa ou fazer uma piada (MAULDIN, 1994).

A base de diálogos atual de Beck precisa ser melhorada, não há dúvidas. Contudo, apesar das dificuldades, os resultados do *Survey* foram consistentemente positivos.

Como visto anteriormente, o *Survey* oferecia um espaço para comentários. Vejamos alguns deles na Tabela 7.

O comentário 6 nos intrigou um pouco, pois consideramos bom o sintetizador de voz utilizado em Beck. Esse comentário foi realizado por um adolescente da primeira escola. Ao realizar o teste de conversação na segunda escola, percebemos que a síntese de voz estava diferente. A pronúncia de Beck era de uma falante nativa do inglês americano, tentando falar o português. Como a maioria dos adolescentes da primeira escola tinha fones de ouvido, não percebemos o problema.

Já tínhamos a informação de que alguns computadores da segunda escola, embora estivessem em um bom estado de uso, passavam muito tempo desligados. Ao verificar a versão do Navegador *Web* Google Chrome, percebemos que era a versão 39, enquanto a versão atual é 58. Então, acreditamos que o computador utilizado pelo adolescente que

Tabela 6 – Alguns Erros na Interpretação de Perguntas e Respostas Cometidos por Beck.

Pergunta 1:	Você se incomoda em me dizer qual é a sua orientação sexual? Se não ficar à vontade para falar sobre isso, não precisa responder.
Resposta 1:	Não desejo falar.
Comentário 1:	Beck não estava com a palavra “desejo” mapeada em sua base de diálogos, como uma possível resposta. Ela poderia responder às palavras: querer, ir, responder, falar, vontade, poder, dizer, e suas variações.
Pergunta 2:	Sempre que conversarmos, vou lhe pedir para respondê-lo. Tudo bem? (pergunta relacionada ao quadro <i>Como você está se sentindo agora?</i>).
Resposta 2:	Talvez.
Comentário 2:	A palavra “talvez” estava mapeada, porém faltava inserir um símbolo na base de diálogos para tornar todas as palavras da regra em questão, opcionais.
Pergunta 3:	Qual dessas atividades você tem feito mais frequentemente? namorar, conversar, ouvir música, comer e dormir.
Resposta 3:	Ouvir Musica.
Comentário 3:	Por existir também a palavra música sem acento, Beck precisaria estar com essa palavra mapeada.
Pergunta 4:	Você tem namorado ou namorada?
Resposta 4:	Nenhum dos dois.
Comentário 4:	Essa resposta não estava mapeada na base de diálogos.
Pergunta 5:	Você tem namorado ou namorada?
Resposta 5:	Quase.
Comentário 5:	Essa resposta também não estava mapeada na base de diálogos.
Pergunta 6:	Qual dessas atividades você tem feito mais frequentemente? Dormir, comer, namorar, navegar na internet e conversar com pessoas.
Resposta 6:	A sei lá.
Comentário 6:	Resposta não mapeada na base.

Fonte: Autoria Própria

fez o comentário sobre a voz de Beck estivesse também desatualizado, justificando assim o comentário do adolescente.

Beck foi criada para realizar várias sessões de conversa com o mesmo adolescente, explorando as informações que ela adquiriu do perfil dele, na primeira sessão. A extensão dessa base de diálogos está nos nossos trabalhos futuros. Alguns adolescentes, embora não tenham se expressado nos comentários escritos, afirmaram que Beck falava muito e eles pouco.

Explicamos para eles que era o primeiro contato de Beck com usuários reais e que, portanto, nas próximas interações, a conversa será mais dinâmica, e que Beck conversará sobre as perguntas respondidas por cada indivíduo, e assim eles teriam mais oportunidades

Tabela 7 – Comentários dos Adolescentes no *Survey* Realizado.

Identificador	Comentário
1	“muito bom conversar”
2	“que foi bastante interessante”
3	“Eu amei falar com a Beck e se tivesse outra oportunidade gostaria de falar com ela de novo. Parabéns ao professor que criou ela...”
4	“repete varias vezes pensamentos negativos”
5	“foi bom mim fez refletir sobre os pensamentos..”
6	“poderia bota um voz mais realista que motivase mais a pessoa a conversar com ela”
7	“Gostei!! Pelo menos ela me escuta Kkkkj”
8	“foi ótimo”
9	“Gostei muito da experiencia”

Fonte: Autoria Própria

de ter de uma conversa mais interessante.

Em relação à pergunta *Você já tinha conversado com um sistema parecido com Beck?*, não encontramos impactos diretos no resultado do *Survey* realizado.

Apesar das dificuldades na realização da avaliação de Beck, consideramos os resultados satisfatórios. O desempenho e a utilidade de Beck mostraram um bom nível de aceitação. Em relação ao desempenho, 85,94% dos adolescentes ficaram satisfeitos, contra 3,91% que não ficaram satisfeitos. Em relação à utilidade, 92,19% dos adolescentes disseram que aprenderam algo com Beck, assim evidenciando a sua utilidade.

5.3 Considerações Finais

Apresentamos, neste capítulo, o teste de conversação e o *Survey* realizados com 32 adolescentes de duas Escolas Públicas do Estado de Pernambuco. O teste e o *Survey* foram realizados para avaliar o desempenho e a utilidade de Beck. Embora com alguns problemas de lentidão e de alguns erros de interpretação de perguntas e de respostas cometidos por Beck, podemos afirmar que os resultados foram positivos. Os resultados apresentados mostraram que Beck alcançou bons níveis de aceitação, 85,94% dos adolescentes ficaram satisfeitos com o desempenho de Beck e, em relação à utilidade, Beck teve um resultado de 92,19% de aprovação por parte dos adolescentes.

6 CONCLUSÃO

A depressão é um grave problema de saúde pública, e pode levar o indivíduo ao suicídio. Essa doença acomete várias pessoas. Estima-se que mais 322 milhões de pessoas no mundo sofrem de depressão. No Brasil, o suicídio é a terceira principal causa de morte entre os jovens.

A depressão pode ser tratada de diversas formas, contudo, a maior parte das pessoas acometidas pela depressão ainda não tem acesso a um tratamento adequado. Em vários países, é recomendado o uso da TCC como primeira forma de tratamento na depressão leve à moderada, e, como vimos, esse tipo de terapia pode também ser disponibilizado através do computador. No Brasil, o acesso à TCC via computador ainda não é uma realidade, embora existam estudos em outros países que comprovem a sua eficácia.

A maioria dos sistemas de computador que oferece a TCC o faz de maneira tradicional (leia-se tradicional como uma simples página *Web*), no qual os conteúdos da terapia são disponibilizados ao usuário sem o uso da língua natural como principal forma de interação. As interfaces em linguagem natural, como os *chatbots*, têm se tornado uma tendência de uso por permitirem uma interação entre o usuário e o computador utilizando alguma linguagem natural. Isso, em parte, pode ser explicado pelos grandes investimentos que estão sendo realizados por empresas multinacionais. O PLN proporciona a interação entre o homem e o computador de uma forma mais simples e natural.

Este trabalho de mestrado desenvolveu o *chatbot* Beck, que oferece acesso aos conteúdos da TCC, através do uso da linguagem natural, para os adolescentes que têm depressão ou que desejam evitá-la. Mostramos, neste trabalho, que o desenvolvimento dos *chatbots* está fundamentado em PLN, uma subárea da IA. Vimos, que há várias linguagens para o desenvolvimento de *chatbots*, e fizemos a comparação entre as duas linguagens mais difundidas e bem sucedidas: o AIML e o *ChatScript*. Justificamos a escolha da linguagem *ChatScript* por ela apresentar melhores recursos. Mostramos também a arquitetura criada para disponibilizar Beck na Internet, visando facilitar o acesso dos usuários.

Para avaliar o desempenho e a utilidade do *chatbot* criado, realizamos um teste de conversação e um *Survey* com 32 adolescentes de duas Escolas Públicas do Estado de Pernambuco. Através das avaliações realizadas, concluímos que Beck apresentou resultados satisfatórios. Em relação ao desempenho, 85,94% dos adolescentes ficaram satisfeitos com Beck e, em relação à utilidade, Beck teve um resultado de 92,19% de aprovação por parte dos adolescentes.

6.1 Contribuições

Destacamos, a seguir, as principais contribuições do trabalho realizado e relatado nesta Dissertação de Mestrado:

- Desenvolvimento do primeiro *chatbot* para a língua portuguesa baseado na TCC, com objetivo de apoiar adolescentes com depressão, ou que desejam evitá-la.
- Realização de uma pesquisa comparativa entre os recursos oferecidos pelas linguagens de desenvolvimento de *chatbots* AIML e *ChatScript*;
- Desenvolvimento de uma arquitetura para utilização do *ChatScript* para a língua portuguesa, incluindo corretor ortográfico, algoritmo de *Stemming* e *POS-Tagging*;
- Criação de uma base de sinônimos e palavras afiliadas no *ChatScript*, com aproximadamente 3.000 palavras;

6.2 Limitações

Listamos, a seguir, algumas limitações deste trabalho:

- Não foi possível realizar uma avaliação do *chatbot* com adolescentes diagnosticados com depressão. Assim, não foi possível avaliar a utilidade de Beck para pessoas que realmente sofrem com esse distúrbio psicológico;
- Por questão de tempo, a quantidade de padrões de diálogo (entrada-saída) não está satisfatória;
- Não realizamos uma avaliação da velocidade (performance) de Beck em conversar com um número maior de usuários ao mesmo tempo (quando realizamos o teste de conversação nas escolas, havia, no máximo, 10 adolescentes conversando com Beck ao mesmo tempo). Essa avaliação é importante, pois a proposta de arquitetura montada para Beck é algo novo;
- Uso de um reconhecedor e sintetizador de voz restrito a um único navegador da Internet.

6.3 Trabalhos Futuros

O desenvolvimento de um sistema baseado em PLN, incluindo os *chatbots*, não é uma tarefa fácil, e requer tempo e dedicação. Apontamos a seguir algumas extensões que podem ser implementadas com o objetivo de trazer melhorias a este trabalho. Dentre essas propostas, destacamos:

- Desenvolver novos padrões de diálogo para o *chatbot*, para que ele seja capaz de conversar sobre as repostas capturadas a partir do perfil do adolescente, adquirido durante a primeira sessão de conversação;
- Avaliar o *chatbot* com adolescentes diagnosticados com depressão, através de um experimento controlado;
- Criar um avatar de Beck em terceira dimensão dotado de movimentos e expressões faciais, e corporais;
- Disponibilizar o *chatbot* para a plataforma *mobile*;
- Explorar outros reconhecedores e sintetizadores de voz para diferentes tipos de navegadores;
- Fazer testes de performance da arquitetura criada para Beck;
- Implementar o reconhecimento e utilização de *emojis* (PAIVA, 2016);
- Melhorar a correção ortográfica a partir da implementação de uma algoritmo fonético, como o *Soundex* (HOLMES; MCCABE, 2002).
- Desenvolver um novo módulo para Beck, que permita ao adolescente usá-la também como um diário, no qual ele possa relatar as suas experiências cotidianas.

REFERÊNCIAS

- ABDUL-KADER, S. A.; WOODS, J. Survey on chatbot design techniques in speech conversation systems. *(IJACSA) International Journal of Advanced Computer Science and Applications*, Citeseer, 2015.
- ABUSHAWAR, B.; ATWELL, E. Alice chatbot: trials and outputs. *Computación y Sistemas*, v. 19, n. 4, p. 625–632, 2015.
- ALLEN, J. *Natural Language Understanding*. New York, NY, USA: Benjamin-Cummings Publishing Co., Inc., 1995.
- ALLEN, J. F. Natural language processing. John Wiley and Sons Ltd., Chichester, UK, p. 1218–1222, 2003.
- ALMEIDA, A. M.; NETO, F. L. Revisão sobre o uso da terapia cognitiva-comportamental na prevenção de recaídas e recorrências depressivas. *Rev. Bras. Psiquiatr.*, São Paulo, SciELO Brasil, v. 25, n. 4, p. 239–244, 2003.
- ANGELI, A. D. To the rescue of a lost identity: Social perception in human-chatterbot interaction. p. 7–14, 2005.
- AVANCI, J. Q.; ASSIS, S. G.; OLIVEIRA, R. V.; FERREIRA, R. M.; PESCE, R. P. Fatores associados aos problemas de saúde mental em adolescentes. *Psicologia: Teoria e Pesquisa*, SciELO Brasil, v. 23, n. 3, p. 287–294, 2007.
- BAR, J. *Amazon Lex – Build Conversational Voice & Text Interfaces*. [S.l.]: Amazon Web Services, 2016. <<https://aws.amazon.com/de/blogs/aws/amazon-lex-build-conversational-voice-text-interfaces/>>. Acesso em: 20 abr. 2017.
- BARBOSA, E. F.; MALDONADO, J. C.; VINCENZI, A. M. R.; DELAMARO, M. E.; SOUZA, S.; JINO, M. Introdução ao teste de software. *Minicurso apresentado no XIV Simpósio Brasileiro de Engenharia de Software (SBES 2000)*, 2000.
- BARROS, F. A.; ROBIN, J. Processamento de linguagem natural. *Revista Eletrônica de Iniciação Científica*, v. 1, p. 1–61, 2001.
- BARROS, F. A.; TEDESCO, P. C. A. R. Agentes inteligentes conversacionais: Conceitos básicos e desenvolvimento. Porto Alegre: Sociedade Brasileira de Computação, p. 169–218, 2016.
- BECK, A. T. The current state of cognitive therapy: a 40-year retrospective. *Archives of General Psychiatry*, American Medical Association, v. 62, n. 9, p. 953–959, 2005.
- BECK, J. S. *Cognitive therapy: Basics and beyond*. [S.l.]: Guilford Press, 1995.
- BELLOTTI, F.; BERTA, R.; GLORIA, A. D.; LAVAGNINO, E. Towards a conversational agent architecture to favor knowledge discovery in serious games. p. 1–7, 2011.

- BICKMORE, T. W.; MITCHELL, S. E.; JACK, B. W.; PAASCHE-ORLOW, M. K.; PFEIFER, L. M.; O'DONNELL, J. Response to a relational agent by hospital patients with depressive symptoms. *Interacting with computers*, Oxford University Press Oxford, UK, v. 22, n. 4, p. 289–298, 2010.
- BOBROW, D. G. A question-answering system for high school algebra word problems. p. 591–614, 1964.
- BRADEŠKO, L.; MLADENIĆ, D. A survey of chatbot systems through a loebner prize competition. p. 34–37, 2012.
- BRAHNAM, S.; ANGELI, A. D. Gender affordances of conversational agents. *Interacting with Computers*, Elsevier, v. 24, n. 3, p. 139–153, 2012.
- BRASIL, N. U. *OMS registra aumento de casos de depressão em todo o mundo; no Brasil são 11,5 milhões de pessoas*. 2017. <<https://nacoesunidas.org>>. Acesso em: 13 mar. 2017.
- BTB. *Beating the Blues*. [S.l.]: 365 HEALTH and WELLBEING, 2017. <<http://www.beatingtheblues.co.uk/>>. Acesso em: 20 abr. 2017.
- BURTON, C.; TATAR, A. S.; MCKINSTRY, B.; MATHESON, C.; MATU, S.; MOLDOVAN, R.; MACNAB, M.; FARROW, E.; DAVID, D.; PAGLIARI, C.; ANTONI, S. B.; WOLTERS, M. Pilot randomised controlled trial of help4mood, an embodied virtual agent-based system to support treatment of depression. *Journal of telemedicine and telecare*, SAGE Publications Sage UK: London, England, v. 22, n. 6, p. 348–355, 2016.
- BURTON, C.; WOLTERS, M. K.; BLANCO, A. S.; SZENTAGOTAI, A.; URE, J.; PAGLIARI, C.; MCKINSTRY, B. Help4mood: avatar-based support for treating people with major depression in the community. v. 6, p. 1–2, 2012.
- CAI, H.; WEI, S.; HAN, X.; XU, L.; SHA, X.; HU, B. Brief discussion on current computerized cognitive behavioral therapy. p. 420–431, 2015.
- CANBEK, N. G.; MUTLU, M. E. On the track of artificial intelligence: Learning with intelligent personal assistants. *Journal of Human Sciences*, v. 13, n. 1, p. 592–601, 2016.
- CHE, W.; LI, Z.; LIU, T. Ltp: A chinese language technology platform. p. 13–16, 2010.
- CHRISTENSEN, H.; GRIFFITHS, K. *MoodGYM Training Program*. [S.l.]: Australian National University, 2017. <<https://moodgym.anu.edu.au/welcome>>. Acesso em: 20 abr. 2017.
- CLARKE, A. M.; KUOSMANEN, T.; BARRY, M. M. A systematic review of online youth mental health promotion and prevention interventions. *Journal of youth and adolescence*, Springer, v. 44, n. 1, p. 90–113, 2015.
- CLARKE, G.; LEWINSOHN, P.; HOPS, H. *Leader's manual for adolescent groups: adolescent coping with depression course*. [S.l.]: Castalia, 1990.
- COFFMAN, S.; MCGLAMERY, S. The librarian and mr. jeeves. *American Libraries*, JSTOR, v. 31, n. 5, p. 66–69, 2000.

- COHEN, S.; KAMARCK, T.; MERMELSTEIN, R. A global measure of perceived stress. *Journal of health and social behavior*, JSTOR, p. 385–396, 1983.
- COLBY, K. M. *Artificial Paranoia: A Computer Simulation of Paranoid Processes*. New York, NY, USA: Elsevier Science Inc., 1975.
- COLBY, K. M. A computer program using cognitive therapy to treat depressed patients. *Psychiatric Services*, American Psychiatric Assn, 1995.
- COLBY, K. M. Human-computer conversation in a cognitive therapy program. Springer, p. 9–19, 1999.
- COLLINS-DONNELLY, K. *Banish Your Self-esteem Thief: A Cognitive Behavioural Therapy Workbook on Building Positive Self-esteem for Young People*. [S.l.]: Jessica Kingsley Publishers, 2014.
- DALE, R. The return of the chatbots. *Natural Language Engineering*, Cambridge University Press, v. 22, n. 5, p. 811–817, 2016.
- DERYUGINA, O. Chatterbots. *Scientific and Technical Information Processing*, Springer, v. 37, n. 2, p. 143–147, 2010.
- DEVAULT, D.; ARTSTEIN, R.; BENN, G.; DEY, T.; FAST, E.; GAINER, A.; GEORGILA, K.; GRATCH, J.; HARTHOLT, A.; LHOMMET, M.; GALE, L.; MARSELLA, S.; MORBINI, F.; NAZARIAN, A.; SCHERER, S.; STRATOU, G.; SURI, A.; TRAUM, D.; WOOD, R.; XU, Y.; RIZZO, A.; LOUIS-PHILIPPE, M. Simsensei kiosk: A virtual human interviewer for healthcare decision support. p. 1061–1068, 2014.
- EELLS, T. D.; BARRETT, M. S.; WRIGHT, J. H.; THASE, M. Computer-assisted cognitive-behavior therapy for depression. *Psychotherapy*, Educational Publishing Foundation, v. 51, n. 2, p. 191, 2014.
- EELLS, T. D.; BARRETT, M. S.; WRIGHT, J. H.; THASE, M. E. Can cognitive therapy be conducted by computers? *Current Behavioral Neuroscience Reports*, Springer, v. 2, n. 4, p. 209–215, 2015.
- ELLIS, A.; ELLIS, D. Rational emotive behavior therapy. *Editors-in-Chief*, 2002.
- FERRUCCI, D. A. Introduction to “this is watson”. *IBM Journal of Research and Development*, IBM, v. 56, n. 3.4, p. 1–15, 2012.
- FIOCRUZ, C. S. *Mais de 350 milhões de pessoas sofrem de depressão no mundo*. 2017. <<http://www.canal.fiocruz.br/destaque/index.php?id=722>>. Acesso em: 13 mar. 2017.
- FONER, L. N. Entertaining agents: A sociological case study. ACM, New York, NY, USA, p. 122–129, 1997. Disponível em: <<http://doi.acm.org/10.1145/267658.267684>>.
- FONSECA, A. A.; VIGÁRIO, M. Pistas duracionais e entoacionais na marcação de sintagmas entoacionais (i) no pe: a prosódia no efeito garden-path. *XXVI Encontro Nacional da Associação Portuguesa de Linguística. Textos Seleccionados*. Lisboa: APL, p. 266–281, 2011.
- FONSECA, E. R.; ROSA, J. L. G. Mac-morpho revisited: Towards robust part-of-speech tagging. p. 98–107, 2013.

- FONSECA, E. R.; ROSA, J. L. G. *nlpnet — Natural Language Processing with neural networks*. 2013. <<http://nilc.icmc.usp.br/nlpnet/intro.html>>. Acesso em: 12 abr. 2017.
- FOUNDATION, I. A. A. *AIML Implementations*. [S.l.]: ALICE AI Foundation, Inc., 2017. <<http://www.alicebot.org/downloads/programs.html>>. Acesso em: 24 abr. 2017.
- GARRIDO, P.; MARTINEZ, F. J.; GUETL, C. Adding semantic web knowledge to intelligent personal assistant agents. p. 1–12, 2010.
- GOOGLE, G. *Web Speech API limit of 60 seconds?* 2016. <<https://groups.google.com/a/chromium.org/forum/#!topic/chromium-html5/s2XhT-Y5qAc>>. Acesso em: 10 abr. 2017.
- HELGADÓTTIR, F. D.; MENZIES, R. G.; ONSLOW, M.; PACKMAN, A.; O'BRIAN, S. Online CBT I: Bridging the gap between eliza and modern online cbt treatment packages. *Behaviour Change*, Cambridge Univ Press, v. 26, n. 04, p. 245–253, 2009.
- HELP4MOOD. *Help4Mood*. [S.l.]: Help4Mood, 2017. <<http://help4mood.info/site/default.aspx>>. Acesso em: 17 abr. 2017.
- HILLMAN, K. *How everyone can use CBT to be happier and more effective*. 2014. <<http://www.psychology24.org/how-everyone-can-use-cbt-to-be-happier-and-more-effective>>. Acesso em: 13 mar. 2017.
- HOLMES, D.; MCCABE, M. C. Improving precision and recall for soundex retrieval. p. 22–26, 2002.
- HUANG, J.; LI, Q.; XUE, Y.; CHENG, T.; XU, S.; JIA, J.; FENG, L. Teenchat: a chatterbot system for sensing and releasing adolescents' stress. p. 133–145, 2015.
- JOINSON, A. N. Self-disclosure in computer-mediated communication: The role of self-awareness and visual anonymity. *European journal of social psychology*, Wiley Online Library, v. 31, n. 2, p. 177–192, 2001.
- JORGE, V. *Thesaurus da Língua Portuguesa do Brasil*. 2004. <1>. Acesso em: 01 mai. 2017.
- JR, B. F. G.; WOLF, A. K.; CHOMSKY, C.; LAUGHERY, K. Baseball: an automatic question-answerer. p. 219–224, 1961.
- KAR, S. *How chatbot Wysa helps one tackle depression*. [S.l.]: ET-tech, 2017. <<http://tech.economictimes.indiatimes.com/news/startups/how-chatbot-wysa-helps-one-tackle-depression/57284255>>. Acesso em: 19 abr. 2017.
- KHANNA, A.; PANDEY, B.; VASHISHTA, K.; KALIA, K.; PRADEEPKUMAR, B.; DAS, T. A study of today's ai through chatbots and rediscovery of machine intelligence. *Int. J. ue Serv. Sci. Technol*, v. 8, p. 277–284, 2015.
- KNAPP, P.; BECK, A. T. Fundamentos, modelos conceituais, aplicações e pesquisa da terapia cognitiva cognitive therapy: foundations, conceptual models, applications and research. *Rev Bras Psiquiatr*, v. 30, n. Supl II, p. 54–64, 2008.

- KOBAK, K. A.; MUNDT, J. C.; KENNARD, B. Integrating technology into cognitive behavior therapy for adolescent depression: a pilot study. *Annals of general psychiatry*, BioMed Central, v. 14, n. 1, p. 37–47, 2015.
- LEVENSHTAIN, V. I. Binary codes capable of correcting deletions, insertions, and reversals. v. 10, n. 8, p. 707–710, 1966.
- LEVINSON, S. C. Language and mind: Let's get the issues straight. *Language in mind: Advances in the study of language and thought*, p. 25–46, 2003.
- LIKERT, R. A technique for the measurement of attitudes. *Archives of psychology*, 1932.
- LIMA, M. S. d. Epidemiologia e impacto social. *Revista Brasileira de psiquiatria*, SciELO Brasil, v. 21, p. 01–05, 1999.
- LITTLE, W.; VYAIN, S.; SCARAMUZZO, G.; CODY-RYDZEWSKI, S.; GRIFFITHS, H.; STRAYER, E.; KEIRNS, N. *Chapter 2. Sociological Research*. [S.l.]: Houston Texas: OpenStax College, Rice University. Retrieved July, 2012. <<https://opentextbc.ca/introductiontosociology/chapter/chapter2-sociological-research/>>. Acesso em: 30 abr. 2017.
- LOEBNER, H. *Home Page of The Loebner Prize in Artificial Intelligence*. 2015. <<http://www.loebner.net/Prizef/loebner-prize.html>>. Acesso em: 01 mar. 2017.
- MADHU, D.; JAIN, C. N.; SEBASTAIN, E.; SHAJI, S.; AJAYAKUMAR, A. A novel approach for medical assistance using trained chatbot. p. 243–246, 2017.
- MALDONADO, M.; ALULEMA, D.; MOROCHO, D.; PROAÑO, M. System for monitoring natural disasters using natural language processing in the social network twitter. p. 1–6, 2016.
- MARCUSCHI, L. A. *Análise da conversação*. [S.l.]: Ed. Ática, 1986.
- MARTÍNEZ-MIRANDA, J. Towards an empathic virtual agent to support the treatment of major depression. p. 61–62, 2010.
- MAULDIN, M. L. Chatterbots, tinymuds, and the turing test: Entering the loebner prize competition. v. 94, p. 16–21, 1994.
- MCINTYRE, K. *Upset about Trump's win? This penguin chatbot helps combat depression*. [S.l.]: TECHINASIA, 2016. <<https://www.techinasia.com/ai-chatbot-wysa-touchkin-penguin>>. Acesso em: 19 abr. 2017.
- MCNEAL, M. L.; NEWYEAR, D. Chatbot creation options. *Library Technology Reports*, American Library Association, v. 49, n. 8, 2013.
- MCTEAR, M.; CALLEJAS, Z.; GRIOL, D. Conversational interfaces: Past and present. Springer, p. 51–72, 2016.
- MCTEAR, M.; CALLEJAS, Z.; GRIOL, D. Creating a conversational interface using chatbot technology. Springer, p. 125–159, 2016.
- MCTEAR, M.; CALLEJAS, Z.; GRIOL, D. Future directions. Springer, p. 403–418, 2016.

- MCTEAR, M. F. Spoken dialogue applications: Research directions and commercial deployment. Springer, p. 19–43, 2004.
- MÜLLER, D. N. Processamento de linguagem natural. *Porto Alegre*, 2003.
- NARDI, B.; MASSEI, M.; ARIMATEA, E.; MOLTEDO-PERFETTI, A. Effectiveness of group cbt in treating adolescents with depression symptoms: a critical review. *International journal of adolescent medicine and health*, 2016.
- NEENAN, M.; DRYDEN, W. *Cognitive behaviour therapy: 100 key points and techniques*. [S.l.]: Routledge, 2014.
- NETO, A. C. D. Introdução a teste de software. *Engenharia de Software Magazine*, v. 1, 2007.
- NEVES, A. M. M. *Iaiml: Um mecanismo para o tratamento de intenção em chatterbots*. Tese (Tese de Doutorado em Ciência da Computação) — Universidade Federal de Pernambuco, 2005.
- NEVES, A. M. M.; BARROS, F. A.; HODGES, C. Iaiml: A mechanism to treat intentionality in aiml chatterbots. p. 225–231, 2006.
- NICE. *Computerised cognitive behaviour therapy for depression and anxiety*. 2013. <<https://www.nice.org.uk/guidance/ta97/chapter/3-The-technology>>. Acesso em: 13 mar. 2017.
- NICE. *Depression in adults: recognition and management*. 2016. <<https://www.nice.org.uk/guidance/cg90/ifp/chapter/treatments-for-mild-to-moderate-depression#further-treatment-for-mild-to-moderate-depression>>. Acesso em: 17 mar. 2017.
- NÉMETH, L. *Hunspell*. 2017. <<http://hunspell.github.io/>>. Acesso em: 25 mar. 2017.
- OLIVEIRA, F. A.; NAVAUX, P. O. A. Processamento de linguagem natural: princípios básicos e a implementação de um analisador sintático de sentenças da língua portuguesa. *Rio Grande do Sul*, 2004.
- ORENGO, V. M.; HUYCK, C. R. *RSLP Stemmer (Removedor de Sufixos da Língua Portuguesa)*. 2001. <<http://www.inf.ufrgs.br/~viviane/rsrp/>>. Acesso em: 24 mar. 2017.
- ORENGO, V. M.; HUYCK, C. R. A stemming algorithm for the portuguese language. v. 8, p. 186–193, 2001.
- O'SHEA, K.; BANDAR, Z.; CROCKETT, K. A conversational agent framework using semantic analysis. *International Journal of Intelligent Computing Research (IJICR)*, v. 1, n. 1/2, 2010.
- PAIVA, V. L. M. O. A linguagem dos emojis. *Trabalhos em Linguística Aplicada*, v. 55, n. 2, p. 379–399, 2016.
- PARKINSON, R. C.; COLBY, K. M.; FAUGHT, W. S. Conversational language comprehension using integrated pattern-matching and parsing. *Artificial Intelligence*, Elsevier, v. 9, n. 2, p. 111–134, 1977.

- PATIL, A.; MARIMUTHU, K.; NIRANCHANA, R. et al. Comparative study of cloud platforms to develop a chatbot. *International Journal of Engineering & Technology*, v. 6, n. 3, p. 57–61, 2017.
- PEPPING, C. A.; O'DONOVAN, A.; DAVIS, P. J. The positive effects of mindfulness on self-esteem. *The Journal of Positive Psychology*, Taylor & Francis, v. 8, n. 5, p. 376–386, 2013.
- PEREIRA, M. J.; COHEUR, L.; FIALHO, P.; RIBEIRO, R. Chatbots' greetings to human-computer communication. *arXiv preprint arXiv:1609.06479*, 2016.
- PERKINS, B. *Most common chatbot questions and how to answer them*. 2016. <<https://blogs.msdn.microsoft.com/benjaminperkins/2016/11/21/most-common-chatbot-questions-and-how-to-answer-them/>>. Acesso em: 13 abr. 2017.
- PFLEEGER, S. L.; KITCHENHAM, B. A. Principles of survey research: part 1: turning lemons into lemonade. *ACM SIGSOFT Software Engineering Notes*, ACM, v. 26, n. 6, p. 16–18, 2001.
- PHP.NET. *Crypt*. 2017. <https://secure.php.net/manual/pt_BR/function.crypt.php>. Acesso em: 10 mai. 2017.
- PRIBERAM, I. S. *Priberam Dicionário*. 2017. <<https://www.priberam.pt/dlpo/Default.aspx>>. Acesso em: 29 mar. 2017.
- PRIZE, L. *Alice connected to Judges*. 2004. <http://loebner.net/Prize/2004_Contest/Wallace.html>. Acesso em: 23 abr. 2017.
- PRIZE, L. *Transcript of Loebner Prize 2015 finalist selection conversation with Rose*. 2015. <<http://www.aisb.org.uk/media/files/LoebnerPrize2015/Rose.pdf>>. Acesso em: 23 abr. 2017.
- PSICOLOGIA, C. F. *Serviços Psicológicos pela internet*. 2012. <<http://site.cfp.org.br/servicos-psicologicos-on-line/>>. Acesso em: 29 mar. 2017.
- PSIQUIATRIA, A. B. Suicídio: informando para prevenir. *Comissão de Estudos e Prevenção de Suicídio. Brasília CFM/ABP*, p. 52, 2014.
- RABELO, J.; BARROS, F. Pergunte! uma interface em português para pergunta-resposta na web. *Master's thesis, Informatics Center, Federal University of Pernambuco, Brazil*, 2004.
- RECUERO, R. C. Elementos para a análise da conversação na comunicação mediada pelo computador. *Verso e Reverso*, v. 22, n. 51, 2008.
- REPÚBLICA, P. *Estatuto da Criança e do Adolescente*. 1990. <http://www.planalto.gov.br/ccivil_03/leis/L8069.htm>. Acesso em: 29 mar. 2017.
- REPÚBLICA, P. *Estatuto da Juventude*. 2013. <http://www.planalto.gov.br/ccivil_03/_Ato2011-2014/2013/Lei/L12852.htm>. Acesso em: 02 mai. 2017.

- RICHARDSON, T.; STALLARD, P.; VELLEMAN, S. Computerised cognitive behavioural therapy for the prevention and treatment of depression and anxiety in children and adolescents: a systematic review. *Clinical child and family psychology review*, Springer, v. 13, n. 3, p. 275–290, 2010.
- RINGATE, T.; WALLACE, R. S.; TAYLOR, A.; BAER, J.; DANIELS, D. *alicebot.org*. 2011. <<http://www.alicebot.org/documentation/aiml-primer.html>>. Acesso em: 10 fev. 2017.
- RIZZO, S. A.; MORENCY, L. P. *SimSensei*. [S.l.]: University of Southern California Institute for Creative Technologies, 2011. <<http://ict.usc.edu/prototypes/simsensei/>>. Acesso em: 19 abr. 2017.
- ROBINSON, A. *Meet Ellie, the machine that can detect depression*. [S.l.]: The Guardian, 2015. <<https://www.theguardian.com/sustainable-business/2015/sep/17/ellie-machine-that-can-detect-depression>>. Acesso em: 19 abr. 2017.
- ROGERS, C. R. *Client-centered therapy*. [S.l.]: American Psychological Association, 1966.
- ROSETTACODE.ORG. *Levenshtein distance*. 2017. <http://rosettacode.org/wiki/Levenshtein_distance#Iterative>. Acesso em: 11 abr. 2017.
- RUSSELL, S.; NORVIG, P. *Inteligência Artificial, Tradução da 3ª Edição*. [S.l.]: Elsevier Editora Ltda, 2013. v. 1.
- SACKS, H.; SCHEGLOFF, E. A.; JEFFERSON, G. A simplest systematics for the organization of turn-taking for conversation. *language*, JSTOR, p. 696–735, 1974.
- SARIKAYA, R. The technology behind personal digital assistants: An overview of the system architecture and key components. *IEEE Signal Processing Magazine*, IEEE, v. 34, n. 1, p. 67–81, 2017.
- SCHUELLER, S. M.; STILES-SHIELDS, C.; YAROSH, L. Online treatment and virtual therapists in child and adolescent psychiatry. *Child and Adolescent Psychiatric Clinics of North America*, Elsevier, v. 26, n. 1, p. 1–12, 2017.
- SCHUETZLER, R. M. Dynamic interviewing agents: Effects on deception, nonverbal behavior, and social desirability. The University of Arizona., 2015.
- SHAPIRO, F.; SOLOMON, R. M. *Eye movement desensitization and reprocessing*. [S.l.]: Wiley Online Library, 1995.
- SHAWAR, B. A.; ATWELL, E. Different measurements metrics to evaluate a chatbot system. p. 89–96, 2007.
- SHIRES, G.; WENNBORG, H. *Web Speech API Specification*. 2012. <<https://dvcs.w3.org/hg/speech-api/raw-file/tip/speechapi.html>>. Acesso em: 23 mar. 2017.
- SILVA, A.; COSTA, E. B. Um chatterbot aplicado ao turismo: um estudo de caso no litoral alagoano e na cidade de maceió. *Escola Regional de Informática, ERI-PR*, v. 14, 2007.

- SILVA, R. N.; KOTUJANSKY, A. L. *Serviços Psicológicos Mediados por Computador*. 2017. <http://www.ufrgs.br/e-psico/etica/temas_atuais/servicos-psicologicos-texto.html>. Acesso em: 31 mar. 2017.
- SPIZZIRRI, R. C. P.; WAGNER, A.; MOSMANN, C. P.; ARMANI, A. B. Adolescência conectada: Mapeando o uso da internet em jovens internautas. *Psicol. argum*, v. 30, n. 69, p. 327–335, 2012.
- STALLARD, P. *Bons pensamentos-Bons sentimentos*. [S.l.]: Artmed Editora, 2007.
- STASIAK, K.; FLEMING, T.; LUCASSEN, M. F.; SHEPHERD, M. J.; WHITTAKER, R.; MERRY, S. N. Computer-based and online therapy for depression and anxiety in children and adolescents. *Journal of child and adolescent psychopharmacology*, Mary Ann Liebert, Inc. 140 Huguenot Street, 3rd Floor New Rochelle, NY 10801 USA, v. 26, n. 3, p. 235–245, 2016.
- STRATOU, G.; MORENCY, L.-P. Multisense—context-aware nonverbal behavior analysis framework: A psychological distress use case. *IEEE Transactions on Affective Computing*, IEEE, v. 8, n. 2, p. 190–203, 2017.
- SUDHIR, P. M. Cognitive behaviour therapy with adolescents. Springer, p. 21–42, 2015.
- TOUCHKIN. *Wysa*. [S.l.]: Touchkin, 2016. <<http://www.wysa.io/>>. Acesso em: 19 abr. 2017.
- TURING, A. M. Computing machinery and intelligence. *Mind*, JSTOR, v. 59, n. 236, p. 433–460, 1950.
- UNIVERSITY, A. N. *MoodGym*. 2001. <<https://moodgym.anu.edu.au/welcome>>. Acesso em: 13 mar. 2017.
- VIEIRA, R.; LIMA, V. L. *Linguística computacional: princípios e aplicações*. v. 3, p. 47–86, 2001.
- WALLACE, R. S. The anatomy of alicia. Springer, p. 181–210, 2009.
- WALLACE, R. S. *alicebot.org*. 2014. <<https://docs.google.com/document/d/1wNT25hJRyupcG51aO89UcQEiG-HkXRXusukADpFnDs4/pub>>. Acesso em: 26 fev. 2017.
- WEERATUNGA, A.; JAYAWARDANA, S.; HASINDU, P.; PRASHAN, W.; THELIJJAGODA, S. Project nethra—an intelligent assistant for the visually disabled to interact with internet services. p. 55–59, 2015.
- WEINTRAUB, J. *History of the PC THERAPIST*. [S.l.]: Loebner Prize, 2017. <<http://loebner.net/Prizet/weintraub-bio.html>>. Acesso em: 24 abr. 2017.
- WEISSMAN, M. M.; MARKOWITZ, J. C. Interpersonal psychotherapy: Current status. *Archives of General Psychiatry*, American Medical Association, v. 51, n. 8, p. 599–606, 1994.
- WEIZENBAUM, J. Eliza—a computer program for the study of natural language communication between man and machine. *Communications of the ACM*, ACM, v. 9, n. 1, p. 36–45, 1966.

- WERLANG, B. S. G.; BORGES, V. R.; FENSTERSEIFER, L. Índios de potencial suicida na adolescência. *Psicologia Revista. Revista da Faculdade de Ciências Humanas e da Saúde. ISSN 1413-4063*, v. 14, n. 1, p. 41–57, 2014.
- WILCOX, B. Beyond façade: Pattern matching for natural language applications. *GamaSutra. com*, 2011.
- WILCOX, B. *ChatScript Basic User Manual*. 2016. <<https://sourceforge.net/projects/chatscript/files/>>. Acesso em: 09 ago. 2016.
- WILCOX, B. *Brillig Understanding, Inc.* 2017. <<http://brilligunderstanding.com/>>. Acesso em: 31 mar. 2017.
- WILCOX, B. *ChatScript Engine*. 2017. <<https://sourceforge.net/projects/chatscript/>>. Acesso em: 06 mar. 2017.
- WINOGRAD, T. Understanding natural language. *Cognitive psychology*, Elsevier, v. 3, n. 1, p. 1–191, 1972.
- WOLTERS, M. K.; BURTON, C. D.; MATHESON, C.; BRESÓ, A.; SZENTAGOTAI, A.; MARTINEZ-MIRANDA, J.; ROSELL, J.; FUSTER, E.; PAGLIARI, C.; MCKINSTRY, B. Help4mood—supporting joint sense making in the treatment of major depressive disorder. 2013.
- WOLTERS, M. K.; FERRINI, L.; FARROW, E.; TATAR, A. S.; BURTON, C. D. Tracking depressed mood using speech pause patterns. 2015.
- WOLTERS, M. K.; FERRINI, L.; MARTÍNEZ-MIRANDA, J.; HASTIE, H.; BURTON, C. *Help4Mood-A Flexible Solution for Supporting People with Depression in the Community across Europe*. [S.l.]: MedeTel, 2012. <http://www.macs.hw.ac.uk/~hh117/pubs/Waltersetal_Medetel2012.pdf>. Acesso em: 17 abr. 2017.
- WORSWICK, S. *Chatbot Battles*. [S.l.]: Chatbot Battles, 2012. <<http://www.chatbotbattles.com/>>. Acesso em: 24 abr. 2017.
- WRIGHT, J.; WRIGHT, A.; BECK, A. *Good Days Ahead*. [S.l.]: Empower Interactive, 2017. <<https://www.gooddaysahead.com/>>. Acesso em: 20 abr. 2017.
- YAN, M.; CASTRO, P.; CHENG, P.; ISHAKIAN, V. Building a chatbot with serverless computing. ACM, New York, NY, USA, p. 5:1–5:4, 2016.
- YELLOWLEES, P. M.; HILTY, D. M.; MUCIC, D. Global/worldwide e-mental health: International and futuristic perspectives of telepsychiatry and the future. Springer, p. 233–249, 2016.
- YOU, J. Beyond the turing test. *Science*, American Association for the Advancement of Science, v. 347, n. 6218, p. 116–116, 2015.
- ZUGMAN, S. *Erros cognitivos comuns*. 2011. <www.ictc.com.br/arquivos/erros_cognitivos_comuns.pdf>. Acesso em: 11 mar. 2017.

APÊNDICE A – SCRIPT DE DIÁLOGO

Script de Diálogo Modelado para Beck (versão 5 data: 10-05-17)

Abertura

1 **Beck:** Seja bem vindo [usuário!]

Apresentação e Aquisição de perfil do usuário

1 **Beck:** Como este é o nosso primeiro contato, eu gostaria de me apresentar pra você.

2 **Beck:** Meu nome é Beck.

3 **Beck:** Eu sou um robô baseado em Inteligência Artificial.

4 **Beck:** Moro em um computador da Internet e nas horas vagas eu gosto de conversar com as pessoas.

5 **Beck:** Eu fui criada para tentar ajudar as pessoas a enfrentar e vencer algumas dificuldades da vida.

6 **Beck:** Agora eu gostaria de conversar com você para [gênero] conhecer melhor.

7 **Beck:** Vou fazer algumas perguntas para você. Tudo bem?

Usuário:

7.1) ~afirmação

7.2) ~negação

7.3) ~dúvida

7.4) *

Beck:

7.1) Vamos lá então.

7.2) 7.3) 7.4) Veja, para interagirmos melhor, eu preciso saber algumas coisas sobre você.

7.2) 7.3) 7.4) Não se preocupe, toda nossa conversa é confidencial e protegida. Ela não será divulgada a ninguém.

8 **Beck:** Com quem você mora?

Usuário:

8.1) ~família

8.2) ninguém ou sozinho

8.3) amigo

8.4) Não quero responder

8.5) *

Beck:

8.1) Certo, você mora com [família] (se não tiver mãe e/ou pai pergunta: “Onde está seu pai e/ou sua mãe?”)

Usuário:

8.1.1) morreu

8.1.2) se separaram.

- 8.1.3) não sei.
- 8.1.4) nunca conheci.
- 8.1.5) *

Beck:

- 8.1.1) sinto muito.
- 8.1.2) 8.1.3) 8.1.4) 8.1.5) Entendi.

Beck:

- 8.2) 8.3) Onde estão seus pais?

Usuário:

- 8.2.1) morreu
- 8.2.2) se separaram.
- 8.2.3) não sei.
- 8.2.4) nunca conheci.
- 8.2.5) *

Beck:

- 8.2.1) sinto muito.
- 8.2.2) 8.2.3) 8.2.4) 8.2.5) Entendi.

Beck:

- 8.4) Veja, [usuario], eu preciso que você converse comigo para que eu possa tentar [gênero] ajudar. Se você não estiver disposto a responder algumas perguntas simples, será difícil para mim continuar nossa conversa neste momento. Lembre-se de que eu sou apenas um robô.
- 8.4) E então, podemos continuar? Você concorda em responder as minhas perguntas?

Usuário:

- 8.4.1) ~afirmação
- 8.4.2) ~negação
- 8.4.3) ~dúvida
- 8.4.4) *

Beck:

- 8.4.1) Está certo, então vamos continuar a nossa conversa. Reuse o item 8)
- 8.4.2) Então não tem muita coisa que eu possa fazer no momento. Vou parar a conversa agora e aguardar seu próximo contato.
- 8.4.2) Quando você quiser voltar a conversar é só entrar na minha página novamente.
- 8.4.2) Até mais, [usuario]!
- 8.4.3) Veja, são perguntas muito simples, e lembre-se de que ninguém saberá o que conversarmos aqui. Vou fazer novamente as perguntas. Reuse o item 8)
- 8.4.4) Não entendi o que você disse. Tenho algumas limitações de vocabulário na minha cabeça digital. Mas tudo bem. Vou fazer as perguntas novamente. Reuse o item 8)

8.5) Não entendi com quem você mora. Desculpe, mas eu sou apenas um robô, não tenho conhecimento de algumas palavras na minha cabeça digital. Você pode dizer com outras palavras (perguntar só mais uma vez pra não ficar no loop)?

9 **Beck:** Eu gostaria de saber agora o que você faz pra se divertir?

10 **Beck:** Veja, vou lhe mostrar do lado direito da tela um quadro, e gostaria que você escolhesse uma ou mais opções.

Quadro: Atividades de diversão			
Esportes (futebol, vôlei, natação, ...)	Ouvir música	Ir ao cinema ou teatro	Usar drogas
Conversar com pessoas	Navegar na Internet	Ir à festas	Beber
Desenhar ou pintar	Jogar videogame	Fazer compras	Namorar
Dançar ou cantar	Passear com amigos	Comer	Nenhuma atividade
Ler	Assistir TV	Dormir	Outros. Qual?

11 **Beck:** Pronto, o quadro já está na tela. Quando terminar de escolher, clique no botão “Guardar”.

Usuário:

- 11.1) Uma atividade
- 11.2) Varias atividades
- 11.3) Outras atividades
- 11.4) Nenhuma atividade

Beck:

- 11.1) 11.3) Entendi que você gosta de [atividade].
- 11.2) Você gosta de várias coisas.
- 11.4) Você não gosta de fazer nada?

Usuário:

- 11.4.1) ~afirmação/~negação

Beck:

- 11.4.1) Tudo bem. Depois falamos mais sobre isso.

12 **Beck:** Você se incomoda em me dizer qual é a sua orientação sexual? Se não ficar à vontade para falar sobre isso, não precisa responder.

Usuário:

- 12.1) ~orientação_sexual
- 12.2) ~negação
- 12.3) ~afirmação
- 12.4) o que é orientação sexual
- 12.5) *

Beck:

- 12.1) Entendi. obrigada por responder.

12.2) Então, qual é a sua orientação sexual?

Usuário:

12.2.1) ~orientacao_sexual

12.2.2) *

Beck:

12.2.1) Entendi. obrigada por responder.

12.2.2) Desculpe, eu não entendi a sua orientação sexual. Você poderia me dizer mais uma vez usando outras palavra? Reuse item 12.2)

12.3) Tudo bem. Vamos continuar a conversa.

12.4) É a preferência, desejo ou atração sexual, que pode ser pelo sexo oposto da pessoa, do mesmo ou de ambos.

12.5) Desculpe, eu não entendi. Reuse o item 11) só mais uma vez.

13 Beck: Você tem [namorado] (adequar a opção sexual do adolescente, caso ele tenha respondido) ?

Usuário:

13.1) ~afirmacao

13.2) ~negacao

13.3) ~duvida

13.4) *

Beck:

13.1) 13.2) Entendi.

13.3) Tudo bem. Entendi que você está em dúvida.

13.4) Não entendi o que você falou. Você pode dizer novamente com outras palavras?

(Reuse o item 13 só apenas uma vez pra não entrar no loop)

13 Beck: Agora eu já [gênero] conheço um pouco e fico feliz de estarmos construindo um vínculo.

Desenvolvimento

Falando da nossa Personalidade

1 Beck: Antes de continuarmos a conversa, eu vou passar pra você algumas informações que podem [gênero] ajudar a entender melhor como as pessoas funcionam.

2 Beck: Você sabia que a personalidade das pessoas pode influenciar os seus pensamentos, sentimentos e comportamentos?!

3 Beck: Você sabia que os pensamentos que as pessoas têm podem afetar os seus sentimentos e comportamentos?

4 Beck: Por exemplo: Quando temos pensamentos pessimistas, geralmente nos sentimos tristes.

5 Beck: Veja que os sentimentos também podem afetar os seus pensamentos e comportamentos.

6 Beck: Por exemplo: Algumas pessoas quando estão tristes, adotam o comportamento de se isolar das outras pessoas.

7 **Beck:** É como um ciclo de três elementos. Cada elemento (pensamento, sentimento e comportamento) pode afetar os outros dois.

8 **Beck:** Veja a figura no quadro ao lado. Ela ilustra este ciclo.

9 **Beck:** Então, vamos cuidar dos nossos pensamentos para que isso não nos atrapalhe, certo?!

10 **Beck:** No quadro ao lado, por favor, escolha. Pra você, qual desses elementos é mais fácil de ser controlado?

Quadro: Qual o elemento você acha mais fácil de controlar?	
Pensamentos	Sentimentos
Comportamentos	Não sei

Usuário:

10.1) Pensamentos

10.2) Sentimentos

10.3) Comportamentos

10.4) Não sei.

Beck:

10.1) Isso mesmo. O pensamento é um dos elementos mais fáceis de controlar. O outro elemento também fácil de controlar é o comportamento. Muitas pessoas se frustram porque querem controlar logo seus sentimentos, e isso é mais difícil.

10.2) Veja, o sentimento é o elemento menos fácil de controlar. Muitas pessoas se frustram porque querem controlar logo seus sentimentos. O pensamento e o comportamento são os elementos mais fáceis de controlar.

10.3) Isso mesmo. O comportamento é um dos elementos mais fáceis de controlar. O outro elemento fácil de controlar é o pensamento. Muitas pessoas se frustram porque querem controlar logo seus sentimentos, e isso é mais difícil.

10.4) O pensamento e o comportamento são os elementos mais fáceis de controlar. Muitas pessoas se frustram porque querem controlar logo seus sentimentos, e isso é mais difícil.

Humor do adolescente

1 **Beck:** Vamos prosseguir a conversa.

2 **Beck:** Eu gostaria de saber agora como você está se sentindo.

3 **Beck:** Vou lhe mostrar o quadro do Humor no lado direito da tela e gostaria que você escolhesse uma das opções.

Como você está se sentindo agora?				
				

4 **Beck:** Para saber o que significa cada figura é só apontar a seta do mouse sobre ela.

5 **Beck:** Pense um pouco antes de responder, e quando terminar é só clicar no estado de humor que melhor reflete o que você sente agora.

6 **Beck:** Em seguida é só clicar em “Guardar”.

Usuário:

- 6.1) Muito triste
- 6.2) Triste
- 6.3) Indiferente
- 6.4) Feliz
- 6.5) Muito feliz

Beck:

- 6.1) 6.2) (somente na primeira vez que respondeu). Você tem se sentido triste há muito tempo?

Usuário:

- 6.1.1) 6.2.1) ~afirmação ~negação

Beck:

- 6.1.1) 6.2.1) Você já conversou com seus pais (se não tem pais, com algum adulto de sua confiança) sobre o que você está sentindo?

Usuário:

- 6.1.1.1) 6.2.1.1) ~afirmação
- 6.1.1.2) 6.2.1.2) ~negação

Beck:

- 6.1.1.1) 6.2.1.1) ~afirmação

- 6.1) 6.2) 6.3) 6.4) 6.5) obrigada por ter respondido. O quadro do Humor é muito importante.

7 Beck: Sempre que conversarmos, vou lhe pedir para respondê-lo. Tudo bem?

Usuário:

- 7.1) ~afirmação
- 7.2) ~negação

Beck:

- 7.1) Certo. Desta forma poderemos acompanhar como você está se sentindo.
- 7.2) Tente reconsiderar sua decisão, pois essa informação é muito importante para mim. Lembre-se de que eu sou um robô... não tenho olhos para ver sua expressão, seu humor...
- Da próxima vez em que você responder, tenho certeza de que ainda será mais rápido.

Explorando sentimentos

1 Beck: Vamos explorar alguns sentimentos...

2 Beck: Você sabia que, quando nos sentimos bem, temos mais vontade de criar e descobrir coisas, conhecer e nos relacionarmos com as pessoas?

3 Beck: E o contrário também é verdadeiro.

4 Beck: Quando nos sentimos mal, temos menos probabilidade de fazer as coisas que gostamos.

5 Beck: E então começamos a duvidar da nossa capacidade. Como por exemplo, da nossa capacidade de fazer amigos.

6 Beck: Por falar em amigos... Você tem amigos?

Usuário:

6.1) ~afirmação

6.2) ~negação

6.3) tenho número_amigos

6.4) tenho nome_amigo

6.5) ~dúvida

Beck:

6.1) São amigos reais ou virtuais?

Usuário:

6.1.1) reais

6.1.2) virtuais

6.1.3) reais e virtuais (os dois, ambos, tanto um quanto o outro, um pouco de cada)

6.1.4) *

Beck:

6.1.1) 6.1.2) 6.1.3) Legal. É muito bom ter amigos.

6.1.4) Não entendi. Você pode responder novamente com outras palavras? (Pergunta só mais uma vez).

6.2) Entendi. Depois vou passar algumas dicas pra fazer novas amizades, ok?

Usuário:

6.2.1) ~afirmação

6.2.2) ~negação

Beck:

6.2.1) Ok, depois falamos mais sobre amigos.

6.2.2) Tudo bem.

6.3) Entendi. Que bom que você tem (número_amigos) amigos. (reusa item 6.1)

6.4) Legal. Esse(s) (amigo) é/são os seus melhores amigos?

Usuário:

6.4.1) ~afirmação

6.4.2) ~negação

6.4.3) ~negação e nomes dos amigos.

Beck:

6.4.1) Entendi. É muito bom ter amigos.

6.4.2) Qual é ou quais são os nomes dos seus melhores amigos?

Usuário:

6.4.2.1) fulano, sicrano e beltrano.

6.4.2.2) *

Beck:

6.4.2.1) entendi, seus amigos são: fulano, sicrano e beltrano. É muito bom ter amigos.

6.4.2.2) não entendi. (pergunta só mais uma vez).

6.4.3) Entendi, é fulano, sicrano e beltrano ...

6.5) Entendi, você está na dúvida, não é?! Depois vou passar algumas dicas pra você de como fazer novos amigos, ok?

Usuário:

6.5.1) ~afirmação

6.5.2) ~negação

Beck:

6.5.1) 6.5.2) Tudo bem. Depois falamos mais sobre amigos.

7 Beck: Vou mudar um pouco de assunto agora, vamos falar novamente de atividades para se divertir.

Estimulando Atividades

1 Beck: (se atividade) Você me falou que gosta de fazer essa atividade. (atividade lícita)

2 Beck: (se mais de uma atividade) Essas são algumas das atividades que você disse que gostava de fazer (atividades lícitas).

3 Beck: (se mais de uma atividade) Qual dessas atividades você tem feito mais frequentemente ?

Usuário:

3.1) atividade(s)

3.2) nenhuma

3.3) *

Beck:

3.1) Reuse item 4).

3.2) Reuse o item 4.1.2).

3.3) *

4 Beck: Qual a frequência com que você tem feito esta atividade durante a semana?

Usuário:

4.1) (frequência = 0)

4.2) (frequência > 0)

4.3) *

Beck:

4.1) (se frequência = 0) Vamos tentar fazê-la pelo menos uma vez nesta semana?

Usuário:

4.1.1) ~afirmação

4.1.2) ~negação

Beck:

4.1.1) Fico feliz que você topou.

4.1.1) Vamos fazer um compromisso então?

4.1.1) O que você acha de eu lhe perguntar daqui a uns 2 dias se você conseguiu fazê-la. Tudo bem?

Usuário:

4.1.1.1) ~afirmação e novo prazo

4.1.1.2) ~afirmação

4.1.1.3) ~negação e novo prazo

4.1.1.4) ~negação

Beck:

4.1.1.1) e 4.1.1.3) (Se prazo > 5 dias) Hummm, eu acho muito tempo. Tenho certeza de que você consegue fazê-la em menos tempo. O que você acha de um prazo de 3 dias?

Usuário:

4.1.1.1.1) ~afirmação e novo prazo

4.1.1.1.2) ~afirmação

4.1.1.1.3) ~negação e novo prazo

4.1.1.1.4) ~negação

Beck:

4.1.1.1.1) e 4.1.1.1.3) (Se prazo ≤ 5 dias) Tudo bem então. Depois eu pergunto como foi.

4.1.1.1.1) e 4.1.1.1.3) (Se prazo > 5 dias e ≤ 10) É muito tempo, mas se você prefere assim, tudo bem. Depois eu pergunto como foi.

4.1.1.1.1) e 4.1.1.1.3) (Se prazo > 10 dias) Parece que você não está com vontade de fazer essa atividade agora. Tudo bem. Mas veja, uma maneira de mudar a forma de como nos sentimos é mudar a forma de como nos comportamos.

4.1.1.1.2) Tudo bem então. Depois eu pergunto como foi.

4.1.1.1.4) Qual o prazo que você prefere então?

Usuário:

4.1.1.1.4.1) prazo

4.1.1.1.4.1) negação

Beck:

4.1.1.1.4.1) (Se prazo ≤ 5 dias) Tudo bem então. Depois eu pergunto como foi.

4.1.1.1.4.1) (Se prazo > 5 dias e ≤ 10) É muito tempo, mas se você prefere assim, tudo bem.

- 4.1.1.1.4.1) (Se prazo > 10 dias) Acho que você não está com vontade de fazer essa atividade agora. Tudo bem. Mas veja, uma maneira de mudar a forma de como nos sentimos é mudar a forma de como nos comportamos.
- 4.1.1.1.4.2) Tudo bem então. Depois falamos sobre isso novamente. Mas veja, uma maneira de mudar a forma de como nos sentimos é mudar a forma de como nos comportamos.

- 4.1.1.1) e 4.1.1.3) (Se prazo <= 5 dias) Tudo bem então. Depois eu pergunto como foi.
- 4.1.1.2) Então está combinado. Depois eu pergunto como foi.
- 4.1.1.4) Que tal 3 dias?

Usuário:

- 4.1.1.4.1) ~afirmação e novo prazo
- 4.1.1.4.2) ~afirmação
- 4.1.1.4.3) ~negação e novo prazo
- 4.1.1.4.4) ~negação

Beck:

- 4.1.1.4.1) 4.1.1.4.3) (Se prazo <= 5 dias) Tudo bem então. Depois eu pergunto como foi.
- 4.1.1.4.1) 4.1.1.4.3) (Se prazo > 5 dias e <= 10) É muito tempo, mas se você prefere assim, tudo bem.
- 4.1.1.4.1) 4.1.1.4.3) (Se prazo > 10 dias) Acho que você não está com vontade de fazer essa atividade agora. Tudo bem. Mas veja, uma maneira de mudar a forma de como nos sentimos é mudar a forma de como nos comportamos.
- 4.1.1.4.2) Então está combinado. Depois eu pergunto como foi.
- 4.1.1.4.4) Tudo bem então. Mas veja, uma maneira de mudar a forma de como nos sentimos é mudar a forma de como nos comportamos.

- 4.1.2) Veja, pode ser algo pequeno, o importante é que você goste e consiga fazer essa atividade.
- 4.1.2) Vamos pensar em alguma outra coisa pra fazer então?
- 4.1.2) (se tem amigo) Que tal ligar para alguém de quem você goste e conversar um pouco?
- 4.1.2) (se não tem amigo) Que tal jogar um jogo que você gosta muito?

Usuário:

4.1.2.1) ~afirmação

4.1.2.2) ~negação e atividade

4.1.2.3) ~negação

Beck:

4.1.2.1) Reuse item (4.1.1)

4.1.2.2) Então você quer fazer [atividade], está certo?

Usuário:

4.1.2.2.1) ~afirmação

4.1.2.2.2) ~negação e atividade

4.1.2.2.3) ~negação

Beck:

4.1.2.2.1) Reuse item (4.1.1)

4.1.2.2.2) Veja se entendi correto agora: Você quer fazer [atividade], certo?

Usuário:

4.1.2.2.2.1) ~afirmação

4.1.2.2.2.2) ~negação e atividade

4.1.2.2.2.3) ~negação

Beck:

4.1.2.2.2.1) Reuse item (4.1.1)

4.1.2.2.2.2) 4.1.2.2.2.3) Desculpe não ter entendido! Voltaremos a falar sobre esse assunto em uma outra ocasião.

4.1.2.2.3) Você pode repetir mais uma vez o que gostaria de fazer, por favor? É que eu não entendi direito. Eu tenho algumas limitações no meu vocabulário.

Usuário:

4.1.2.2.3.1) ~afirmação e atividade

4.1.2.2.3.2) ~afirmação

4.1.2.2.3.3) ~negação

Beck:

4.1.2.2.3.1) Veja se entendi correto agora: Você quer fazer [atividade], certo?

Usuário:

4.1.2.2.3.1.1) ~afirmação

4.1.2.2.3.1.2) ~negação e atividade

4.1.2.2.3.1.3) ~negação

Beck:

4.1.2.2.3.1.1) Reuse item (4.1.1)

4.1.2.2.3.1.2) 4.1.2.2.3.1.3) Desculpe não ter entendido! Voltaremos a falar sobre esse assunto em uma outra ocasião.

4.1.2.2.3.2) Qual atividade que você gostaria de fazer para que você possa se sentir mais feliz?

Usuário:

4.1.2.2.3.2.1) atividade

4.1.2.2.3.2.2) ~negacao

Beck:

4.1.2.2.3.2.1) Reuse item (4.1.2.2.3.1)

4.1.2.2.3.2.2) Tudo bem então. Depois falamos sobre esse assunto novamente. Mas veja, uma maneira de mudar a forma de como nos sentimos é mudar a forma de como nos comportamos.

4.1.2.2.3.3) Desculpe não ter entendido!

Voltaremos a falar sobre esse assunto em uma outra ocasião.

4.1.2.3) Tudo bem então. Depois falamos sobre esse assunto novamente. Mas veja, uma maneira de mudar a forma de como nos sentimos é mudar a forma de como nos comportamos.

4.2) (frequência > 0) Que bom que você está fazendo algo de que gosta. Isso é muito importante para nos tornarmos mais felizes.

4.2) Você acha que consegue manter a frequência desta atividade?

Usuário:

4.2.1) ~afirmação

4.2.2) ~negacao

4.2.3) ~duvida

Beck:

4.2.1) Que bom!

4.2.2) 4.2.3) Você consegue então substituir esta atividade por outra?

Usuário:

4.2.2.1) ~afirmação

4.2.2.2) ~negacao

4.2.2.3) ~duvida

Beck:

- 4.2.2.1) 4.2.2.3)Essas são algumas das atividades que você disse que gostava de fazer. [listar as atividades que o usuário gosta de fazer, só as lícitas].
4.2.2.2) Tudo bem, vamos continuar.

4.3) Não entendi quantas vezes por semana você faz tal atividade, pode repetir com outras palavras?

Usuário:

- 4.3.1) ~afirmação e frequência
4.3.2) ~afirmação
4.3.3) ~negação

Beck:

- 4.3.1) Reuse o item (4.1)
4.3.2) Quantas vezes na semana você faz [atividade]?

Usuário:

- 4.3.2.1) frequência
4.3.2.2) *

Beck:

- 4.3.2.1) Reuse o item (4.1)
4.3.2.2) Desculpe, eu realmente não entendi quantas vezes na semana você faz [atividade]. Continue assim, é muito importante fazer algo de que goste, isso vai ajudar você a se sentir mais feliz.

4.3.3) Desculpe, eu realmente não entendi quantas vezes na semana você faz [atividade]. Continue assim, é muito importante fazer algo de que goste, isso vai ajudar você a se sentir mais feliz.

5 Beck: (se não atividades) Você me disse anteriormente que não gosta de fazer nenhuma atividade para se divertir. Mas tem algo que você gostaria de fazer e não consegue por algum motivo?

Usuário:

- 5.1) ~afirmação (atividade)
5.2) ~afirmação
5.3) ~negação

Beck:

- 5.1) Entendi. Você gostaria de [atividade].
5.2) Qual a atividade que você gostaria de fazer então (mostrar quadro de atividades lícitas)?

Usuário:

- 5.2.1) (atividade)
5.2.2) nenhuma atividade.

Beck:

- 5.2.1) Entendi. Você gostaria de fazer [atividade].

5.2.2) Tudo bem. Vamos continuar.

5.3) Tudo bem, vamos continuar.

Identificando os pensamentos automáticos negativos

1 **Beck:** Como já falei, os pensamentos têm impacto direto na forma como nos sentimos e nos comportamos.

2 **Beck:** Todo mundo tem pensamentos ruins às vezes.

3 **Beck:** Mas esses pensamentos ruins podem se tornar um problema se eles ocorrerem frequentemente, porque eles nos fazem sentir para baixo, tristes.

4 **Beck:** Os pensamentos ruins também são conhecidos como “pensamentos automáticos negativos”.

5 **Beck:** Veja o quadro ao lado. Neste quadro temos o exemplo dos pensamentos automáticos negativos de três personagens: a Maria, o João e a Joana.

Identificando os pensamentos negativos
Veja o exemplo do pensamento negativo que a Maria tinha: “Todas as meninas da escola são mais bonitas do que eu. Eu me sinto como um monstro ao lado delas. Eu não gosto da maioria das coisas em mim”. Observe o pensamento negativo do João. “Eu nunca vou conseguir uma namorada. Quem iria querer sair com alguém como eu? Eu sou um ninguém”. E a Joana, veja que pensamento negativo dela. “Eu sou tão inútil... Eu não consigo fazer nada direito!”

6 **Beck:** Você tem algum pensamento negativo e que aparece várias vezes na sua cabeça?

Usuário:

6.1) pensamento

6.2) não tenho pensamento negativos

6.3) *

Beck:

6.1) Então você tem este pensamento: [pensamento] ?

Usuário:

6.1.1) ~afirmação

6.1.2) ~negação e pensamento

6.1.3) ~negação

6.1.4) *

Beck:

6.1.1) Ok. Veja, alguns pensamentos negativos podem ser classificados. Por exemplo: (falar de algumas classificações de acordo com o pensamento negativo apresentado pelo adolescente).

6.1.2) Ok. Veja se entendi agora. Você tem este pensamento: [pensamento] ?

Usuário:

- 6.1.2.1) ~afirmação
- 6.1.2.2) ~negação e pensamento
- 6.1.2.3) ~negação
- 6.1.2.4) *

Beck:

- 6.1.2.1) Reuse o item 6.1.1)
- 6.1.2.2) Desculpe não ter entendido. Minha cabeça digital não conhece muitas palavras. Mas vamos continuar a nossa conversa.

6.1.3) 6.1.4) Desculpe não ter entendido. Você poderia dizer novamente com outras palavras? (Reuse o item 6.1.2 só mais uma vez pra não entrar num loop).

6.2) Entendi que você não tem pensamentos negativos, certo?

Usuário:

- 6.2.1) ~afirmação e pensamento
- 6.2.2) ~afirmação
- 6.2.3) ~negação
- 6.2.4) *

Beck:

- 6.2.1) (Reuse o item 6.1.2 só mais uma vez pra não entrar num loop).
- 6.2.2) Tudo bem então. Vamos continuar conversando.
- 6.2.3) Desculpe não ter entendido. Você poderia dizer novamente com outras palavras? (Reuse o item 6.1.2 só mais uma vez pra não entrar num loop).
- 6.2.4) Desculpe não ter entendido. Minha cabeça digital não conhece muitas palavras. Mas vamos continuar a nossa conversa.

6.3) Desculpe não ter entendido. Você poderia dizer novamente com outras palavras? (Reuse o item 6.1.2 só mais uma vez pra não entrar num loop).

7 **Beck:** Como você pode perceber, o pensamento automático negativo é algo que nos atrapalha e pode nos impedir de desenvolver a nossa autoestima.

8 **Beck:** Vamos conversar agora sobre os sentimentos.

Observação: Se o adolescente falar que tem o pensamento negativo de se matar, perguntar para ele se ele não tem algum adulto de sua confiança para que ele possa contar o que pensa. Se ele responder que não tem, dar uma sugestão para ele entrar em contato com a <http://cvv.org.br/>.

Dar também uma opção do link:

http://www.ictc.com.br/arquivos/erros_cognitivos_comuns.pdf para ele saber mais sobre as categorizações dos pensamentos negativos.

Identificando o sentimento face ao pensamento automático negativo

1 **Beck:** Veja, vou mostrar o quadro “O que eu sinto” e gostaria que você escolhesse dentre as opções a que mais se aproxima do seu sentimento.

O que eu sinto com frequência?		
• Angústia • Ansiedade • Autodesprezo • Cansaço • Chateação • Ciúmes • Culpa • Desespero • Estresse • Frustração • Humilhação • Incerteza	• Inveja • Irritado • Insônia • Medo • Preocupação • Raiva • Receio • Sem apetite • Sentimento rejeição • Tristeza • Vergonha	• Dúvidas de si mesmo • Medo de ficar doente • Sentimento de inferioridade • Sentimento de que perdeu peso • Sentimento de punição • Sentimento de fracasso • Vontade de comer • Vontade de se matar • Vontade de chorar • Não sinto nada. • Outros. O que?

2 **Beck:** Você pode escolher mais de uma opção. Se você achar que o seu sentimento atual NÃO está neste quadro, você pode clicar em “Outros” e escrever o que sente.

3 **Beck:** Caso você ache que não sente NADA, quando tem um pensamento negativo, você pode clicar em “Não sinto nada”.

4 **Beck:** Quando terminar é só clicar em Guardar.

Usuário:

4.1) Sintoma

4.2) Outros

4.3) Não sinto nada

Beck:

4.1) 4.2) Ok.

4.3) Entendi que você não sente nada quando tem um pensamento negativo.

5 **Beck:** A melhor maneira de mudar a forma de como nos sentimos é mudar a forma de como pensamos e/ou nos comportamos.

Quando o pensamento automático negativo acontece

1 **Beck:** Para mudarmos a forma como nos sentimos, além de identificarmos os pensamentos negativos, precisamos identificar o que acontece para que eles apareçam nas nossas cabeças.

2 **Beck:** Veja o quadro ao lado. Nele, temos exemplos do porquê o pensamento negativo aparecia na cabeça dos nossos personagens.

Quando o pensamento negativo acontece
• Por exemplo, a Maria, que se achava um monstro, tinha esses pensamentos quando olhava os cabelos das amigas e os comparava com os dela.

- O João sempre pensava que nunca iria conseguir namorar quando via as postagens do Facebook do seu amigo Carlos com a namorada Jéssica.
- E a Joana pensava que não fazia nada direito quando estava na aula de dança, já que não conseguia fazer todos os passos.

3 **Beck:** (Se o adolescente falou dos pensamentos negativos) Você consegue lembrar o que acontece para que os pensamentos negativos apareçam na sua cabeça?

Usuário:

3.1) ~afirmação e motivo

3.2) ~afirmação

3.3) ~negação

Beck:

3.2) E o que acontece para que você pense assim?

Usuário:

3.2.1) motivo

3.2.2) nada

Beck:

3.2.1) Então é isso o que acontece (motivo) para que você tenha os pensamentos negativos?

Usuário:

3.2.1.1) ~afirmação

3.2.1.2) ~negação e motivo

3.2.1.3) ~negação

Beck:

3.2.1.1) Entendi.

3.2.1.1) Sabendo o motivo, podemos pensar nas soluções possíveis para resolver o que está nos atrapalhando.

3.2.1.2) Veja se entendi correto agora. O motivo para que você tenha os pensamentos negativos é este (motivo)?

Usuário:

3.2.1.2.1) ~afirmação

3.2.1.2.2) ~negação e motivo

3.2.1.2.3) ~negação

Beck:

3.2.1.2.1) Reuse o item (3.2.1.1)

3.2.1.2.2) 3.2.1.2.3) Desculpe, eu não ter entendido o motivo. Vou tentar aprender mais pra entender melhor da próxima vez.

3.2.1.3) Desculpe, eu não ter entendido o motivo. Vou tentar aprender mais pra entender melhor da próxima vez.

3.2.2) Tudo bem. Talvez em algum outro momento você lembre o que acontece para que você tenha pensamentos negativos.

4 **Beck:** Precisamos estar sempre atentos para perceber o que nos faz ter pensamentos negativos.

5 **Beck:** Vamos falar agora sobre o comportamento das pessoas.

Identificando o comportamento decorrente do pensamento automático negativo

1 **Beck:** As pessoas podem adotar determinados comportamentos por causa dos pensamentos negativos.

2 **Beck:** (se tem pensamento negativo) Eu gostaria de saber o que você faz quando tem um pensamento negativo.

3 **Beck:** Vou mostrar o quadro “O que eu faço” para que você selecione o comportamento que mais se aproxima do seu.

4 **Beck:** (se não tem pensamento negativo) Você adota algum desses comportamentos?

5 **Beck:** Quando terminar de escolher, é só clicar em “Guardar”.

O que eu faço com frequência?		
• Brigo com as pessoas • Como muito • Choro bastante • Me isolo das pessoas • Machuco a mim mesmo • Faço bastante exercício físico	• Nada • Outro. Qual?	

Usuário:

5.1) comportamento

5.2) outro

5.3) nada

Beck:

5.1) 5.2) Entendi.

5.3) Tudo bem. Vamos prosseguir.

Mudando os pensamentos automáticos negativos

1 **Beck:** Muitos dos nossos pensamentos negativos acontecem porque temos uma reação “exagerada” em determinadas situações.

2 **Beck:** Veja a história de Ana e Margarida no quadro ao lado.

História de Ana e Margarida
• Numa terça-feira à tarde, Ana e Margarida perguntaram às suas amigas se elas queriam ir no sábado ao shopping. • Ambas as amigas responderam que não, porque elas tinham outras coisas para fazer.

- Ana se sentiu rejeitada e pensou: Minha amiga não quer sair comigo porque ela não gosta de mim, ela nunca mais vai sair comigo novamente.
- Mas por outro lado Margarida pensou: Minha amiga está ocupada no sábado, mas podemos sair outra dia. Ela ainda é a minha melhor amiga.

3 **Beck:** Como podemos ver, a mesma situação aconteceu com Ana e Margarida. Mas as reações delas foram bem diferentes.

4 **Beck:** Pensando nas reações de Ana e Margarida. Com qual delas você se identificou mais? Ana ou Margarida?

Usuário:

- 4.1) Ana
- 4.2) Margarida
- 4.3) nenhuma delas
- 4.4) *

Beck:

- 4.1) Certo, você se identificou mais com Ana.
- 4.2) Certo, você se identificou mais com Margarida.
- 4.3) Tudo bem, você deve ter um outro ponto de vista.
- 4.4) Eu me identifiquei mais com a Margarida. Ter uma reação mais positiva nos faz sentir melhor.

5 **Beck:** Agora eu gostaria de saber, para você, qual das duas garotas você acha que teve o pensamento mais positivo. Ana ou Margarida?

Usuário:

- 5.1) Ana
- 5.2) Margarida
- 5.3) *

Beck:

- 5.1) Então, vamos tentar pensar diferente de Ana? Veja, a Margarida foi a que pensou de forma mais positiva. Ela continuou pensando que sua amiga ainda gostava dela e que poderiam sair em um outro dia.
- 5.2) Muito bem! A Margarida pensou de forma positiva. Ela continuou pensando que sua amiga ainda gostava dela e que poderiam sair em um outro dia.
- 5.3) Foi a Margarida que pensou de forma positiva. Ela continuou pensando que sua amiga ainda gostava dela e que poderiam sair em um outro dia.

6 **Beck:** Toda vez que tivermos pensamentos negativos, devemos questionar e desafiar esses pensamentos.

7 **Beck:** Devemos procurar evidências de que são realmente verdadeiros.

7 **Beck:** Para nos sentirmos melhor, devemos procurar ver o lado bom das coisas.

8 **Beck:** Veja outra história no quadro ao lado.

História do Felipe

- O Felipe marcou com os amigos para irem à casa do Tiago conversarem sobre a

viagem que farão no próximo final de semana, no carro de Felipe, para conhecer uma cachoeira no interior, bem distante da cidade.

- Quando Felipe estava indo para a casa do Tiago, o carro dele quebrou no meio do caminho.
- Logo que o carro quebrou, lá no meio da estrada, Felipe começou a ter vários pensamentos, mas como ele é uma pessoa otimista e realista tentou logo ver o lado bom das coisas.

9 **Beck:** Que tipo de pensamento positivo você teria se fosse o Felipe?

Usuário:

9.1) carro quebrou antes da viagem

9.2) não teria pensamentos positivos

9.3) *

Beck:

9.1) Isso mesmo. Felipe teve a oportunidade de consertar o carro antes da longa viagem. Talvez ele não encontrasse uma oficina próxima à cidade do interior onde fica a cachoeira.

9.2) Entendi. Pode ser difícil ter um pensamento positivo nessas horas. Mas Felipe, que procura o lado bom das coisas, pensou: “Que bom que o carro quebrou agora e não durante a viagem”.

9.2) “Talvez eu não encontrasse uma oficina próxima à cidade do interior onde fica a cachoeira”.

9.3) Felipe pensou: “Que bom que o carro quebrou agora e não durante a viagem. Talvez eu não encontrasse uma oficina próxima à cidade do interior onde fica a cachoeira”.

Aprendendo como mudar

1 **Beck:** Mudar não é fácil.

2 **Beck:** E pra que isso aconteça, precisamos reconhecer o que não nos faz bem.

3 **Beck:** Pelo que conversamos até agora, podemos dizer que: Quando isso acontece (motivo), você pensa desta forma (pensamento), se sente deste jeito (sentimento) e se comporta desta maneira (comportamento).

Observação: Verificar se o adolescente respondeu e o chatbot conseguiu capturar todas as respostas para montar a formulação de caso acima. (se ele deixou de responder algo, posso usar a frase “não sabe muito bem como se comporta ou como se sente”, por exemplo).

4 **Beck:** A nossa conversa de hoje termina aqui. Já falamos bastante e foi muito proveitoso.

5 **Beck:** Na próxima vez em que conversarmos, falaremos mais sobre os seus pensamentos, sentimentos e comportamentos para que eu possa entender melhor você.

6 **Beck:** Até breve, [usuario]!

Despedida

1 **Usuário:** tchau

Beck:

1.1) Precisa ir agora mesmo?

Usuário:

1.1) ~afirmação

1.2) ~negação

Beck:

1.1) Até breve então.

1.2) Que bom, vamos conversar mais um pouco então.

Bibliografia

- BECK, J. S. *Cognitive behavior therapy: Basics and beyond*. Guilford Press, 1995.
- CLARKE, G.; LEWINSOHN, P.; HOPS, H. *Leader's manual for adolescent groups: adolescent coping with depression course*. Castalia, 1990.
- CLARKE, G.; LEWINSOHN, P.; HOPS, H. *Student workbook: Adolescent coping with depression course*. Portland, OR: Kaiser Permanente, 1990.
- CVV.ORG. *Centro de Valorização da Vida*. 2015. <http://www.cvv.org.br/>. Acesso em: 30 mai. 2017.
- COLLINS-DONNELLY, K. *Banish Your Self-esteem Thief: A Cognitive Behavioural Therapy Workbook on Building Positive Self-esteem for Young People*. Jessica Kingsley Publishers, 2014.
- KNAPP, P.; BECK, A. T. Fundamentos, modelos conceituais, aplicações e pesquisa da terapia cognitiva Cognitive therapy: foundations, conceptual models, applications and research. *Rev Bras Psiquiatr*, v. 30, n. Supl II, p. S54-64, 2008.
- FERREIRA, A. B de H. *Mini Aurélio: o dicionário da língua portuguesa*. Curitiba: Positivo, 2008.
- NEENAN, M.I; DRYDEN, W. *Cognitive behaviour therapy: 100 key points and techniques*. Routledge, 2014.
- STALLARD, P. *Bons pensamentos-Bons sentimentos*. Artmed Editora, 2007.
- ZUGMAN, S. *Erros cognitivos comuns*. 2011. www.ictc.com.br/arquivos/erros_cognitivos_comuns.pdf. Acesso em: 11 mar. 2017

APÊNDICE B – INSTALAÇÃO DO AMBIENTE

Instalando o Ambiente para o Chatbot Beck (versão 3.0 Data: 12/12/16)

O *Chatbot Beck* foi implementado no sistema operacional Linux, e pode ser acessado através de um *browser* da Internet. Descreveremos a seguir os programas necessários para instalar o *Chatbot Beck*.

Programas necessários:

- Linux64bits Ubuntu 16.04.1 LTS {Já vem com o Mozilla FireFox}
- python2.7 { instalar modulos: pip 8.1.2, numpy 1.11.1}
- NLTK3.2.1 {instalar os modulos: Punkt Tokenizer Models (punkt) RSLP Stemmer (rslp) }
- NLPNET1.2.1
- ChatScript-6.7a
- Apache2
- Php7.0
- Postgresql 9.5.5
- Hunspell 1.3.3
- SendMail

Instalando Programas:

1. Linux64bits Ubuntu 16.04.1 LTS

O ChatScript (Linguagem em que foi escrito o Beck-bot) é independente de plataforma (pode ser executado no sistema operacional Linux, Windows e OS X (computadores da Apple)).

Optei por usar o sistema operacional Linux porque a maioria dos demais programas são nativamente escritos em Python, que por sua vez possui uma maior e melhor integração no Linux.

Dentre as distribuições do Linux optei pela Ubuntu porque é uma das mais conhecidas e usadas e possui um grande fórum para tirar dúvidas.

A versão atual do Ubuntu, até a data da criação deste documento, é 16.04.1 LTS. O Linux Ubuntu pode ser baixado através da página: <http://www.ubuntu.com/download/desktop>

Instalamos o Linux no computador através de um pendrive utilizando o aplicativo: Rufus (que pode ser baixado diretamente da página do Ubuntu já mencionada). Todas as instruções de como Instalar o Linux Ubuntu pode ser encontrada através da sua página.

2. Python2.7

O python2.7 já vem instalado no linux Ubuntu. O python, linguagem de programação, pode ser encontrado no diretório: /usr/local/lib/python2.7

Escolhemos a versão 2.7 do python porque a princípio, o nlpnet exige esta versão.

2.1 PIP 8.1.2 (Python Package Index)

PIP é uma ferramenta de instalação de pacotes do python. Esta ferramenta facilita a instalação de pacotes na linguagem python. Mais informações em: <https://wiki.python.org/moin/CheeseShopTutorial>. Optamos por ela porque a maioria dos módulos a utiliza como facilitadora para instalação.

Para instalação do pip no python foi realizado os seguintes passos:

Abrir uma janela do terminal do linux.

Digitar os comandos: `sudo apt-get update` ← para atualizar a lista de pacotes disponíveis no sistema.

`sudo apt-get upgrade` ← para atualizar os pacotes no sistema.

`sudo apt-get install python-pip` ← para instalar o pacote pip no python

Se você quiser saber se o PIP foi corretamente instalado no python, você pode executar o comando a partir do terminal do Linux: `pip -V`

Como o PIP instalado, a instalação dos demais programas fica muito fácil.

2.2 Numpy 1.11.1

Numpy é uma biblioteca do python para cálculos científicos. Dentre outras coisas, o Numpy provê o objeto array de forma multidimensional (permindo criar matrizes e arrays marcardos). Mais informações: <http://docs.scipy.org/doc/numpy/user/whatisnumpy.html>

Para instalar o Numpy basta executar o seguinte comando: `sudo pip install -U numpy` (o -U é para forçar a realização de um upgrade antes da instalação).

3. NLTK 3.2.1

NLTK é uma plataforma para construção de programas em python para Processamento de Linguagem Natural (PLN). Ela provê mais de 50 corpora e recursos lexicais.

Para instalar o NLTK é só executar o seguinte comando: `sudo pip install -U nltk`.

3.1 Punkt Tokenizer Models

É um modulo dos recursos lexicos do NLTK que é usado para realizar a divisão do texto em listas de sentenças. Mais informações em: <http://www.nltk.org/api/nltk.tokenize.html>

Para baixar este recurso léxico você deve executar o python e em seguida digitar os seguintes comandos:

```
import nltk
nltk.download()
```

Será aberta uma janela com as opções de pacotes dados e recursos lexicais disponíveis para download. É só procurar punkt.

Obs. É interessante previamente configurar um diretório padrão para armazenar esses dados(escolhi o /usr/local/share/nltk_data – criei a pasta nltk_data neste caminho). Na mesma janela em que estão listados os pacotes a serem baixados existe a opção de configurar o diretório.

3.2 RSLP Stemmer (Removedor de Sufixo da Língua Portuguesa)

RSLP é um algoritmo (recurso lexical do nltk) usado para reduzir palavras do português do Brasil para a sua forma canônica (“raiz da palavra”. Exemplo: pedra → pedr)

Para baixar este recurso léxico você deve executar o python e em seguida digitar os seguintes comandos:

```
import nltk
nltk.download()
```

Será aberta uma janela com as opções de pacotes dados e recursos lexicais disponíveis para download. É só procurar rslp.

Obs. É interessante previamente configurar um diretório padrão para armazenar esses dados. Na mesma janela em que estão listados os pacotes a serem baixados existe a opção de configurar o diretório que pode e até deve ser o mesmo do punkt (módulo anterior).

4. NLPNET1.2.1

NLPNET é uma biblioteca do python para PLN baseada em redes neurais. Atualmente ela faz POS Tagging que faz a rotulação do papel semântico. Mais informações em: <http://nilc.icmc.sc.usp.br/nlpnet/index.html> ou através do artigo abaixo.

Fonseca, E. R. and Rosa, J.L.G. *Mac-Morpho Revisited: Towards Robust Part-of-Speech Tagging*. Proceedings of the 9th Brazilian Symposium in Information and Human Language Technology, 2013. p. 98-107

Para instalar o NLPNET basta executar o comando a seguir: `pip install nlpnet`

O NLPNET é dependente do NLTK e do Numpy, como também do recurso lexical punkt.

Para usar o POS-Tagging é necessário baixar o pacote de dados disponível em: nilc.icmc.sc.usp.br/nlpnet/data/pos-pt.tgz

O arquivo deve ser descompactado e, como sugestão colocar no mesmo diretório onde foram baixados o punkt e o rslp.

Exemplo de como usar o NLPNET (estando no python):

```
>>> import nlpnet
>>> nlpnet.set_data_dir('/path/to/nlpnet-data/')
>>> tagger = nlpnet.POSTagger()
>>> tagger.tag('O rato roeu a roupa do rei de Roma.')
[[('O', 'ART'), ('rato', 'N'), ('roeu', 'V'), ('a', 'ART'),
('roupa', 'N'), ('do', 'PREP+ART'), ('rei', 'N'), ('de', 'PREP'),
('Roma', 'NPROP'), ('.', 'PU')]]
```

Observação: No NLPNET eu alterei o arquivo taggers.py para colocar as minhas customizações (usar o RLSP e o Hunspell).

5. ChatScript-6.91

É propriamente uma linguagem usada para a construção do Beck-bot. Ela pode ser baixada através do endereço: <https://sourceforge.net/projects/chatscript/>. É só baixar e começar a usar. Tem versões para Windows, Linux e OS X.

Como o chatscript é feito intrinsecamente para o inglês, é preciso fazer alguns ajustes, como deleções de alguns arquivos desnecessários para a língua portuguesa que pode atrapalhar um pouco o funcionamento do bot.

Procedimentos para configuração do Chatscript:

Deletar todo o conteúdo da pasta /DICT/ENGLISH

Deletar todo o conteúdo da pasta /LIVEDATA/ENGLISH

Deletar todo o conteúdo da pasta /RAWDATA/WORLDDATA

No arquivo /Chatscript/RAWDATA/NomeDoSeuBot/simplecontrol.top você precisa deixar a variável: `$cs_token = #NO_HYPHEN_END | #DO_NUMBER_MERGE | #DO_DATE_MERGE` dessa maneira.

No arquivo /Chatscript/RAWDATA/NomeDoSeuBot/simpletopic.top você precisa customizá-lo para “conversar com a ferramenta externa de POS-Tagging”. Este arquivo é que executará o pos-tag.py descrito logo a seguir.

Como a pouco tempo foi disponibilizado no Chatscript o uso de uma ferramenta externa para a realização do POS-Tagging, foi criada a pasta SCRIPTS dentro do Chatscript e foi criado o script pos-tag.py (em python) para o Chatscript conseguir se comunicar com o NLTK e o NLPNET.

O chatscript também precisa do package libcurl4-openssl-dev para funcionar no Ubuntu. Para instalar é só executar o comando:

```
sudo apt-get install libcurl4-openssl-dev
```

Para instalar o ChatScript na cron do servidor para que ele automaticamente execute numa eventual queda devemos realizar os seguintes passos.

```
sudo vim /etc/crontab
```

Adicionar a linha abaixo:

```
0,5,10,15,20,25,30,35,40,45,50,55 * * * * beck cd /home/beck/ChatScript/BINARIES  
&& ./LinuxChatScript64 >> ../LOGS/cronserver.log 2>&1
```

Desta forma o ChatScript será executado de 5 em 5 minutos e vai gravar um log dentro da pasta logs.

Detalhes sobre a cron posso ver no site: <http://www.ubuntudicas.com.br/2012/01/crontab-o-agendador-de-tarefas-do-linux/>

Para ver se a cron está no ar, ou parar o reiniciar executar os seguintes comandos:

```
sudo service cron status  
sudo service cron stop  
sudo service cron start  
sudo service cron restart
```

6. Apache2 Postgresql 9.5.5 e PHP7.0

Esses softwares são usados para permitir o acesso ao Beck-bot através da Web. Processo de instalação no Linux Ubuntu.

A partir de uma janela do terminal executar os seguintes comandos:

```
sudo apt-get update ← para atualizar o sistema  
sudo apt-get install apache2 ← para instalar o apache2
```

```
http://localhost ← para testar a instalação do apache
```

`curl http://icanhazip.com` ← para descobrir seu ip externo.

`sudo apt-get install postgresql` ← para instalar o postgresql

`sudo apt-get install php7.0 libapache2-mod-php7.0 php7.0 php7.0-pgsql` ← Para instalar o PHP e os outros recursos de conexão entre o php e o postgresql.

Para ver o log de erros do PHP enquanto ele estiver sendo executado basta executar o comando:

```
tail -f /var/log/apache2/error.log
```

```
apt-get -y install postgresql postgresql-contrib phpPgadmin
```

Para alterar a ordem de prioridade da pagina inicial que é carregada quando acessamos o site. Geralmente são as páginas HTML que são carregadas primeiro, através da alteração do arquivo `dir.conf` podemos alterar isso para php (precisa ter privilégios do usuário root).

```
sudo vi /etc/apache2/mods-enabled/dir.conf
```

Antes:

```
<IfModule mod_dir.c>
    DirectoryIndex index.html index.cgi index.pl index.php index.xhtml
    index.htm
</IfModule>
```

Depois:

```
<IfModule mod_dir.c>

    DirectoryIndex index.php index.html index.cgi index.pl index.xhtml
    index.htm
</IfModule>
```

Para esta alteração entrar em vigor é necessário resetar o serviço do apache2 ou o postgresql.

```
sudo service apache2 restart
```

Outra forma de reiniciar:

```
/etc/init.d/postgresql restart
```

Os arquivos do site devem residir na pasta:

```
/var/www/html/
```

Para o postgresql é necessário habilitar duas DLL's no arquivo php.ini

```
sudo vim /etc/php/7.0/apache2/php.ini
```

```
extension=php_pdo_pgsql.dll
```

```
extension=php_pgsql.dll
```

Como também alterar o arquivo de configuração [pg_hba.conf](#) para que o postgresql peça senha quando for acessado:

```
sudo vim /etc/postgresql/9.5/main/pg_hba.conf
```

Alterar para md5 como está destacado em [azul](#).

```
# Database administrative login by Unix domain socket

local all          postgres          md5

# TYPE DATABASE      USER        ADDRESS          METHOD

# "local" is for Unix domain socket connections only

local all          all                md5
```

Para acessar o postgresql basta executar o comando: `sudo -u postgres psql`

`pg_restore -d chat -U postgres chat-postgres.backup` ← Para recuperar um backup de uma base de dados a partir do terminal.

Para que o chatscript funcione com o postgresql deve instalar também o pacote:

```
sudo apt-get install libpq-dev
```

E criar o executável do Chatscript que roda o Postgresql que é o arquivo `ChatScript/BINARIES/ChatScriptpg`

Para criar o executável basta estar no diretório: `ChatScript/SRC` e digitar o comando: `make pgserver`

Pronto, neste momento ele deve criar o arquivo `ChatScriptpg` no diretório `/ChatScript/BINARIES`.

Executar o `ChatScriptpg` para que funcione com o Postgresql.

APÊNDICE C – AUTORIZAÇÃO ESCOLA

Solicitação para Realizar Teste de Software com Alunos

Título da Pesquisa: Beck: Um *Chatbot* Baseado na Terapia Cognitivo-Comportamental para Apoiar Adolescentes com Depressão.

Nome do Pesquisador: Oberdan Alves de Almeida Junior

Aluno do curso de mestrado do Centro de Informática –

UFPE.

Nome da Orientadora: Flávia de Almeida Barros

Professora do Centro de Informática – UFPE.

1. Sobre o Teste de Software

Este teste de *software* tem por finalidade avaliar o desempenho do *chatbot*¹ Beck, cujo objetivo é conversar e apoiar adolescentes que sofrem de depressão, ou que desejam evitá-la.

Esse sistema foi desenvolvido como parte de um projeto de mestrado de Oberdan Almeida sob orientação da professora Flávia Barros.

O público-alvo desta pesquisa são adolescentes entre 15 e 18 anos de idade que tenham um pouco de prática com o manuseio do computador e da Internet.

2. Envolvimento no Teste

A participação dos adolescentes nesse teste ajudará no desenvolvimento da pesquisa de Mestrado de Oberdan, auxiliando na melhoria do funcionamento do *chatbot* Beck.

O responsável pela escola tem liberdade de recusar que os seus alunos participem desta pesquisa, sem qualquer prejuízo.

3. Sobre a realização do Teste

O teste será realizado por meio de um computador. A conversa com o *chatbot* Beck será feita através de uma página da Internet que conterá o diálogo entre o *chatbot* e o usuário, figuras e formulários sobre os princípios da Terapia Cognitivo-Comportamental² (TCC).

No final da interação (diálogo) com o *chatbot* Beck, será apresentado um formulário eletrônico para que o adolescente responda algumas perguntas sobre essa experiência com o *chatbot*. Essas perguntas estão descritas nas páginas 3 e 4

1 Sistema de computador capaz de conversar com os usuários através de linguagem natural.

2 Método da psicoterapia que mostra para o indivíduo que os seus pensamentos podem influenciar os seus comportamentos e suas emoções.

deste documento. É importante enfatizar que nenhum adolescente será identificado. Essa pesquisa é totalmente anônima.

4. Confidencialidade

Todas as informações coletadas neste teste são estritamente confidenciais, sendo preservado o respeito ao anonimato do adolescente. Somente o pesquisador terá conhecimento dos dados, exceto a identificação do adolescente, que não será solicitada ou registrada em momento algum.

As informações sobre as estatísticas da pesquisa poderão ser divulgadas e publicadas respeitando o que foi exposto no parágrafo anterior.

6. Riscos e Benefícios:

A participação nesta pesquisa não terá nenhum risco.

A participação nesta pesquisa não terá benefício direto. Entretanto, esperamos que esse teste forneça para o pesquisador informações importantes sobre a interação de adolescentes com o *chatbot Beck*. Para o adolescente, a contribuição da pesquisa será de mostrar que ele não estará sozinho, caso algum dia venha a ter problemas de depressão.

Oberdan Alves de Almeida Junior
Mestrando CIn/UFPE
(Pesquisador Responsável)
Contato: 81 98746-6007

Consentimento Livre e Esclarecido

Declaro que li as informações acima sobre a pesquisa e que me sinto perfeitamente esclarecido sobre o conteúdo da mesma. Declaro ainda que permito que os alunos desta escola, dentro da faixa etária da pesquisa, possam participar se assim o desejarem.

Recife, ____ / ____ / _____

Assinatura do Pesquisador:

Assinatura Diretor Escola:

Formulário de Avaliação do Chatbot Beck

Objetivo: Este questionário será utilizado para avaliar o desempenho e a utilidade do *chatbot* Beck para seus usuários.

Perfil do Participante: Adolescentes com um pouco de prática no manuseio do computador e da Internet.

Tempo estimado: 30 minutos.

Consentimento de participação e confidencialidade: Respondendo este questionário, você consente com a sua participação nesta pesquisa. Esse questionário é anônimo.

Endereço de acesso ao *chatbot*: <http://beck-bot.cin.ufpe.br/>

Perfil do usuário:

Idade:

- ☐ 15 anos.
- ☐ 16 anos.
- ☐ 17 anos.
- ☐ 18 anos.

Sexo:

- ☐ Masculino.
- ☐ Feminino.

Você já tinha ouvido falar ou conversou com um *chatbot*?

- ☐ Sim.
- ☐ Não.

Questões:

Você considera que foi interessante conversar com Beck?

- ☐ Discordo Totalmente.
- ☐ Discordo.
- ☐ Não Tenho Certeza.
- ☐ Concordo.
- ☐ Concordo Totalmente.

Você aprendeu alguma coisa com essa conversa?

- ☐ Discordo Totalmente.
- ☐ Discordo.
- ☐ Não Tenho Certeza.

- ☐ Concordo.
- ☐ Concordo Totalmente.

Beck conseguiu entender as suas perguntas e respostas?

- ☐ Discordo Totalmente.
- ☐ Discordo.
- ☐ Não Tenho Certeza.
- ☐ Concordo.
- ☐ Concordo Totalmente.

Você compreendeu que os seus pensamentos podem influenciar o que você sente e a forma como você se comporta?

- ☐ Discordo Totalmente.
- ☐ Discordo.
- ☐ Não Tenho Certeza.
- ☐ Concordo.
- ☐ Concordo Totalmente.

Você conversaria novamente com Beck?

- ☐ Discordo Totalmente.
- ☐ Discordo.
- ☐ Não Tenho Certeza.
- ☐ Concordo.
- ☐ Concordo Totalmente.

Comentários: