

**UNIVERSIDADE DO SAGRADO CORAÇÃO**

**MOACIR LUIZ DA SILVA JUNIOR**

**DESENVOLVIMENTO DE UM CHATBOT PARA O  
ATENDIMENTO DE CLIENTES DE UMA EMPRESA  
UTILIZANDO O IBM WATSON**

BAURU  
2017

**MOACIR LUIZ DA SILVA JUNIOR**

**DESENVOLVIMENTO DE UM CHATBOT PARA O  
ATENDIMENTO DE CLIENTES DE UMA EMPRESA  
UTILIZANDO O IBM WATSON**

Trabalho de Conclusão de Curso apresentado ao centro de Ciências Exatas e Sociais Aplicadas como parte dos requisitos para obtenção do título de bacharel em Ciência de Computação, sob orientação do Prof. Me. Patrick Pedreira Silva.

BAURU  
2017

S5865d	<p data-bbox="497 1370 861 1406">Silva Junior, Moacir Luiz da</p> <p data-bbox="497 1451 1299 1550">Desenvolvimento de um chatbot para o atendimento de clientes de uma empresa utilizando o IBM Watson / Moacir Luiz da Silva Junior. -- 2017.</p> <p data-bbox="539 1559 638 1594">75f. : il.</p> <p data-bbox="539 1639 1114 1675">Orientador: Prof. M.e Patrick Pedreira Silva.</p> <p data-bbox="497 1720 1299 1818">Trabalho de Conclusão de Curso (Graduação em Ciência da Computação) - Universidade do Sagrado Coração - Bauru – SP.</p> <p data-bbox="497 1863 1299 1930">1. Chatbot. 2. Atendimento. 3. Clientes. 4. Funcionários. 5. Base de conhecimento. I. Silva, Patrick Pedreira. II. Título.</p>
--------	--

**MOACIR LUIZ DA SILVA JUNIOR**

**DESENVOLVIMENTO DE UM CHATBOT PARA O ATENDIMENTO DE  
CLIENTES DE UMA EMPRESA UTILIZANDO O IBM WATSON**

Trabalho de Conclusão de Curso apresentado ao centro de Ciências Exatas e Sociais Aplicadas como parte dos requisitos para obtenção do título de bacharel em Ciência de Computação, sob orientação do Prof. Me. Patrick Pedreira Silva.

Bauru, 28 de novembro de 2017.

Banca examinadora:

---

Prof. Me. Patrick Pedreira Silva  
Universidade do Sagrado Coração

---

Prof. Dr. Elvio Gilberto da Silva  
Universidade do Sagrado Coração

---

Prof. Me. Renan Caldeira Menechelli  
Universidade do Sagrado Coração

## **AGRADECIMENTOS**

Primeiramente agradeço aos meus pais pelo apoio e por ajudarem com meu transporte de Botucatu até Bauru através do ônibus, permitindo o meu estudo no curso de Ciência da Computação na Universidade do Sagrado Coração.

Ao orientador do Trabalho de Conclusão de Curso e professor Me. Patrick Pedreira Silva, por me ajudar a encontrar o IBM Watson, que é uma ferramenta excelente para o desenvolvimento de *chatbots*, além de sua experiência com a área de Inteligência Artificial, que foi de grande importância para este projeto.

Ao corpo docente do curso de Ciência da Computação, que sempre me guiou durante todo o curso e me mostrou como a área de Computação pode ser aplicada em diferentes setores.

Sou grato aos meus amigos da Universidade e de Botucatu, pelo apoio e suporte desde o início do curso.

## RESUMO

Este trabalho apresenta a criação de um *chatbot* responsável pelo atendimento de clientes de uma empresa, utilizando conceitos de Inteligência Artificial (IA), Processamento de Linguagem Natural (PLN) e Aprendizado de Máquina (AM). O IBM Watson foi utilizado como ferramenta para a base de conhecimento e desenvolvimento. O levantamento bibliográfico foi efetuado antes de tudo, com a busca de material relacionado a IA, PLN, AM, *chatbots*, Engenharia de Software e sistemas para desenvolvimento utilizados no mercado atualmente, como o IBM Watson. Informações importantes foram levantadas através de entrevistas com os funcionários e pesquisas pelo site da empresa. Os assuntos procurados pelos clientes também foram pesquisados por meio do setor de atendimento. Com isto, a base de conhecimento foi planejada e modelada para o *chatbot*, que foi testado com funcionários e clientes para alcançar um desempenho satisfatório. Através dos testes, foi possível refinar a base de conhecimento, expandindo as informações englobadas pela mesma. Por fim, o *chatbot* foi implementado no Facebook da empresa, aumentando a velocidade do atendimento e otimizando inúmeros processos.

**Palavras-chave:** Chatbot. Atendimento. Clientes. Funcionários. Base de conhecimento.

## **ABSTRACT**

This work presents the creation of a chatbot responsible for attendance of a company's customers, using concepts of Artificial Intelligence (AI), Natural Language Processing (NLP) and Machine Learning (ML). IBM Watson was used as a tool for the knowledge base and development. The bibliographic survey was accomplished before everything, with the search of material related to AI, NLP, ML, chatbots, Software Engineering and systems for development used in market today, such as IBM Watson. Important information was raised through interviews with employees and surveys on the company's website. The subjects sought by customers were also researched through the attendance sector. With this, the knowledge base was designed and modeled for chatbot, which has been tested with employees and customers to achieve satisfactory performance. Through the tests, it was possible to refine the knowledge base by expanding the information encompassed by it. Lastly, chatbot was implemented on the company's Facebook, increasing the speed of attendance and optimizing countless processes.

**Keywords:** Chatbot. Attendance. Customers. Employees. Knowledge base.

## LISTA DE ILUSTRAÇÕES

Figura 1 – Exemplo de uma conversa com Eliza .....	31
Figura 2 – Exemplo de uma conversa com Alice Seguros .....	32
Figura 3 – Exemplo de uma conversa com Cloudia .....	33
Figura 4 – Exemplo de uma conversa com IFood Pizza Bot .....	33
Figura 5 – Exemplo de uma conversa com Waypoint .....	34
Figura 6 – Exemplo de uma conversa com Mr Joboto .....	34
Figura 7 - Exemplo de uma conversa com Ben.....	35
Figura 8 - Exemplo de uma conversa com Ana.....	35
Figura 9 - Arquitetura da API Conversation.....	45
Figura 10 - Exemplo de uma intenção.....	46
Figura 11 - Exemplo de uma entidade .....	47
Figura 12 - Exemplo de um ramo de diálogo.....	48
Figura 13 - Exemplo de uma conversa.....	49
Figura 14 - Histórico de conversas com usuários.....	50
Figura 15 - Diagrama de Caso de Uso do chatbot .....	54
Figura 16 - Diagrama de Atividades do chatbot .....	55
Figura 17 - Exemplos de intenções cadastradas.....	56
Figura 18 - Exemplos cadastrados na intenção “#AplicaçãoProdutos” .....	57
Figura 19 - Exemplos de entidades cadastradas no Watson .....	57
Figura 20 - Produtos cadastrados na entidade “@ProdutosQuímicos” .....	58
Figura 21 - Assuntos cadastrados no Watson.....	59
Figura 22 - Ramo de diálogo “Utilização” .....	60
Figura 23 - Ramo de diálogo “Localização da Empresa” .....	61
Figura 24 - Exemplo de uma conversa.....	62
Figura 25 - Área de criação de um novo App.....	63
Figura 26 - Área de criação do App “SDK for Node.JS” .....	63
Figura 27 - Credenciais do Conversation .....	64
Figura 28 - Campo de Geração de token .....	65
Figura 29 - Campo Webhooks.....	65
Figura 30 - Estrutura do sistema .....	66
Figura 31 - Conversa com o chatbot através do Messenger .....	67
Figura 32 - Mensagens registradas.....	68

Figura 33 - Conversa registrada.....	68
Figura 34 - Mensagens de um cliente .....	69
Figura 35 - Outras dúvidas dos clientes .....	69
Figura 36 - Exemplo de uma detecção incorreta.....	70
Figura 37 - Área para editar a Intenção.....	70

## LISTA DE ABREVIATURAS E SIGLAS

AM	- Aprendizado de Máquina
API	- Application Programming Interface
APP	- Application
CMU	- Carnegie Mellon University
DEC	- Digital Equipment Corporation
FAQs	- Frequently Asked Questions
GPS	- General Problem Solver
IA	- Inteligência Artificial
IBM	- International Business Machines
LT	- Logic Theorist
MCC	- Microelectronics and Computer Consortium
MIT	- Massachusetts Institute of Technology
PLN	- Processamento de Linguagem Natural
SPLN	- Sistema de Processamento de Linguagem Natural
UML	- Unified Modeling Language

## SUMÁRIO

<b>1 INTRODUÇÃO</b> .....	11
<b>2 OBJETIVOS</b> .....	13
2.1 OBJETIVO GERAL .....	13
2.2 OBJETIVOS ESPECÍFICOS .....	13
<b>3 INTELIGÊNCIA ARTIFICIAL</b> .....	14
3.1 FUNDAMENTOS DE IA .....	14
3.2 HISTÓRIA DA IA .....	15
<b>3.2.1 A GestaçãO da Inteligência Artificial (1943-1955)</b> .....	15
<b>3.2.2 O nascimento da IA (1956)</b> .....	16
<b>3.2.3 Entusiasmo inicial, grandes expectativas (1952-1969)</b> .....	16
<b>3.2.4 Uma dose de realidade (1966-1973)</b> .....	17
<b>3.2.5 Sistemas Baseados em Conhecimento (1969-1979)</b> .....	17
<b>3.2.6 A IA se Torna uma Indústria (De 1980 Até a Atualidade)</b> .....	18
3.3 AGENTES INTELIGENTES .....	18
<b>3.3.1 Ambientes</b> .....	19
3.4 APLICAÇÕES DE IA .....	20
3.5 PROCESSAMENTO DE LINGUAGEM NATURAL .....	21
<b>3.5.1 Teste de Turing</b> .....	22
<b>3.5.2 Evolução dos Sistemas de PLN</b> .....	23
<b>3.5.3 Aplicações de PLN</b> .....	24
<b>3.5.4 Análise Léxico-morfológica</b> .....	25
<b>3.5.5 Análise Sintática</b> .....	25
<b>3.5.6 Análise Semântica</b> .....	26
<b>3.5.7 Análise Pragmática</b> .....	27
3.6 APRENDIZADO DE MÁQUINA .....	27
3.7 CHATBOTS .....	29
<b>3.7.1 Exemplos de Chatbots</b> .....	31
<b>4 ENGENHARIA DE SOFTWARE</b> .....	36
4.1 UML .....	37
<b>4.1.1 Diagramas UML</b> .....	38
4.2 TESTE DE SOFTWARE .....	38
<b>5 IBM WATSON</b> .....	41

5.1 APLICAÇÕES DO WATSON .....	41
5.2 APIS DO WATSON .....	42
5.3 CONVERSATION.....	44
<b>5.3.1 Intenções.....</b>	<b>45</b>
<b>5.3.2 Entidades .....</b>	<b>47</b>
<b>5.3.3 Diálogo .....</b>	<b>47</b>
<b>5.3.4 Testes e Melhorias .....</b>	<b>49</b>
<b>6 TRABALHOS CORRELATOS.....</b>	<b>51</b>
<b>7 METODOLOGIA .....</b>	<b>53</b>
7.1 MODELAGEM UML DO CHATBOT .....	53
7.2 BASE DE CONHECIMENTO .....	55
<b>7.2.1 Intenções.....</b>	<b>55</b>
<b>7.2.2 Entidades .....</b>	<b>57</b>
<b>7.2.3 Diálogo .....</b>	<b>58</b>
7.3 TESTES E MELHORIAS.....	61
7.4 INTEGRAÇÃO DO CHATBOT .....	63
7.5 ESTRUTURA DO SISTEMA .....	66
<b>8 RESULTADOS E DISCUSSÕES.....</b>	<b>67</b>
<b>9 CONSIDERAÇÕES FINAIS.....</b>	<b>71</b>
<b>REFERÊNCIAS.....</b>	<b>73</b>

## 1 INTRODUÇÃO

A Inteligência Artificial (IA) pode ser definida como uma área da Ciência da Computação que procura desenvolver agentes inteligentes, sistemas que podem se adaptar a diversas situações e responder da melhor forma possível.

Dentro da IA existe um subcampo chamado Processamento de Linguagem Natural (PLN), que busca criar softwares para a resolução de alguma tarefa, dependendo de informações apresentadas em língua natural, que podem ser coletadas de forma textual ou sonora.

Além disto, há o subcampo denominado Aprendizado de Máquina (AM), que tem como objetivo criar sistemas que aprendam baseados em experiências anteriores, escolhendo a melhor opção. Para isto, é utilizado o método da indução, que busca gerar conclusões através de exemplos.

Neste contexto existem os *chatbots*, agentes inteligentes que se comunicam com pessoas através de texto e possuem inúmeras finalidades. Para gerar respostas aos usuários, são necessárias bases de conhecimento, com informações sobre o assunto abordado pelo *chatbot*, que variam de acordo com a aplicação do mesmo.

Por causa do crescimento da Internet, os *chatbots* se tornaram acessíveis a todos, e ao mesmo tempo, os sites tiveram a necessidade de uma interface para fornecer informações de forma rápida e eficiente às pessoas.

No Brasil há vários *chatbots* que podem auxiliar as pessoas de diferentes formas, sendo destinados ao tratamento de problemas sociais, pesquisa de viagens online, busca de oportunidades de trabalho, entre outras.

Atualmente as empresas buscam conversar com seu público da melhor forma possível e vários processos podem ser melhorados com a implementação de *chatbots*, como o atendimento de clientes, venda de produtos, informações de serviços, entre outros.

Outra área importante é a Engenharia de Software, que visa garantir a qualidade de projetos, utilizando métodos e ferramentas, além de abordar a importância de se desenvolver softwares de acordo com as necessidades do cliente.

A *International Business Machines* (IBM) foi responsável pelo desenvolvimento do Watson, que é um sistema cognitivo utilizado para múltiplas finalidades, sendo a conversação por texto uma das principais.

O Watson possui uma interface que trabalha com Intenções, Entidades e Diálogo. Uma Intenção é uma finalidade apresentada por um cliente, como a resposta de uma pergunta. Através dela, o Watson pode escolher o fluxo de diálogo correto para a resposta. Uma Entidade representa um tipo de dado que é importante para a finalidade de um usuário. Ao reconhecer as entidades, é possível escolher ações específicas para cumprir uma intenção. O diálogo utiliza as Intenções e Entidades, além do contexto do aplicativo, para responder o cliente de forma adequada. Também há uma área de testes do *chatbot*, para verificar as respostas e refinar a base de conhecimento.

Neste contexto, a empresa estudada é do ramo de Tanatopraxia e possui muitos produtos e serviços à disposição de uma quantidade significativa de clientes, procurando sempre atendê-los da melhor forma. Há inúmeros assuntos que são abordados com frequência e de forma repetitiva, sobrecarregando o setor de atendimento e atrasando vários processos.

Por meio do *chatbot* desenvolvido, o foco é tirar dúvidas rapidamente e, conseqüentemente, diminuir a quantidade de trabalho do setor de atendimento. Os temas abordados são essenciais para o cliente, como as funções e propriedades de produtos, informações sobre cursos da empresa, localização, assessoria, exportação, dúvidas técnicas, entre outros.

## 2 OBJETIVOS

A seguir serão apresentados os objetivos deste trabalho.

### 2.1 OBJETIVO GERAL

Desenvolver um *chatbot* empresarial de relacionamento com o cliente utilizando o IBM Watson.

### 2.2 OBJETIVOS ESPECÍFICOS

- a) Efetuar o levantamento bibliográfico;
- b) Adquirir informações importantes sobre a empresa;
- c) Pesquisar sobre os assuntos procurados pelos clientes com o setor de atendimento;
- d) Planejar e modelar a base de conhecimento;
- e) Desenvolver o *chatbot* através das informações coletadas utilizando a ferramenta IBM Watson;
- f) Realizar testes com o *chatbot*;
- g) Refinar a base de conhecimento;
- h) Integrar o *chatbot* com o Facebook da empresa.

### 3 INTELIGÊNCIA ARTIFICIAL

A Inteligência Artificial (IA) pode ter várias definições, dependendo da estratégia para estudo utilizada, que pode estar relacionada a processos de pensamento e raciocínio, comportamento, fidelidade ao desempenho humano e sucesso em comparação ao conceito de racionalidade. (RUSSEL; NORVIG, 2013).

De acordo com Winston (1992), a IA tem como foco a construção de sistemas capazes de perceber, raciocinar e agir.

Segundo Rich e Knight (1994), a IA procura desenvolver computadores para a realização de tarefas que, até o momento, o seres humanos fazem melhor.

“Inteligência Artificial envolve utilizar métodos baseados no comportamento inteligente de humanos e outros animais para solucionar problemas complexos”. (COPPIN, 2010, p. 4).

Segundo Russel e Norvig (2013), há sistemas que pensam ou agem como seres humanos e sistemas que pensam ou agem racionalmente. Uma abordagem focada nos seres humanos envolve hipóteses e confirmação experimental, e uma racionalista envolve uma combinação de matemática e engenharia.

Portanto, IA está relacionada com o estudo e a construção de sistemas inteligentes, que podem pensar ou agir de forma humana ou racional, sendo chamados de agentes inteligentes.

#### 3.1 FUNDAMENTOS DE IA

Russel e Norvig (2013, p. 34) afirmam que

Os filósofos (desde 400 a.C.) tornaram a IA concebível, considerando as ideias de que a mente é, em alguns aspectos, semelhante a uma máquina, de que ela opera sobre o conhecimento codificado em alguma linguagem interna e que o pensamento pode ser usado para escolher as ações que deverão ser executadas.

Os matemáticos forneceram as ferramentas para manipular declarações de certeza lógica, bem como declarações incertas e probabilísticas. Eles também definiram a base para a compreensão da computação e do raciocínio sobre algoritmos.

Também existiram contribuições de outras pessoas para a IA, como os economistas, que formalizaram o problema de tomar decisões que maximizam o

resultado esperado. Os neurocientistas descobriram sobre como a mente trabalha e a forma como ela se assemelha e se diferencia dos computadores. Os psicólogos adotaram a ideia de que os seres humanos e os animais podem ser considerados máquinas de processamento de informações. Os linguistas mostraram que o uso da linguagem se ajusta a esse modelo. Os engenheiros de computação forneceram máquinas cada vez mais poderosas que tornam possíveis as aplicações de IA. Também surgiu a teoria de controle, que trabalha com a criação de dispositivos que agem de forma ótima com base na resposta do ambiente. (RUSSEL; NORVIG, 2013).

## 3.2 HISTÓRIA DA IA

Com o tempo, a Inteligência Artificial pôde se desenvolver e avançar muito, principalmente nas últimas décadas, tornando-se uma das áreas de maior importância para a computação e a sociedade atualmente. Nas seções seguintes há um resumo de sua história.

### 3.2.1 A Geração da Inteligência Artificial (1943-1955)

Pode-se dizer que o primeiro grande trabalho reconhecido como IA foi realizado por Warren McCulloch e Walter Pitts (1943). (RUSSELL; NORVIG, 2013).

Russel e Norvig (2013, p.20) afirmam que

Eles se basearam em três fontes: o conhecimento da fisiologia básica e da função dos neurônios no cérebro; uma análise formal da lógica proposicional criada por Russell e Whitehead; e a teoria da computação de Turing. Esses dois pesquisadores propuseram um modelo de neurônios artificiais, no qual cada neurônio se caracteriza por estar “ligado” ou “desligado”, com a troca para “ligado” ocorrendo em resposta à estimulação por um número suficiente de neurônios vizinhos.

Contudo, foi Alan Turing o primeiro a articular uma visão completa da IA em seu artigo de 1950 “Computing Machinery and Intelligency”. Ele apresentou o Teste de Turing, onde sugeriu um teste baseado na impossibilidade de distinguir entre entidades inegavelmente inteligentes, “os seres humanos”. (RUSSELL; NORVIG, 2013).

### 3.2.2 O nascimento da IA (1956)

John McCarthy, Hyman Minsky, Claude Shannon e Nathaniel Rochester foram os principais idealizadores da época. Eles organizaram um seminário de dois meses em Dartmouth, em 1956, onde havia mais seis participantes: Trenchard More (Princeton), Arthur Samuel (IBM), Allen Newell e Herbert Simon da Carnegie Mellon University (CMU), Ray Solomonoff e Oliver Selfridge do Massachusetts Institute of Technology (MIT). Os destaques desse encontro foram: Allen Newell e Herbert Simon, com o programa de raciocínio Logic Theorist (LT). (RUSSELL; NORVIG, 2013).

Russel e Norvig (2013, p. 21) afirmam que

O seminário de Dartmouth não trouxe nenhuma novidade, mas apresentou uns aos outros todos os personagens importantes da história. Nos 20 anos seguintes, o campo seria dominado por essas pessoas e por seus alunos e colegas do MIT, da CMU, de Stanford e da IBM.

### 3.2.3 Entusiasmo inicial, grandes expectativas (1952-1969)

Neste período houve entusiasmo e grandes expectativas, mas poucos progressos. Newell e Simon desenvolveram o General Problem Solver (GPS) ou solucionador de problemas gerais, que imitava o comportamento humano na resolução de problemas. (RUSSELL; NORVIG, 2013).

Alguns dos primeiros programas de IA foram produzidos pela IBM nesta época. A partir de 1952, Arthur Samuel desenvolveu programas para jogos de damas que aprendiam a jogar em um nível amador elevado. Em 1958, McCarthy definiu a linguagem de alto nível Lisp, que se tornou a linguagem de programação dominante na IA. (RUSSELL; NORVIG, 2013).

Nesta época, Marvin Minsky orientou alunos do MIT na criação dos domínios limitados conhecidos como micromundos, que utilizavam a IA para a resolução de problemas limitados como cálculo integral, analogia geométrica e álgebra. (RUSSELL; NORVIG, 2013).

### 3.2.4 Uma dose de realidade (1966-1973)

Nesta época surgiram dificuldades em relação aos programas desenvolvidos quando eram aplicados em problemas mais difíceis ou complexos.

Russel e Norvig (2013, p. 25) afirmam que

O primeiro tipo de dificuldade surgiu porque a maioria dos primeiros programas não tinha conhecimento de seu assunto; eles obtinham sucesso por meio de manipulações sintáticas simples.

O segundo tipo de dificuldade foi a impossibilidade de tratar muitos dos problemas que a IA estava tentando resolver. A maior parte dos primeiros programas de IA resolvia problemas experimentando diferentes combinações de passos até encontrar a solução. Essa estratégia funcionou inicialmente porque os micromundos continham pouquíssimos objetos e, conseqüentemente, um número muito pequeno de ações possíveis e sequências de soluções muito curtas.

Uma terceira dificuldade surgiu devido a algumas limitações fundamentais nas estruturas básicas que estavam sendo utilizadas para gerar o comportamento inteligente.

### 3.2.5 Sistemas Baseados em Conhecimento (1969-1979)

Em 1969, a Universidade de Stanford criou o programa DENDRAL, para desenvolver soluções capazes de encontrar as estruturas moleculares orgânicas a partir da espectrometria de massa das ligações químicas presentes em uma molécula desconhecida. Na época, personagens importantes como Edward Feigenbaum (antigo aluno de Herbert Simon), Bruce Buchanan (filósofo transformado em cientista de computação) e Joshua Lederberg (geneticista premiado com um prêmio Nobel) constituíram equipe para resolver o problema e o DENDRAL foi capaz de solucionar graças ao seu modo automático de tomar decisões. (RUSSELL; NORVIG, 2013).

O DENDRAL teve sua importância para o desenvolvimento de programas inteligentes, porque representou o primeiro sistema bem-sucedido de conhecimento intensivo: sua habilidade derivava de um grande número de regras de propósito específico. (RUSSELL; NORVIG, 2013).

### 3.2.6 A IA se Torna uma Indústria (De 1980 Até a Atualidade)

O primeiro sistema especialista comercial bem-sucedido, o R1, iniciou sua operação na Digital Equipment Corporation (DEC). O programa contribuiu para configurar pedidos de novos sistemas de computador; em 1986, ele já fazia a empresa faturar cerca de 40 milhões de dólares por ano. Em 1988, o grupo de IA da DEC já possuía 40 sistemas especialistas entregues, com outros sendo produzidos. A Du Pont tinha 100 desses sistemas em uso e 500 em desenvolvimento, economizando aproximadamente 10 milhões de dólares por ano. Quase todos os conglomerados importantes dos Estados Unidos possuíam seu próprio grupo de IA e estavam usando ou investigando sistemas especialistas. (RUSSELL; NORVIG, 2013).

Em 1981, os japoneses anunciaram o projeto Fifth Generation, um plano de 10 anos para montar computadores inteligentes por meio da utilização do Prolog. Em Resposta, os Estados Unidos constituíram a Microelectronics and Computer Consortium (MCC) como um consórcio de pesquisa projetado para assegurar a competitividade nacional. Em ambos os casos, a IA fazia parte de um amplo esforço, incluindo o projeto de chips e a pesquisa da interface humana. (RUSSELL; NORVIG, 2013).

## 3.3 AGENTES INTELIGENTES

Russel e Norvig (2013) consideram que a Inteligência Artificial está relacionada principalmente a uma ação racional. No caso ideal, um agente inteligente adota a melhor ação possível em uma situação.

“Um agente é uma entidade capaz de realizar alguma tarefa, geralmente para auxiliar um usuário humano.” (COPPIN, 2010, p. 470).

“Um agente é tudo o que pode ser considerado capaz de perceber seu ambiente por meio de sensores e de agir sobre esse ambiente por intermédio de atuadores.” (RUSSEL; NORVIG, 2013, p. 41).

O agente pode perceber o ambiente através de câmeras, microfones, teclados ou mouses, e agir sobre o mesmo por meio de vídeos, autofalantes, impressoras, garras de robôs, entre outros dispositivos.

Para a construção de um agente, é necessário definir as percepções, ações, metas e ambiente do mesmo. Ele pode ter conhecimentos variados como as propriedades do mundo, os estados desejáveis, consequências de suas ações, avaliação de seus próprios conhecimentos, entre outros.

Quando o agente tem autonomia, pode interagir com o mundo e tirar informações sobre o mesmo. Através de seu conhecimento inicial e experiências, consegue adquirir novos conhecimentos. Também existe o conceito de onisciência, que está relacionado com agentes que conhecem os efeitos de suas ações e agem de acordo com eles. (RUSSELL; NORVIG, 2013).

Existem medidas de desempenho que avaliam o sucesso de um agente na realização de uma tarefa. Quando ele faz alguma ação no ambiente, por meio da percepção, acaba gerando uma sequência de estados. Se forem desejáveis, significa que teve um bom desempenho. (RUSSELL; NORVIG, 2013).

“[...] um agente racional deve selecionar uma ação que se espera venha a maximizar sua medida de desempenho, dada a evidência fornecida pela sequência de percepções e por qualquer conhecimento interno do agente.” (RUSSELL; NORVIG, 2013, p. 45).

### 3.3.1 Ambientes

Para a criação de agentes, é necessário conhecer os ambientes de interação, já que possuem grande influência sobre os mesmos. São divididos em algumas categorias:

- a) Completamente observável (vs. parcialmente observável): “Se os sensores de um agente permitem acesso ao estado completo do ambiente em cada instante, dizemos que o ambiente de tarefa é completamente observável;” (RUSSELL; NORVIG, 2013, p. 50).
- b) Agente único (vs. multiagente): “[...] um agente que resolve um jogo de palavras cruzadas sozinho está claramente em um ambiente de agente único, enquanto um agente que joga xadrez está em um ambiente de dois agentes;” (RUSSELL; NORVIG, 2013, p. 50).
- c) Determinístico (vs. estocástico): “Se o próximo estado do ambiente é completamente determinado pelo estado atual e pela ação executada

pelo agente, dizemos que o ambiente é determinístico;” (RUSSEL; NORVIG, 2013, p. 51).

- d) Episódico (vs. sequencial): “Em cada episódio, o agente recebe uma percepção e em seguida executa uma única ação. É crucial que o episódio seguinte não dependa das ações executadas em episódios anteriores;” (RUSSEL; NORVIG, 2013, p. 51).
- e) Estático (vs. dinâmico): “Se o ambiente puder se alterar enquanto um agente está deliberando, dizemos que o ambiente é dinâmico para esse agente; caso contrário, ele é estático;” (RUSSEL; NORVIG, 2013, p. 51).
- f) Discreto (vs. contínuo): “A distinção entre discreto e contínuo aplica-se ao estado do ambiente, ao modo como o tempo é tratado, e ainda às percepções e ações do agente;” (RUSSEL; NORVIG, 2013, p. 52).
- g) Conhecido (vs. desconhecido): “Estritamente falando, essa distinção não se refere ao ambiente em si, mas ao estado de conhecimento do agente (ou do projetista) sobre as “leis da física” no meio ambiente.” (RUSSEL; NORVIG, 2013, p. 52).

### 3.4 APLICAÇÕES DE IA

Atualmente existem muitas aplicações de IA para a sociedade, estando relacionadas com tarefas do dia a dia, tarefas formais e tarefas especialistas. (ROSA, 2011).

Nas tarefas do dia a dia, a IA pode ser aplicada em percepção (visão e fala) e línguas naturais (entendimento, geração e tradução). Nas tarefas formais, a IA trabalha com jogos (xadrez, jogos eletrônicos, etc.) e matemática (geometria, lógica e cálculo integral). Também é responsável por tarefas especialistas, como análise científica, diagnóstico médico, análise financeira e engenharia (projetos, descoberta de falhas e planejamento de manufatura). (ROSA, 2011).

Segundo Coppin (2010), os agentes inteligentes têm muitas aplicações, sendo utilizados para ajudar as pessoas a resolver problemas enquanto usam seus computadores. Também podem percorrer a Internet, encontrando documentos interessantes para as pessoas. Os robôs, que são a materialização física dos agentes, também são aplicados em diversas áreas, como a exploração de oceanos

e outros mundos. São capazes de viajar para ambientes em que o ser humano não consegue viver.

Coppin (2010, p. 21) afirma que

A Inteligência Artificial é especialmente útil nas situações em que métodos tradicionais seriam muito lentos. Problemas combinatórios, tais como alocar professores e alunos em salas de aulas, não são bem resolvidos por técnicas tradicionais da ciência da computação. Nestes casos as heurísticas e técnicas fornecidas pela Inteligência Artificial podem obter excelentes soluções.

### 3.5 PROCESSAMENTO DE LINGUAGEM NATURAL

No contexto de IA, existe a área de Processamento de Linguagens Naturais (PLN), que procura criar agentes inteligentes capazes de processar linguagens naturais, a fim de se comunicarem com seres humanos e adquirir informação a partir da linguagem escrita. (RUSSEL; NORVIG, 2013).

“Há mais de um trilhão de páginas de informações na Web [...]. Um agente que deseja adquirir conhecimento precisa entender (pelo menos parcialmente) a ambígua e confusa linguagem que os seres humanos usam.” (RUSSEL; NORVIG, 2013, p. 970).

Dias da Silva et al. (1999, p. 6) afirmam que

Uma outra possibilidade, cuja realização é sem dúvida muito mais complexa, continua sendo um desafio: criar programas capazes de interpretar mensagens codificadas em línguas naturais. Por que não investigar meios que façam com que as máquinas “aprendam” as línguas naturais e sejam capazes de decifrá-las?  
[...] Desde então, criar programas computacionais “inteligentes”, até mesmo capazes de “compreender” as línguas e, por meio delas, simular uma interação verbal com o usuário, tem se revelado um empreendimento polêmico, complexo e desafiador, porém, fascinante.

De acordo com Gonzalez e Lima (2003), PLN trabalha com os aspectos da comunicação humana, como sons, palavras, sentenças e discursos, considerando formatos e referências, estruturas e significados, contextos e usos.

Todas as definições de PLN possuem duas características: armazenamento em computador e manipulação de dados linguísticos. Portanto, a área de PLN pode

ser definida como a capacidade de um computador em processar a linguagem humana utilizada no dia a dia. (ROSA, 2011).

### 3.5.1 Teste de Turing

Foi criado por Alan Turing (1950), com o objetivo de avaliar uma IA em relação à inteligência humana, tendo importância até os dias atuais. São feitas perguntas a um computador e ele só é aprovado no teste caso o interrogador humano não consiga descobrir se as respostas são de uma pessoa ou não. É um teste que permanece importante até os dias de hoje e exige que o computador tenha as seguintes capacidades:

- a) “Processamento de linguagem natural para permitir que ele se comunique com sucesso em um idioma natural;” (RUSSEL; NORVIG, 2013, p. 4).
- b) “Representação de conhecimento para armazenar o que sabe ou ouve;” (RUSSEL; NORVIG, 2013, p. 4).
- c) “Raciocínio automatizado para usar as informações armazenadas com a finalidade de responder a perguntas e tirar novas conclusões;” (RUSSEL; NORVIG, 2013, p. 4).
- d) “Aprendizado de máquina para se adaptar a novas circunstâncias e para detectar e extrapolar padrões.” (RUSSEL; NORVIG, 2013, p. 4).

Também existe o teste de Turing total, que inclui vídeo, permitindo que o interrogador possa testar a percepção da IA, além de dar a oportunidade de entregar objetos físicos. Para ser aprovado neste teste, o computador precisa de:

- a) “Visão computacional para perceber objetos;” (RUSSEL; NORVIG, 2013, p. 5).
- b) “Robótica para manipular objetos e movimentar-se.” (RUSSEL; NORVIG, 2013, p. 5).

Russel e Norvig (2013, p. 5) afirmam que

[...] Turing merece crédito por projetar um teste que permanece relevante depois de 60 anos. Ainda assim, os pesquisadores da IA têm dedicado pouco esforço à aprovação no teste de Turing, acreditando que seja mais importante estudar os princípios básicos da inteligência do que reproduzir um exemplar. O desafio do “voo artificial” teve sucesso quando os irmãos Wright e outros

pesquisadores pararam de imitar os pássaros e começaram a usar túneis de vento e aprender sobre aerodinâmica. Os textos de engenharia aeronáutica não definem como objetivo de seu campo criar “máquinas que voem exatamente como pombos a ponto de poderem enganar até mesmo outros pombos”.

### 3.5.2 Evolução dos Sistemas de PLN

Segundo Dias da Silva et al. (1999), a evolução dos Sistemas de Processamento de Linguagem Natural (SPLN) começou na década de 50 com o desenvolvimento da Tradução Automática, focando em sistematização computacional das classes de palavras da gramática tradicional e identificação computacional de poucos tipos de constituintes operacionais.

Na década de 60 há o surgimento de novas aplicações e a criação de formalismos. Nesta época os destaques são os primeiros tratamentos computacionais das gramáticas livres de contexto, a criação dos primeiros analisadores sintáticos e as primeiras formalizações do significado em termos de redes semânticas. (DIAS DA SILVA et al., 1999).

A consolidação dos estudos de PLN aconteceu na década de 70, com a implementação de parcelas das primeiras gramáticas e analisadores sintáticos, e a busca de formalização de fatores pragmáticos e discursivos. (DIAS DA SILVA et al., 1999).

A sofisticação dos Sistemas ocorreu na década de 80, com o desenvolvimento de teorias linguísticas motivadas pelos estudos de PLN. Na década de 90 surgem os Sistemas baseados em “representações do conhecimento”, com a criação de projetos complexos que buscam a integração dos vários tipos de conhecimentos linguísticos e extralinguísticos, e das estratégias de inferência envolvidas nos processos de produção, manipulação e interpretação de objetos linguísticos. (DIAS DA SILVA et al., 1999).

Com o tempo, as técnicas de PLN avançaram muito e podem ter muitas aplicações atualmente. Há muitos agentes inteligentes que utilizam PLN, focando na área empresarial, social, educacional, tradução, entre outras.

### 3.5.3 Aplicações de PLN

De acordo com Dias da Silva et al. (1999), há muitas possibilidades na construção de SPLN, com graus diferentes de sofisticação linguística. Os tipos de aplicação são abordados abaixo:

- a) Manipulação de bases de dados: Esses sistemas realizam a comunicação entre o usuário e a base de dados, “traduzindo” instruções codificadas em frases, digitadas em um terminal, para a linguagem do sistema de gerenciamento de dados, que é responsável por manipular as informações. Esses SPLNs podem ser definidos como “sistemas de perguntas e respostas”.
- b) Sistemas tutores: Existem dois tipos de sistemas de estudo por computador: os tradicionais e os inteligentes. Nos tradicionais, os conteúdos são estruturados e mostrados no monitor em forma de instrução programada e ramificada, definidas pelo projetista do sistema. Por outro lado, nos sistemas inteligentes os conteúdos são estruturados como “redes de conhecimentos”, compostas de fatos, regras e relações que permitem ao sistema simular a situação em que aluno e professor conversam sobre assuntos específicos.
- c) Sistemas de automação de tarefas administrativas: Esses sistemas ajudam nas tarefas de rotina de setores administrativos e gerenciais de empresas e instituições.
- d) Programação automática: Esses sistemas são desenvolvidos a fim de facilitar a interação entre o programador e a máquina, tendo uma estrutura complexa.
- e) Sistemas de processamento de textos científicos: Esse tipo de sistema converte textos científicos para uma base de dados, permitindo que o usuário obtenha informações sobre os mesmos através de perguntas.
- f) Sistemas especializados: Esse tipo de sistema utiliza parcelas do conhecimento humano para resolver problemas, tendo inúmeras aplicações.
- g) Tradução automática: Esses sistemas têm a finalidade de traduzir uma língua para outra, possuindo vários níveis de sofisticação.

### 3.5.4 Análise Léxico-morfológica

Segundo Dias da Silva et al. (1999), as palavras da língua também podem ser segmentadas em termos do seu conteúdo significativo. A morfologia trabalha com morfemas, que são as unidades mínimas compostas de significado, sendo divididos em dois tipos: gramaticais – quando se trata de definir os marcadores da flexão das palavras –, e lexicais – quando alguns elementos são associados a uma base na formação de novas palavras da língua.

“O analisador léxico-morfológico trabalha com a estrutura das palavras e classifica as mesmas em diferentes categorias.” (GONZALEZ; LIMA, 2003).

Segundo Rosa (2011), na análise léxico-morfológica as palavras são decompostas e classificadas em categorias lexicais de acordo com um dicionário. Por exemplo, o termo *documentos* seria quebrado em *documento* + (plural) *s*. Depois deve ser atribuída a categoria lexical *substantivo* ao termo *documento*.

Müller (2003, p. 3) constata que

Este tipo de análise é necessário para que o tamanho do dicionário não fique muito extenso, uma vez que é mais simples o armazenamento do radical da palavra e seus afixos. Estes são os componentes que formam uma palavra juntamente com o radical, como prefixos e sufixos. Um exemplo de prefixo é *des* na palavra *desesperança* e de sufixo é *mos* na palavra *calamos*.

O tratamento computacional deste tipo de análise é relativamente simples. Baseia-se em regras que analisam as palavras e as classificam segundo tabelas de afixos. Por exemplo, a entrada *zinho* de uma tabela de sufixos está associada a um diminutivo de um substantivo, portanto, a palavra *bonezinho* é o diminutivo da palavra *boné*, que é seu radical. Desta forma, são reconhecidas as palavras que não estão na sua forma padrão, já adequando-as para a fase posterior de análise sintática.

### 3.5.5 Análise Sintática

O analisador sintático trabalha com agrupamento de palavras, analisando a constituição das frases. A análise sintática é o procedimento que avalia as formas de combinação das regras gramaticais, com a finalidade de gerar uma estrutura de árvore que represente a estrutura sintática da sentença analisada. Se a sentença for ambígua, o analisador sintático irá obter todas as possíveis estruturas sintáticas que a representam. (GONZALEZ; LIMA, 2003).

Rosa (2011) afirma que o analisador sintático atribui uma estrutura gramatical a uma sentença. Ele é responsável por pegar as palavras individualmente e decidir como se ligam para formar sentenças.

As palavras possuem combinações coerentes com a sintaxe definida pela linguagem. “[...] Não é coerente, por exemplo, a colocação de um adjetivo antes de um artigo. A verificação disto é tarefa da análise sintática.” (MÜLLER, 2003, p.3).

Dias da Silva et al. (1999, p. 28) afirmam que

A organização das palavras na sentença acarreta a definição desses itens lexicais em termos de suas funções gramaticais. Trata-se de reconhecer as regras pelas quais a distribuição das formas são determinadas e esse exercício é o objeto de estudo da Sintaxe. Quando as palavras são combinadas entre si para formar um enunciado dotado de um sentido completo, sua distribuição na sentença não ocorre de maneira aleatória, mas, ao contrário, essa disposição segue regras estruturais bastante definidas. Essas regras determinam, por exemplo, o emprego dos pronomes, a aplicação da crase, a realização da concordância. Na manipulação dessas regras, faz-se uso de um conjunto de categorias definido em termos da sua função sintática, das quais são exemplos as categorias sujeito, objeto direto, complemento nominal, adjunto adverbial e assim por diante.

### **3.5.6 Análise Semântica**

A semântica investiga o significado presente nas palavras, para que tenham sentido numa sentença. Da mesma forma, há o significado de expressões, de sentenças, enfim, de unidades mais complexas da língua. (DIAS DA SILVA et al., 1999).

Os maiores desafios encontrados na análise semântica estão relacionados às ambiguidades do tipo polissemia (palavras com muitos significados, ex.: “cabo”) e homonímia (palavras com mesma grafia, mas significados diferentes, ex.: “ponto”). (DIAS DA SILVA et al., 1999).

De acordo com Rosa (2011), na análise semântica a sintaxe é traduzida para alguma forma lógica que representa o significado da sentença, permitindo a realização de inferências. Como exemplo, o autor afirma que a questão “Os documentos estão no sistema?” poderia ser respondida, através de uma representação semântica adequada. O sistema poderia inferir que a recuperação de documentos significa que havia documentos a serem recuperados.

### 3.5.7 Análise Pragmática

Segundo Rosa (2011), a análise pragmática investiga a sentença no contexto, considerando um certo conhecimento sobre o domínio e sobre as intenções dos envolvidos na conversação.

De acordo com Müller (2003), esta análise busca nas demais sentenças a compreensão do contexto que falta à sentença que está sendo analisada, fugindo à estrutura de apenas uma frase.

A aplicação desse conhecimento em PLN é muito importante, especialmente quando o foco do processamento são as referências anafóricas, que retomam sentidos já desenvolvidos no decorrer da conversação (ex.: ele, isso, etc.) e as dêiticas, que retomam as circunstâncias (ex.: hoje, daqui a pouco, lá, etc.). (DIAS DA SILVA et al., 1999).

## 3.6 APRENDIZADO DE MÁQUINA

De acordo com Monard e Baranauskas (2003), Aprendizado de Máquina (AM) é uma área da IA que busca desenvolver técnicas computacionais sobre o aprendizado e construir sistemas capazes de adquirir conhecimento de forma automática. Um sistema de aprendizado toma decisões baseado em experiências anteriores, procurando a melhor opção.

Para que os computadores aprendam com experiências passadas, é utilizado um princípio de inferência denominado indução, no qual são geradas conclusões genéricas a partir de um conjunto de exemplos. Os algoritmos de AM aprendem a induzir uma função ou hipótese capaz de resolver um problema a partir de dados relacionados com o problema. (FACELI et al., 2011).

Monard e Baranauskas (2003) afirmam que

A indução é a forma de inferência lógica que permite obter conclusões genéricas sobre um conjunto particular de exemplos. Ela é caracterizada como o raciocínio que se origina em um conceito específico e o generaliza, ou seja, da parte para o todo. Na indução, um conceito é aprendido efetuando-se inferência indutiva sobre os exemplos apresentados. Portanto, as hipóteses geradas através da inferência indutiva podem ou não preservar a verdade. Mesmo assim, a inferência indutiva é um dos principais métodos utilizados para derivar conhecimento novo e prever eventos futuros.

É importante que número de exemplos seja suficiente e que estejam bem definidos, para que se obtenham hipóteses úteis para um determinado tipo de problema. O aprendizado indutivo é efetuado a partir de raciocínio sobre exemplos fornecidos por um processo externo ao sistema de aprendizado. (MONARD; BARANAUSKAS, 2003).

O aprendizado indutivo pode ser dividido em supervisionado e não supervisionado. No aprendizado supervisionado o algoritmo de aprendizado, ou indutor, classifica novos exemplos em classes predefinidas (resolução de problemas preditivos). Já no aprendizado não supervisionado o indutor é responsável pelo agrupamento de exemplos semelhantes (resolução de problemas descritivos). (MONARD; BARANAUSKAS, 2003).

Segundo Monard e Baranauskas (2003), há diversos paradigmas em AM, listados abaixo:

- a) Simbólico: representações simbólicas de um problema através da análise de exemplos e contraexemplos, estando na forma de expressão lógica, árvore de decisão, regras ou rede semântica;
- b) Estatístico: utiliza modelos estatísticos para encontrar uma aproximação do conceito induzido;
- c) Baseado em Exemplos: classifica um novo exemplo com base em uma classificação similar conhecida;
- d) Conexionista: inspirado no modelo biológico do sistema nervoso (Redes Neurais);
- e) Genético: modelo biológico de aprendizado, onde há uma população de elementos de classificação que competem para fazer a predição (analogia com a teoria de Darwin).

De acordo com Faceli et al. (2011), há várias aplicações de técnicas de AM na solução de problemas reais, listadas a seguir:

- a) Reconhecimento de palavras faladas;
- b) Predição de taxas de cura de pacientes com diferentes doenças;
- c) Detecção do uso fraudulento de cartões de crédito;
- d) Condução de automóveis de forma autônoma em rodovias;
- e) Ferramentas que jogam gamão e xadrez de forma semelhante a campeões;

- f) Diagnóstico de câncer por meio da análise de dados de expressão gênica.

Faceli et al. (2011, p. 3) afirmam que

Além do grande volume de aplicações que se beneficiam das características da área de AM, outros fatores têm favorecido a expansão dessa área, como o desenvolvimento de algoritmos cada vez mais eficazes e eficientes e a elevada capacidade dos recursos computacionais atualmente disponíveis.

Outras motivações para as pesquisas em AM incluem a possibilidade de aumentar a compreensão de como se dá o aprendizado nos seres vivos. Além disso, algumas tarefas são naturalmente mais bem definidas por meio de exemplos. Os modelos gerados são ainda capazes de lidar com situações não apresentadas durante seu desenvolvimento, sem necessariamente necessitar de uma nova fase de projeto.

### 3.7 CHATBOTS

De acordo com Laven (2003), um *chatbot* (ou *chatterbot*) é um programa que tem como objetivo conversar de forma escrita com uma pessoa, tentando fazê-la acreditar que está conversando com outra pessoa.

Os *chatbots* podem abordar diversos assuntos, entendendo frases e perguntas de usuários e respondendo de forma adequada, como se fosse uma pessoa. (SIQUEIRA, 2005).

Ferreira e Uchôa (2006) afirmam que

*Chatbot* ou *chatterbot* é um programa de computador criado com o propósito de simular a habilidade de conversação de um ser humano. Apesar de ser um propósito simples de definir, a implementação de um programa de computador capaz de atingir esse propósito é algo extremamente complexo. Como até hoje não foi possível duplicar a mente humana, capaz de raciocínio e criatividade, os programas de computador que tentam simulá-la geralmente usam recursos da psicologia para “fingir” que são seres humanos.

Através de PLN e uma base de conhecimento sobre assuntos do mundo real, os *chatbots* atuais podem ser utilizados para entretenimento, suporte online, portais corporativos, jogos, projetos educacionais, culturais, treinamentos, call centers e auxílio no ensino à distância. A aplicação depende apenas do conteúdo criado por

redatores e programadores que ensinam o personagem formando a base de conhecimento. (SIQUEIRA, 2005).

Há muitas vantagens na utilização de *chatbots* empresariais, como a diminuição do trabalho de atendentes reais, permitindo que possam realizar outras tarefas mais produtivas na empresa. Também fica mais fácil para visitantes encontrarem informações importantes, bastando perguntar, ao invés de procurar por todo o site sobre o assunto desejado. A informação se torna mais acessível para o usuário, gerando um conforto e estimulando a busca por novos assuntos.

De acordo com Silva (2002), os *chatbots* podem assumir diversos usos comerciais, sendo de vários tipos:

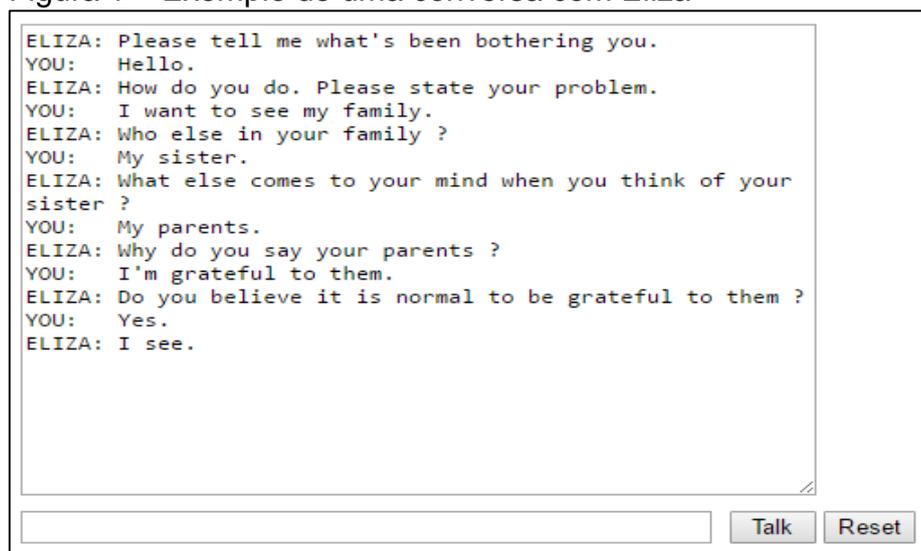
- a) Entretenimento: o objetivo é divertir o usuário, simulando o comportamento humano;
- b) FAQs (Frequently Asked Questions): respondem as perguntas dos usuários de acordo com suas bases de FAQs. Eles podem ser usados para informar sobre empresas ou produtos, ou responder dúvidas de alunos em ambientes virtuais de ensino;
- c) Suporte ao Consumidor: é parecido com o tipo FAQs, sendo que este tipo de *chatbot* precisa interagir de uma melhor forma com o usuário, para descobrir o problema e assim fornecer uma solução que agrade ao cliente. Para isto, é necessário que ele faça algumas perguntas ao usuário, para que baseado nas respostas dadas, consiga gerar a melhor solução, e assim resolver os problemas do mesmo;
- d) Marketing: responsáveis por fazer a propaganda de produtos ou serviços. Eles possuem características tanto dos *chatbots* de entretenimento, quanto dos de FAQs, pois além de conversarem com o usuário tentando demonstrar as vantagens e utilidades dos produtos, eles precisam estar preparados para responderem perguntas sobre dúvidas ou funcionalidades existentes nos produtos;
- e) Propósito Geral: são *chatbots* com um funcionamento muito específico, não podendo ser classificados nas categorias anteriores.

### 3.7.1 Exemplos de Chatbots

Considerando todas as finalidades dos *chatbots*, há muitos exemplos que podem ser citados. A seguir, são abordados os destaques encontrados através da pesquisa realizada.

De acordo com Leonhardt (2005), Eliza foi um dos primeiros *chatbots*, desenvolvido por Joseph Weizenbaum em 1966 no MIT, sendo um dos programas de IA mais antigos e conhecidos. Seu objetivo é simular um psicanalista em uma conversa com seu paciente, fazendo com que ele reflita sobre suas emoções, problemas e convicções. Eliza estimula o paciente a contar sobre seus problemas, por meio de perguntas. A Figura 1 mostra uma conversa com Eliza<sup>1</sup>.

Figura 1 – Exemplo de uma conversa com Eliza



Fonte: Elaborada pelo autor.

Leonhardt (2005, p. 24) afirma que

O algoritmo usado pelos programadores para tentar dar ao diálogo de Eliza uma certa naturalidade e credibilidade é bastante interessante. Sua arquitetura é composta de uma base de regras e um programa. A base de regras é utilizada para processar as sentenças do usuário, enquanto que o programa tem como objetivo ler as sentenças digitadas pelo usuário e processar a base de regras em busca daquela que está associada à sentença digitada. [...]

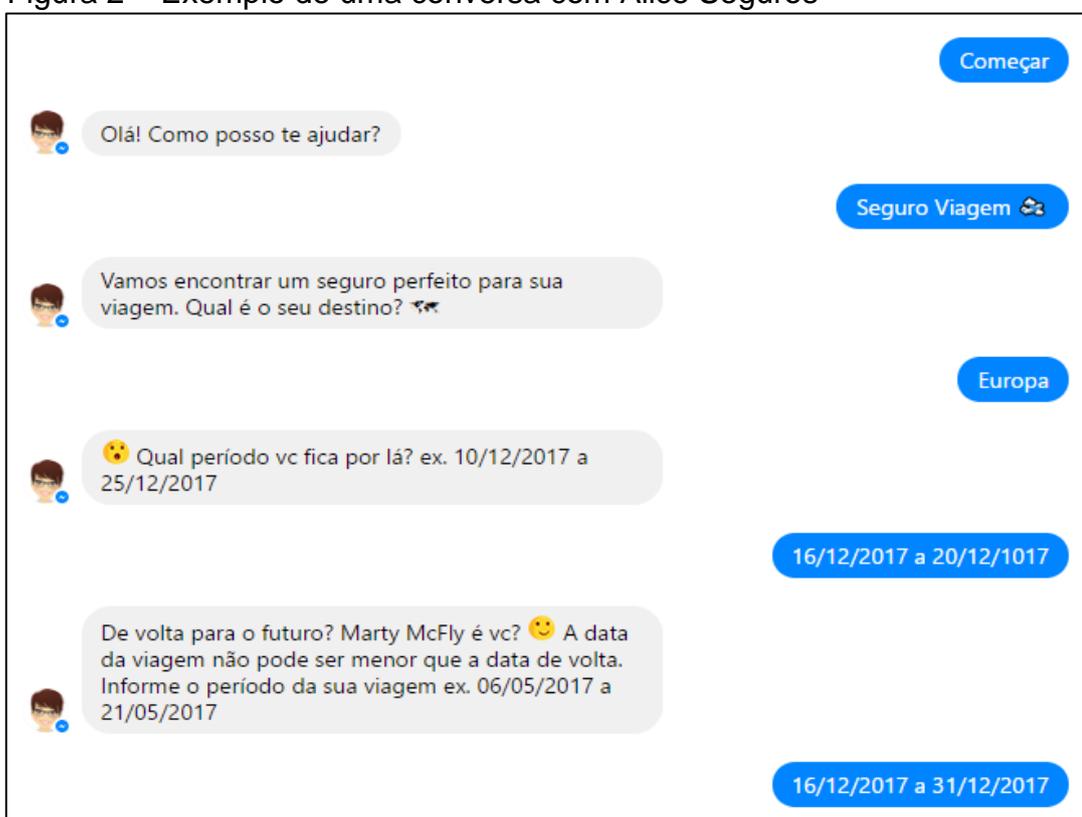
<sup>1</sup> LANDSTEINER, N. Eliza. **Masswerk.at**, 2005. Disponível em: <<http://www.masswerk.at/elizabot/>>. Acesso em: 6 maio 2017.

Embora seja um dos *chatterbots* mais antigos já criados, o princípio de funcionamento interno do software Eliza ainda é utilizado por *chatterbots* modernos. [...]

No Brasil, há muitos *chatbots* atualmente, sendo integrados a plataformas como Messenger, Skype, Slack, Telegram, entre outras. A seguir, são abordados vários projetos nacionais.

Alice Seguros<sup>2</sup> (Figura 2) é um *chatbot* especializado em contratar seguros para os usuários, sendo integrado com o Messenger, Telegram e Slack.

Figura 2 – Exemplo de uma conversa com Alice Seguros



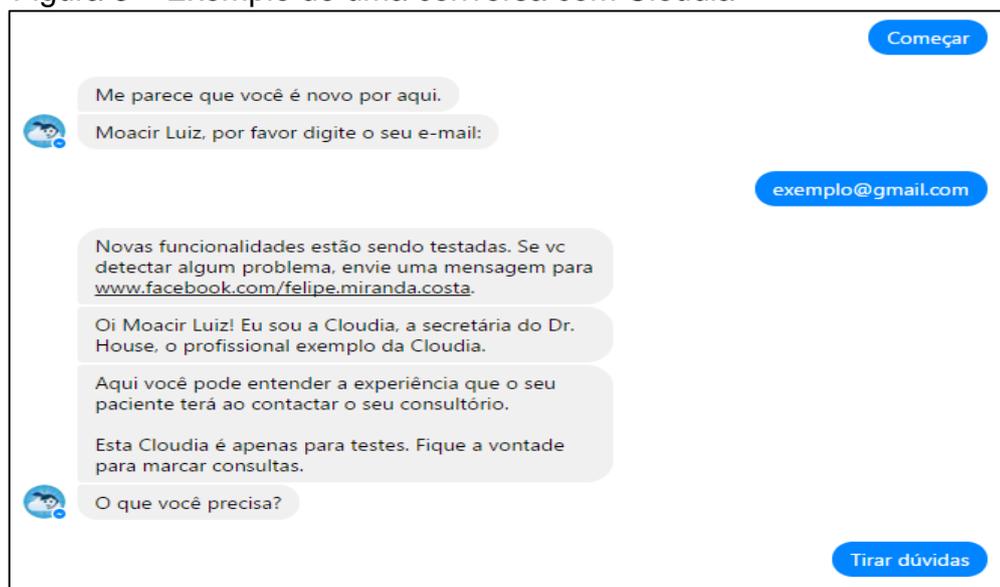
Fonte: Elaborada pelo autor.

Cloudia<sup>3</sup> (Figura 3) é um *chatbot* que ajuda profissionais da saúde a se comunicarem com seus pacientes ou clientes, sendo integrado com o Messenger.

<sup>2</sup> ALICE seguros. **Botsbrasil.com.br**, 2016. Disponível em: <<http://www.botsbrasil.com.br/alice-seguros/>>. Acesso em: 6 maio 2017.

<sup>3</sup> CLOUDIA. **Botsbrasil.com.br**, 2016. Disponível em: <<http://www.botsbrasil.com.br/cloudia/>>. Acesso em: 6 maio 2017.

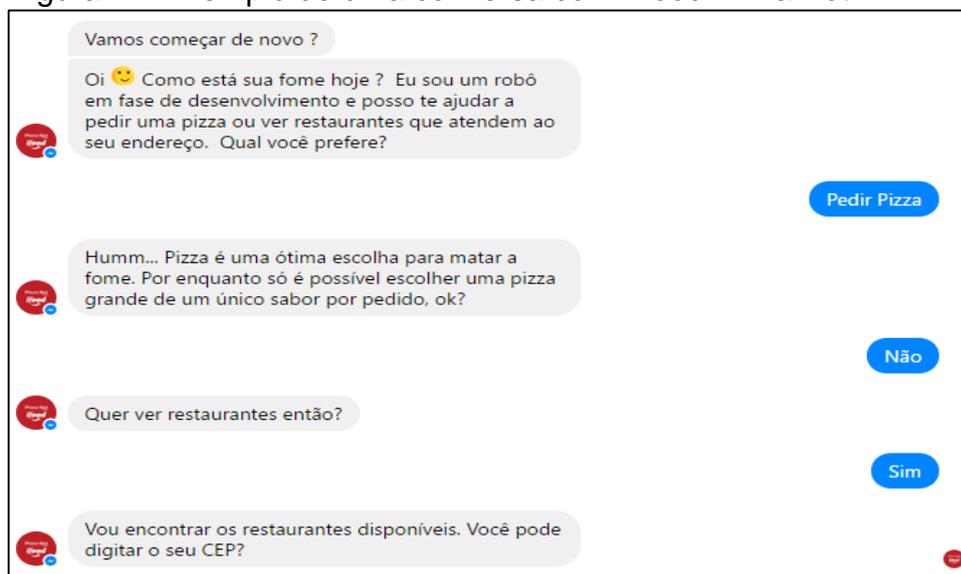
Figura 3 – Exemplo de uma conversa com Cludia



Fonte: Elaborada pelo autor.

IFood Pizza Bot<sup>4</sup> (Figura 4) tem como objetivo vender pizzas e mostrar restaurantes da região, sendo integrado com o Messenger.

Figura 4 – Exemplo de uma conversa com IFood Pizza Bot



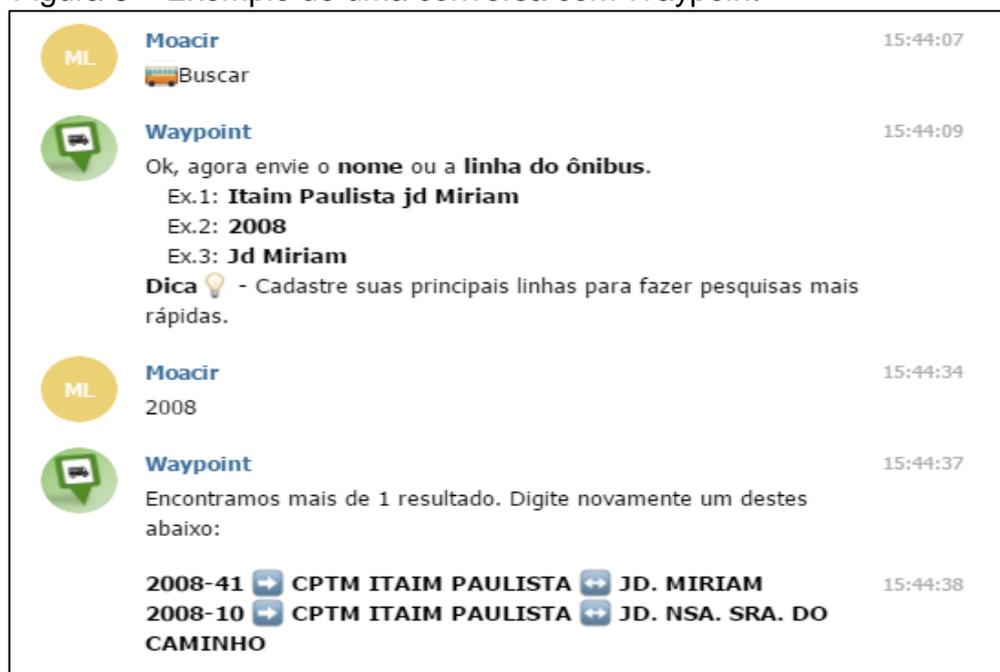
Fonte: Elaborada pelo autor.

Waypoint<sup>5</sup> (Figura 5) é capaz de descobrir a localização de linhas de ônibus e ver como está o metrô, sendo integrado com o Telegram.

<sup>4</sup> IFOOD pizzaria. **Botsbrasil.com.br**, 2016. Disponível em: <<http://www.botsbrasil.com.br/ifood-pizzaria/>>. Acesso em: 6 maio 2017.

<sup>5</sup> WAYPOINT. **Botsbrasil.com.br**, 2016. Disponível em: <<http://www.botsbrasil.com.br/waypoint/>>. Acesso em: 7 maio 2017.

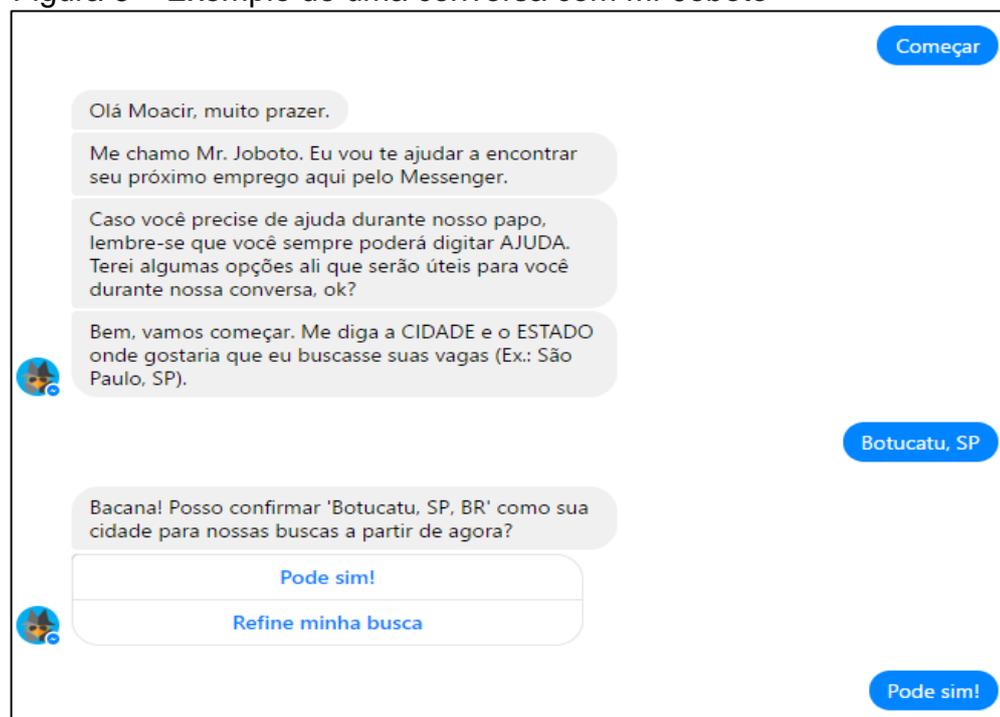
Figura 5 – Exemplo de uma conversa com Waypoint



Fonte: Elaborada pelo autor.

Mr Joboto<sup>6</sup> (Figura 6) é responsável por buscar oportunidades de trabalho, sendo integrado com o Messenger.

Figura 6 – Exemplo de uma conversa com Mr Joboto

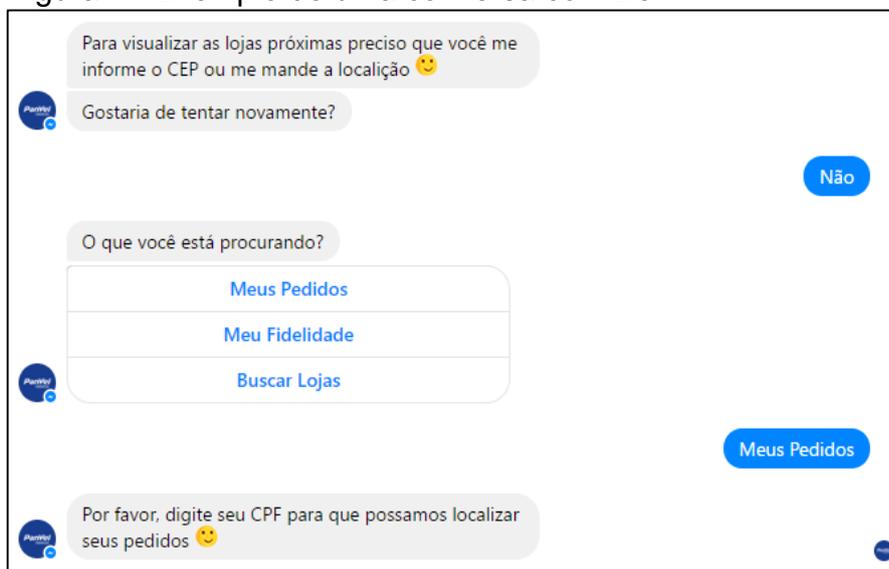


Fonte: Elaborada pelo autor.

<sup>6</sup> MR joboto. **Botsbrasil.com.br**, 2016. Disponível em: <<http://www.botsbrasil.com.br/mr-joboto/>>. Acesso em: 7 maio 2017.

Ben<sup>7</sup> (Figura 7) é um *chatbot* da rede de farmácias Paniel, tendo várias funções, como a consulta de pedidos.

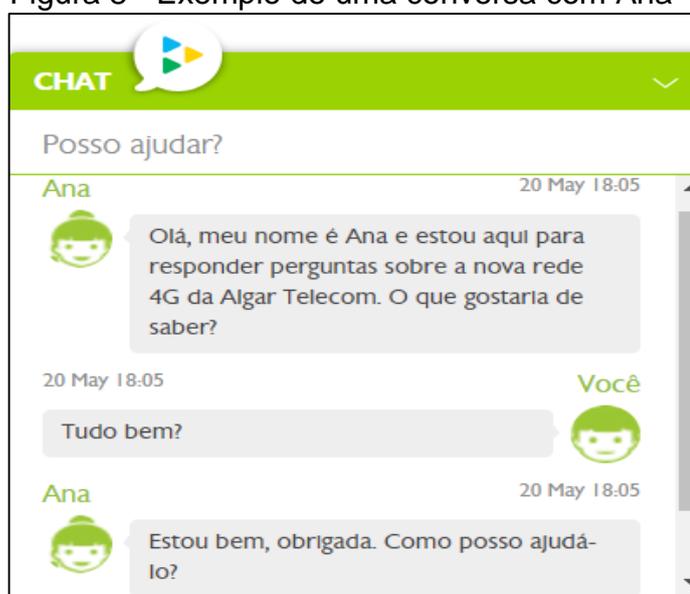
Figura 7 - Exemplo de uma conversa com Ben



Fonte: Elaborada pelo autor.

Ana<sup>8</sup> (Figura 8) é um *chatbot* da empresa de telecomunicações Algar Telecom, sendo capaz de responder perguntas sobre 4G através de um chat.

Figura 8 - Exemplo de uma conversa com Ana



Fonte: Elaborada pelo autor.

<sup>7</sup> BEN. [Botsbrasil.com.br](http://www.botsbrasil.com.br), 2016. Disponível em: <<http://www.botsbrasil.com.br/ben/>>. Acesso em: 7 maio 2017.

<sup>8</sup> ANA. [Algartelem.com.br](https://www.algartelem.com.br), 2017. Disponível em: <<https://www.algartelem.com.br/para-voce/celular/4g>>. Acesso em: 7 maio 2017.

## 4 ENGENHARIA DE SOFTWARE

Durante o desenvolvimento de um software, podem surgir diversos problemas, comprometendo a qualidade do mesmo. Muitas vezes o projeto fica mais caro que o previsto ou não é entregue no prazo. O cliente pode ficar insatisfeito com o resultado, já que não houve uma comunicação correta com o desenvolvedor. Também há muitos casos em que a qualidade do software é duvidosa, já que não são utilizados métodos eficientes para testes. (PRESSMAN, 2011).

Neste contexto, a Engenharia de Software surge como uma área da computação que tem como objetivo garantir a qualidade do projeto desenvolvido, envolvendo um processo, métodos de gerenciamento e desenvolvimento, bem como ferramentas. (PRESSMAN, 2011).

De acordo com Pressman (2011), a Engenharia de Software possui as seguintes camadas:

- a) Foco na qualidade: a base da Engenharia de software é o comprometimento com a qualidade, aprimorando os processos de forma contínua;
- b) Processo: permite o desenvolvimento do software dentro do prazo e de forma racional. Também determina a metodologia usada para a entrega do software, controle do gerenciamento e produção de produtos derivados (documento, relatórios, formulários, etc.);
- c) Métodos: camada responsável por fornecer informações técnicas para o desenvolvimento de software, envolvendo tarefas como levantamento de requisitos, modelagem de projeto, desenvolvimento do software, testes, entre outras;
- d) Ferramentas: são responsáveis por dar suporte para os processos e métodos, melhorando o desempenho dos mesmos.

Durante o desenvolvimento de um software, é importante que exista um planejamento de tudo o que será feito, para que o software seja desenvolvido de acordo com as necessidades do cliente e entregue no prazo previsto.

Neste contexto, deve existir um processo de software, que é uma abordagem adaptável, permitindo à equipe de software selecionar o conjunto adequado de ações e tarefas que serão realizadas. Neste caso, há uma metodologia de processo,

que engloba várias atividades essenciais para um projeto de software. (PRESSMAN, 2011).

Segundo Pressman (2011), há cinco atividades metodológicas que podem ser aplicadas para o desenvolvimento de qualquer software: Comunicação, Planejamento, Modelagem, Construção e Emprego.

Antes de começar a desenvolver um sistema, é extremamente importante conversar com o cliente. Através do levantamento de requisitos, é possível descobrir as necessidades do cliente e determinar as características do software. (PRESSMAN, 2011).

Na etapa de planejamento, é criado um plano de projeto de software, que define as tarefas técnicas, os riscos prováveis, os recursos necessários, os produtos resultantes e um cronograma de trabalho. (PRESSMAN, 2011).

Após isto, é feita a modelagem, através das informações coletadas com o cliente. São criados modelos para entender as necessidades do software e o projeto que será desenvolvido. (PRESSMAN, 2011).

Com isto, é possível iniciar a etapa de construção, que envolve a programação e os testes necessários para verificar erros no código. (PRESSMAN, 2011).

Por fim, o software é entregue ao cliente, que realiza a avaliação e fornece um feedback. (PRESSMAN, 2011).

Gudwin (2015, p. 2) afirma que

Assim como quando construímos uma casa, é necessária uma planta, um projeto, que nos permita visualizar o sistema, antes que ele seja construído. Da mesma forma, é necessária uma linguagem de modelagem padronizada para que qualquer engenheiro, de posse da documentação, possa ter uma ideia de como o programa será construído. É óbvio que é necessária uma metodologia de desenvolvimento para que o esforço seja utilizado de maneira racional, fazendo com que a construção de nosso sistema seja feita de maneira mais eficiente.

#### 4.1 UML

No contexto de Engenharia de Software, existe a linguagem UML (Unified Modeling Language), utilizada para modelar todas as etapas do processo de

desenvolvimento de software, bem como produzir todos os artefatos de software necessários à documentação dessas etapas. (GUDWIN, 2015).

Apesar de ser uma linguagem formal (ou seja, é definida na forma de uma gramática), UML é uma linguagem visual, baseada em diferentes tipos de diagramas. (GUDWIN, 2015).

De acordo com Pressman (2011), os diagramas ajudam os desenvolvedores a construir o software corretamente. Através deles, é possível entender o projeto e explicar para outras pessoas sobre o mesmo.

#### **4.1.1 Diagramas UML**

De acordo com Gudwin (2015), os diagramas representam de forma visual os elementos que fazem parte de um sistema e a relação entre os mesmos. Atualmente, há vários diagramas utilizados, tendo várias finalidades.

Um dos tipos de diagramas úteis para a modelagem são os chamados Diagramas de Atividades. Através deles, é possível demonstrar uma sequência de ações que compõem um processo. Portanto, podem ser usados para descrever o processo de desenvolvimento de software. (GUDWIN, 2015).

Outro tipo é o Diagrama de Caso de Uso, que representa atores, casos de uso e seus relacionamentos. Gudwin (2015, p. 49) afirma que

Casos de uso são abstrações de pequenas histórias narrativas envolvendo a interação entre um ou mais usuários (chamados de atores) e o sistema. A ideia é que estes casos de uso representem, por meio dessas pequenas histórias, as funcionalidades de um sistema. Imagina-se um conjunto de atores necessários a operar o sistema e passa-se a descrever o fluxo dos acontecimentos, onde o ator executa uma ação, e o sistema responde de alguma maneira a essa ação, até que alguma funcionalidade tenha sido contemplada.

#### **4.2 TESTE DE SOFTWARE**

Segundo Pressman (2011), o teste de software é importante para revelar erros cometidos durante a etapa de desenvolvimento, verificando e validando o software. Com isto, é possível garantir que o software realize suas funções corretamente e esteja de acordo com as necessidades do cliente.

Existem muitas estratégias que podem ser utilizadas para testar um software. Há casos extremos em que os testes são executados quando o sistema está completo, para que os erros sejam encontrados. Não é uma estratégia recomendável, já que o sistema ficará com defeitos e não irá agradar o cliente. Também há outra estratégia extrema, onde os testes são executados diariamente, sempre que uma parte do sistema estiver concluída, sendo bastante eficaz. (PRESSMAN, 2011).

Pressman (2011) afirma que a estratégia utilizada pela maioria das equipes de software está entre os dois extremos. Com isto, há uma visão incremental do teste, começando com o teste das unidades individuais do programa, passando para os testes que facilitam a integração de unidades e terminando com testes que usam o sistema concluído.

O teste de unidade é responsável por verificar o componente de software, que é a menor unidade de projeto do software, para que erros sejam encontrados dentro do mesmo. Há um foco na lógica interna de processamento e nas estruturas de dados presentes no componente. (PRESSMAN, 2011).

Após isto, vem o teste de integração, que verifica como os componentes integrados passam a funcionar em conjunto. Podem ser divididos em testes funcionais (ou testes de caixa-preta) e testes estruturais (ou testes de caixa-branca). (GUDWIN, 2015).

Gudwin (2015, p. 143) afirma que

Os testes funcionais visam testar as diferentes funcionalidades especificadas nos casos de uso. Assim, constroem-se diferentes cenários para cada caso de uso, e reproduz-se as ações realizadas pelo usuário em cada cenário, observando-se como o sistema reage, e comparando-se com a reação esperada, conforme a especificação dos casos de uso.

Os testes estruturais utilizam o conhecimento sobre a estrutura do código gerada, para tentar exercitar diferentes trechos de código. Complementando os testes estruturais implementados nos testes de unidade, eles tentam verificar a integração entre múltiplos componentes e exercitar essa integração.

Por fim, há o teste de sistema, que visa testar o sistema como um todo. Deve ser realizado após o teste de integração indicar que o sistema está estabilizado. (GUDWIN, 2015).

De acordo com Gudwin (2015), há vários testes de sistema:

- a) Testes de instalação: verificam se o sistema pode ser instalado na plataforma do cliente e que o mesmo opera corretamente após a instalação. Assim, todas as plataformas do sistema devem ser testadas;
- b) Testes de configuração: verificam se o sistema é capaz de operar corretamente nas diferentes configurações em que possa aparecer. Assim, deve existir um teste específico para cada configuração;
- c) Testes de segurança: geram falhas no sistema para revelar suas fraquezas. Assim, tenta-se usar o sistema de maneiras para as quais ele não tenha sido projetado;
- d) Testes de desempenho: tentam verificar como o sistema se comporta quando os recursos disponíveis são insuficientes. São diferentes dos testes de segurança, pois o objetivo não é fazer o sistema falhar, mas estudar a degradação da qualidade do sistema diante da insuficiência ou baixa disponibilidade dos recursos.

Além disto, também há os testes regressivos, que refazem testes já realizados em um componente (ou conjunto de componentes). É importante definir os testes que precisam ser refeitos, já que nem todos têm esta necessidade. (GUDWIN, 2015).

## 5 IBM WATSON

O IBM Watson é um sistema cognitivo capaz de compreender a linguagem natural de maneira fluída, quase como um humano faria. Foi criado em 2007 pela IBM para competir no programa norte-americano de TV Jeopardy!. Em 2011, o Watson venceu o programa, concorrendo com os dois melhores jogadores humanos até então. (LATTARO, 2016).

O Jeopardy! é um programa do tipo Quiz, baseado em história, literatura, cultura e ciência. Porém, ocorre de maneira contrária a um Quiz tradicional. Nesse modelo, são apresentadas dicas na forma de respostas e os participantes devem responder na forma de perguntas. (LATTARO, 2016).

A vitória do Watson foi possível devido a arquitetura de software DeepQA desenvolvida pela IBM. Leonel (2015, p. 1) afirma que

[...] A arquitetura DeepQA assume e persegue múltiplas interpretações da questão, gera diversas respostas, ou hipóteses, plausíveis, coleciona evidências para essas hipóteses e avalia a evidência para determinar se ela suporta ou refuta a hipótese. [...] o Watson contém centenas de diferentes algoritmos que avaliam a evidência em diferentes dimensões.

O Watson utiliza tecnologia de Processamento de Linguagem Natural para interpretar as perguntas e extrair elementos-chave, tais como os tipos de respostas e relacionamentos entre entidades. Além disso, PLN foi usado para analisar (antes da competição) grandes quantidades de texto não-estruturado (enciclopédias, dicionários, artigos, etc.) que poderiam prover as evidências no suporte de respostas às questões. Alguns dos algoritmos de Watson avaliam se os relacionamentos entre entidades na questão combinam com os da evidência.

### 5.1 APLICAÇÕES DO WATSON

Atualmente o Watson está sendo utilizado por muitas empresas nacionais e internacionais, focando em diferentes áreas. Há múltiplas aplicações:

- a) Atendimento no Bradesco: O Watson está respondendo a dúvidas para gerentes de agências através de um chat interno e, posteriormente, ajudará os atendentes a responder às perguntas dos correntistas. (ESTÚDIO ABC, 2016).
- b) Fitness e saúde na Under Armour: O Watson é responsável por uma análise dos dados registrados no aplicativo da empresa. Ele utilizará

sua capacidade de interpretar dados para orientar o usuário sobre quais caminhos usar para otimizar seu desempenho fitness e melhorar sua saúde. (ESTÚDIO ABC, 2016).

- c) Combate ao câncer no Memorial Sloan Kettering: Com base em milhões de páginas de literatura, o sistema é capaz de sugerir possíveis tratamentos. O Watson atua como um assistente, dotado de um vasto conhecimento que pode ser utilizado para ajudar os médicos até nos casos mais complicados. (ESTÚDIO ABC, 2016).
- d) Ajuda a engenheiros na Engeo: A empresa tem alimentado o Watson com informações fundamentais sobre engenharia e infraestrutura, que podem ser acessadas por profissionais ao redor do mundo, inclusive em países em desenvolvimento e em áreas atingidas por desastres naturais. Os engenheiros podem, assim, obter respostas com base na análise de uma variedade imensa de dados, o que seria difícil sem um sistema informatizado e inteligente. (ESTÚDIO ABC, 2016).
- e) Turismo eficiente na startup WayBlazer: quando um cliente faz uma pesquisa nos websites, as palavras-chave são processadas pelo Watson, que usa sua capacidade de entender linguagem natural para oferecer uma série de opções. (ESTÚDIO ABC, 2016).
- f) Novos serviços financeiros no Citigroup: A empresa armazena uma grande quantidade de dados sobre transações e características dos clientes. Estão utilizando o Watson para a análise dos dados, a fim de descobrir novas formas para facilitar o trabalho dos correntistas, além de novos produtos e maneiras de interagir com os clientes. (ESTÚDIO ABC, 2016).

## 5.2 APIS DO WATSON

Todas as aplicações citadas são realizadas por meio Interfaces de Programação de Aplicativos ou *Application Programming Interfaces* (APIs), que possuem diferentes finalidades. As APIs do Watson estão conectadas com a plataforma de nuvem da IBM, sendo listadas a seguir:

- a) Discovery: usa a análise de dados combinada com a intuição cognitiva para tirar dados não estruturados e enriquecê-los de modo que seja

possível consultá-los para retornar as informações de que precisa. (IBM, 2017).

- b) Discovery News: é um conjunto de dados públicos que foi enriquecido com percepções cognitivas e está incluído no Discovery. Discovery News inclui principalmente fontes de notícias em inglês que são atualizadas continuamente, com mais de 300.000 novos artigos e blogs adicionados diariamente, provenientes de mais de 100.000 fontes. (IBM, 2017).
- c) Document Conversion: converte um único documento HTML, PDF ou Word em um HTML normalizado, texto simples ou um conjunto de unidades de resposta formatadas em JSON que podem ser usadas com outros serviços Watson. (IBM, 2017).
- d) Language Translator: é possível criar um aplicativo que identifica o idioma do texto de entrada e usa um modelo linguístico específico do domínio para traduzir o texto em outro idioma. (IBM, 2017).
- e) Natural Language Classifier: usa algoritmos de aprendizado de máquina para retornar as classes predefinidas de correspondência superiores para entradas de texto curtas. (IBM, 2017).
- f) Natural Language Understanding: usa PLN para analisar características semânticas de qualquer texto. É possível fornecer textos simples, HTML ou um URL público, e a API retornará resultados para os recursos especificados. O serviço limpa HTML antes da análise padrão, o que remove a maioria dos anúncios e outros conteúdos indesejados. (IBM, 2017).
- g) Personality Insights: permite que aplicativos obtenham informações de mídia social, dados corporativos ou outras comunicações digitais. O serviço utiliza análises linguísticas para inferir as características intrínsecas de personalidade dos indivíduos, a partir de comunicações digitais, como e-mail, mensagens de texto, tweets e postagens de fóruns. (IBM, 2017).
- h) Retrieve and Rank: combina dois componentes de recuperação de informações em um único serviço: o poder do Apache Solr e um sofisticado recurso de aprendizado de máquina. Essa combinação fornece aos usuários resultados mais relevantes ao reclassificá-los

automaticamente usando esses algoritmos de aprendizado de máquina. (IBM, 2017).

- i) **Speech to Text:** fornece uma API que permite adicionar recursos de reconhecimento de fala da IBM às aplicações. O serviço transcreve o discurso de vários idiomas e formatos de áudio para texto. (IBM, 2017).
- j) **Text to Speech:** usa as capacidades de síntese de fala da IBM para sintetizar texto em voz natural em uma variedade de idiomas, sotaques e vozes. O serviço suporta pelo menos uma voz masculina ou feminina, às vezes ambas, para cada idioma. (IBM, 2017).
- k) **Tone Analyzer:** utiliza análise linguística para detectar tons de comunicação em texto escrito. Serve para entender conversas e comunicações e, em seguida, responder aos clientes apropriadamente. (IBM, 2017).
- l) **Visual Recognition:** usa algoritmos de aprendizado para identificar cenas, objetos e rostos de celebridades em imagens enviadas para o serviço. É possível criar e treinar um classificador personalizado para identificar assuntos que atendam às necessidades do usuário. (IBM, 2017).

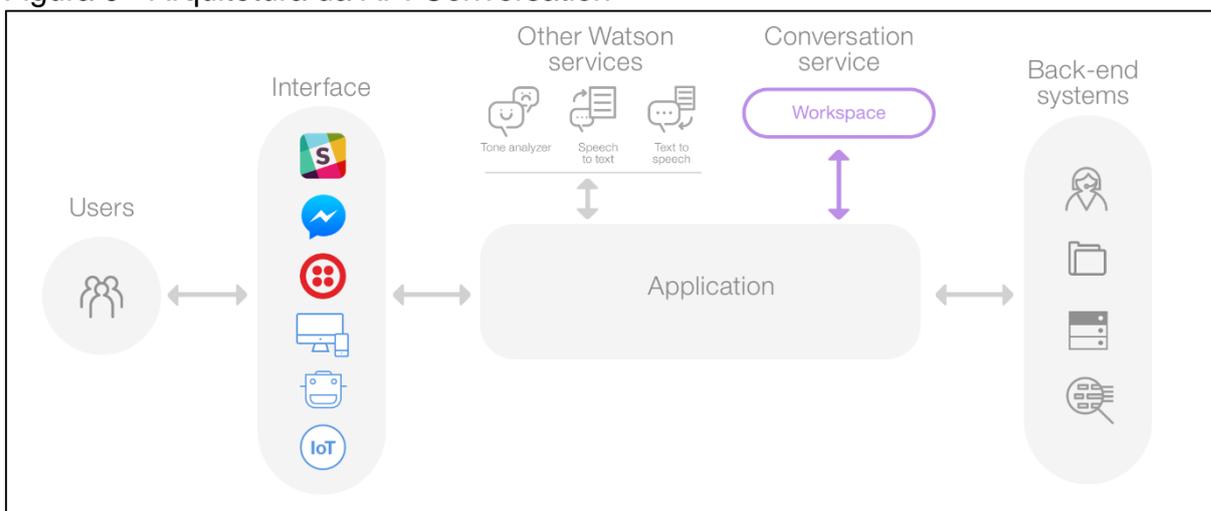
Como o foco do trabalho é desenvolver um *chatbot* empresarial, a API *Conversation* é abordada detalhadamente na próxima seção.

### 5.3 CONVERSATION

Através da API *Conversation* é possível criar um aplicativo que entenda uma entrada em linguagem natural e usa o aprendizado de máquina para responder aos clientes de uma forma que simula uma conversa entre humanos. (IBM, 2017).

É possível integrar o *Conversation* com várias interfaces atualmente, como o Messenger do Facebook, Slack, Telegram, entre outras. A Figura 9 demonstra como é a arquitetura utilizada na API.

Figura 9 - Arquitetura da API Conversation



Fonte: IBM (2017).

De acordo com a IBM (2017), os usuários interagem com a aplicação por meio de uma interface. Por exemplo, uma janela de chat, um aplicativo para celular ou até mesmo um robô com uma interface de voz. A aplicação envia a entrada do usuário para o serviço *Conversation*.

Para isto, é necessário que a aplicação esteja conectada a um *Workspace*, que é um contêiner para o fluxo de diálogo e dados de treinamento. O serviço interpreta a entrada do usuário, direciona o fluxo da conversa e reúne as informações de que precisa. É possível conectar serviços adicionais para analisar a entrada do usuário, como o *Tone Analyzer* ou *Speech to Text*. (IBM, 2017).

A aplicação pode interagir com sistemas back-end baseada na intenção do usuário e informações adicionais. Por exemplo, responder perguntas, abrir ingressos, atualizar informações de conta ou fazer pedidos. (IBM, 2017).

O serviço *Conversation* suporta muitos idiomas, como português brasileiro, inglês, francês, italiano, espanhol, alemão, chinês tradicional, chinês simplificado, holandês e árabe. (IBM, 2017).

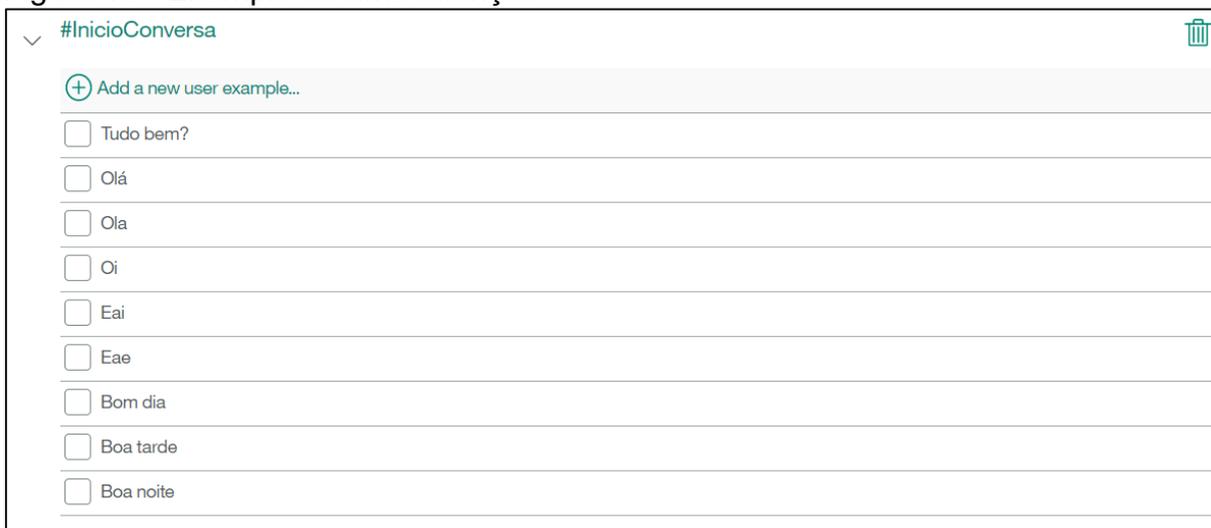
Para construir a base de conhecimento do *chatbot*, o *Conversation* trabalha com Intenções, Entidades e Diálogo, que são abordados nas seções a seguir.

### 5.3.1 Intenções

Uma intenção (ou *intent*, Figura 10) é uma finalidade ou meta demonstrada na entrada de um cliente, como responder a uma pergunta ou processar um pagamento

de fatura. Ao reconhecer a intenção expressa na entrada de um cliente, o serviço *Conversation* pode escolher o fluxo de diálogo correto para responder a ele. (IBM, 2017).

Figura 10 - Exemplo de uma intenção



▼ #InicioConversa 

 Add a new user example...

Tudo bem?

Olá

Ola

Oi

Eai

Eae

Bom dia

Boa tarde

Boa noite

Fonte: Elaborada pelo autor.

Para planejar as intenções da aplicação, é necessário considerar o que os clientes querem fazer e o que o aplicativo é capaz de manipular. Escolher a intenção correta para a entrada de um usuário é o primeiro passo para fornecer uma resposta útil. As intenções identificadas para a aplicação determinarão os fluxos de diálogo que serão criados; Elas também podem determinar com quais sistemas de back-end a aplicação precisa se integrar para completar solicitações de clientes (como bancos de dados de clientes ou sistemas de processamento de pagamentos). (IBM, 2017).

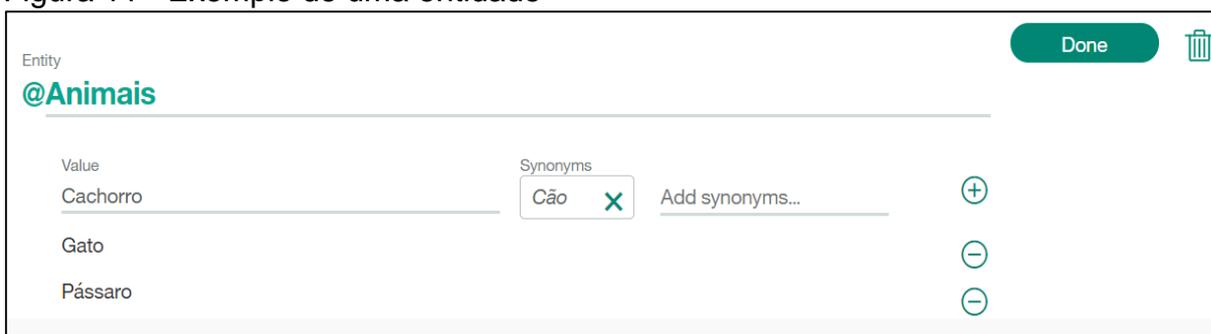
Segundo a IBM (2017), é necessário reunir perguntas reais de clientes, comandos ou outras entradas. Utilizando entradas de usuários reais, é possível ter uma ideia melhor da entrada esperada. Também é importante lembrar que os clientes podem expressar o mesmo tipo de solicitação de maneiras diferentes.

As intenções são identificadas com “#”, que é adicionado automaticamente pela ferramenta. Para criar uma é necessário adicionar exemplos (no mínimo 5) para que o Watson identifique a mesma. É importante colocar muitos exemplos, para que o Watson consiga reconhecer a Intenção facilmente. Outro ponto interessante é que a entrada do usuário não precisa ser igual aos exemplos cadastrados. O Watson reconhece entradas parecidas utilizando PLN.

### 5.3.2 Entidades

Uma entidade (ou *entity*, Figura 11) representa um tipo de dado que é relevante para a finalidade de um usuário. Ao reconhecer as entidades mencionadas na entrada do usuário, o serviço *Conversation* pode escolher as ações específicas a serem tomadas para cumprir uma Intenção. (IBM, 2017).

Figura 11 - Exemplo de uma entidade



Fonte: Elaborada pelo autor.

A IBM (2017, p. 5) afirma que

Uma entidade representa um termo ou objeto na entrada do usuário que fornece esclarecimento ou contexto específico para uma intenção particular. Se intenções representam verbos (algo que um usuário quer fazer), as entidades representam substantivos (como o objeto ou o contexto de uma ação). As entidades permitem que uma única intenção represente múltiplas ações específicas. [...]

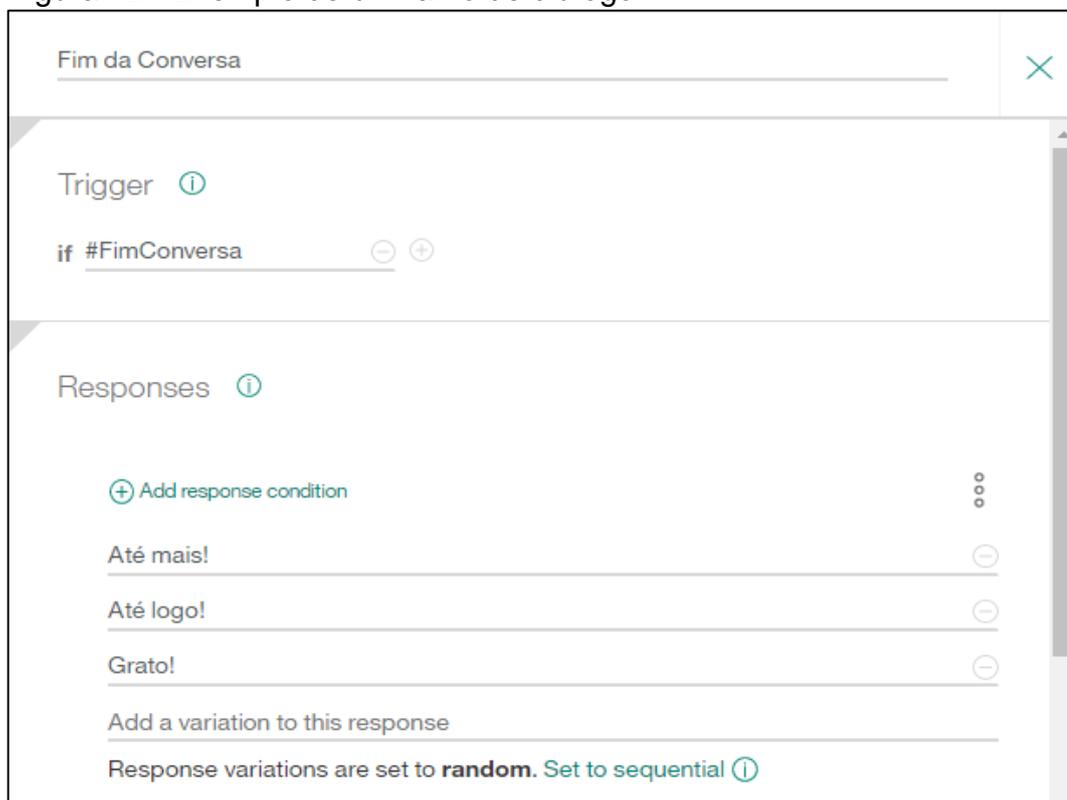
As entidades são identificadas com “@”, que é criado pela ferramenta de forma automática. Ao cadastrar uma entidade, é necessário adicionar os valores que fazem parte da mesma. Cada um pode ter vários sinônimos, melhorando o reconhecimento do Watson.

### 5.3.3 Diálogo

O componente de diálogo (ou *dialog*, Figura 12) do *Conversation* usa as intenções e entidades que são identificadas na entrada do usuário, além do contexto da aplicação, para interagir com o usuário e fornecer uma resposta útil. O diálogo é

representado graficamente como uma árvore, onde é possível criar um ramo para processar cada intenção definida. (IBM, 2017).

Figura 12 - Exemplo de um ramo de diálogo



Fonte: Elaborada pelo autor.

A IBM (2017, p. 6) afirma que

Um diálogo é composto por nós, que definem passos na conversação. Os nós de diálogo são encadeados em uma estrutura em árvore para criar uma conversação interativa com o usuário final. Cada nó inclui condições para o nó estar ativo, bem como um objeto de saída que define a resposta fornecida. Você pode pensar no nó como uma construção if / then: se essa condição for verdadeira, em seguida, retornar esta resposta. A condição mais simples é uma única intenção, o que significa que a resposta é retornada se a entrada do usuário corresponder a essa intenção. Crie um ramo de diálogo que corresponda a cada intenção que você identificou. Esse ramo é usado quando essa intenção é reconhecida.

Primeiramente, é necessário definir a intenção para o ramo de diálogo. Depois são adicionadas as respostas, que podem ser variadas, permitindo que o Watson responda de várias formas sobre um assunto. Elas podem aparecer de forma sequencial ou aleatória.

Quando o Watson reconhece a intenção, retorna a resposta adicionada. É possível trabalhar com respostas gerais, utilizando apenas intenções. Para gerar respostas específicas, é necessário utilizar as entidades.

### 5.3.4 Testes e Melhorias

Durante o desenvolvimento do *chatbot*, os testes são realizados por meio do painel Try it out (Figura 13).

Figura 13 - Exemplo de uma conversa

Try it out Clear

Gostaria de fazer um pedido.

#NovoPedido

Os pedidos devem ser feitos após estar logado no site. Caso já tenha um cadastro, efetue o login, caso contrário cadastre-se no site.

Vocês vendem TV?

#VerificaçãoProduto

Desculpe, não trabalhamos com este produto.

Enter something to test your bot  
Use the up key for most recent

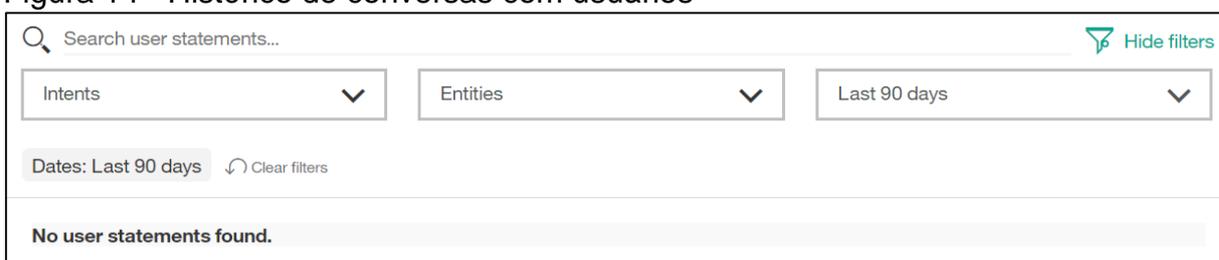
Fonte: Elaborada pelo autor.

Através deste painel, é possível verificar se o Watson está reconhecendo as intenções e entidades corretamente, e retornando uma resposta adequada.

Se o Watson não identificar a intenção correta, é possível corrigi-la. Para isto, é necessário selecionar a intenção exibida e trocar pela intenção correta. Após a correção ser submetida, o sistema automaticamente treina para incorporar os novos dados. (IBM, 2017).

Também há um histórico de conversas com usuários (Figura 14). É possível usá-lo para melhorar a compreensão do *chatbot* das entradas dos usuários, adicionando as mensagens como exemplos de intenções. Ou ensinar novos termos ao *chatbot* adicionando entidades, valores ou sinônimos. (IBM, 2017).

Figura 14 - Histórico de conversas com usuários



The screenshot shows a user interface for viewing chatbot history. At the top, there is a search bar labeled "Search user statements...". Below the search bar are three filter dropdown menus: "Intents", "Entities", and "Last 90 days". To the right of these filters is a "Hide filters" button. Below the filters, there is a "Dates: Last 90 days" button and a "Clear filters" button. The main content area displays "No user statements found."

Fonte: Elaborada pelo autor.

Através do histórico, é possível pesquisar entradas dos usuários, ou filtrar os resultados por meio das intenções, entidades e datas de conversas.

## 6 TRABALHOS CORRELATOS

Na pesquisa realizada foram encontrados *chatbots* relacionados à área empresarial, que foram desenvolvidos com a intenção de se comunicar com clientes. Além disto, também há projetos focados em outros temas importantes, como a área social, a educação, o meio ambiente e a saúde.

No contexto empresarial e social foram encontrados os seguintes trabalhos:

- a) Eliza: foi um dos primeiros *chatbots*, desenvolvido por Joseph Weizenbaum em 1996. Funcionava como um psicólogo, ajudando as pessoas com seus problemas;
- b) Alice Seguros: é um *chatbot* focado em contratar seguros para os usuários;
- c) Cloudia: é um *chatbot* que automatiza a comunicação de profissionais e instituições de saúde com os seus pacientes ou clientes;
- d) IFood Pizza Bot: é capaz de vender pizzas e mostrar restaurantes da região;
- e) Waypoint: descobre a localização de linhas de ônibus e mostra a situação do metrô;
- f) Mr Joboto: *chatbot* desenvolvido para auxiliar as pessoas na busca de oportunidades de trabalho;
- g) Ana: *chatbot* da empresa de telecomunicações Algar Telecom, sendo capaz de responder perguntas sobre 4G;
- h) Ben: *chatbot* da Rede de Farmácias Panvel. Através dele é possível consultar pedidos, verificar pontuação de programa de fidelidade, buscar lojas próximas, entre outras opções.

Além disto, foi encontrado o trabalho de conclusão de curso intitulado “Desenvolvimento e implementação web de um Chatterbot para Educação Ambiental utilizando AIML e PROLOG”, realizado por Vinicius Tonelli de Oliveira da Universidade do Sagrado Coração (USC) em 2016, que criou um *chatbot* com o objetivo de ensinar sobre Educação Ambiental, usando as linguagens AIML, PROLOG e a ferramenta PROGRAM-O para desenvolvimento do mesmo.

No contexto ambiental, também há o trabalho de conclusão de curso intitulado “Implementação de um Atendente Virtual especializado em Licenciamento Ambiental utilizando AIML”, realizado por Murilo Luizetto Gonçalves de Aguiar da USC em

2013, que desenvolveu um *chatbot* para tirar dúvidas relacionadas ao processo de licenciamento ambiental, através da linguagem AIML e da ferramenta PROGRAM-O.

Na área da saúde, foi encontrado o trabalho de conclusão de curso intitulado “Teo: Um Chatterbot para Telessaúde”, realizado por Helcio da Silva da USC em 2014, que criou um *chatbot* para ajudar na prática terapêutica ocupacional no âmbito hospitalar, utilizando a linguagem AIML e a ferramenta PROGRAM-O.

Através da pesquisa realizada, foi possível encontrar muitas aplicações para os *chatbots* nas empresas e outras áreas. No entanto, ainda há poucos trabalhos envolvendo o Watson, que é uma ferramenta recente desenvolvida pela IBM. Portanto, o foco deste trabalho é demonstrar as funcionalidades e potencialidades do Watson na criação de *chatbots* empresariais.

## 7 METODOLOGIA

O IBM Watson foi utilizado por ser uma ferramenta atual com muitas possibilidades, tendo uma interface que permite o desenvolvimento rápido e eficiente de *chatbots* voltados para a área empresarial. Além disto, o Watson também pode ser integrado com muitas plataformas (Messenger, Telegram, Slack, etc.), permitindo que a empresa se comunique com seus clientes de várias formas.

Para o desenvolvimento deste projeto foram necessárias duas fases distintas: pesquisa teórica e criação do *chatbot*.

Primeiramente, houve uma pesquisa relacionada aos assuntos importantes para o projeto como Inteligência Artificial, Processamento de Linguagem Natural, Aprendizado de Máquina, *chatbots*, Engenharia de Software e IBM Watson.

Antes de iniciar o desenvolvimento, também foi necessário levantar informações importantes sobre a empresa escolhida, através de conversas com o gerente. Através delas, percebeu-se que o setor de atendimento da empresa estava sobrecarregado pela quantidade de clientes diários, que faziam perguntas frequentes sobre produtos, cursos e serviços da empresa.

Por meio do setor de atendimento, foi possível levantar muitas informações importantes para a modelagem do *chatbot*. Através dos diagramas criados, também foram coletadas informações relevantes.

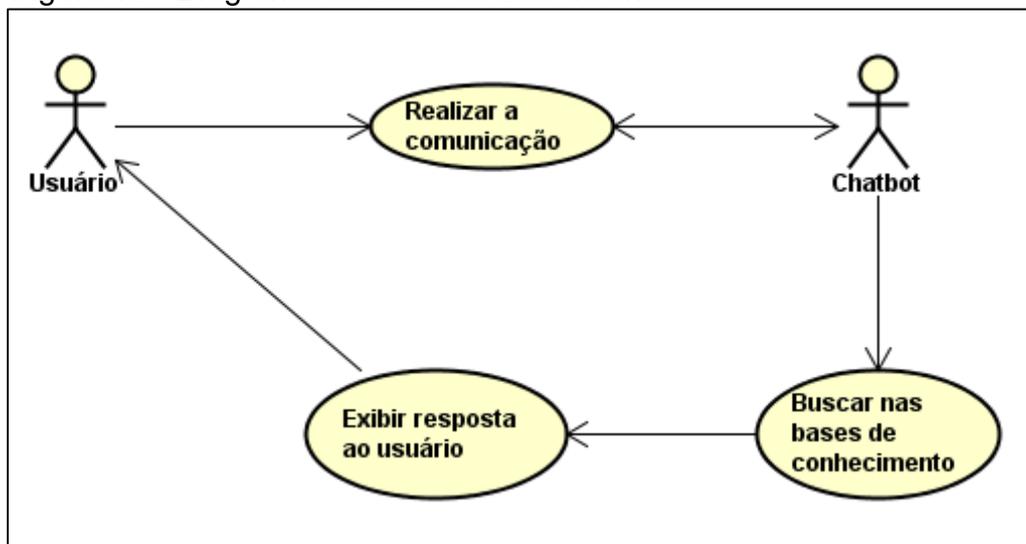
Na segunda fase, a base de conhecimento do *chatbot* foi criada e testes foram realizados com funcionários da empresa, para verificar o desempenho do mesmo. Por meio dos testes foi refinada a base de conhecimento, para que, posteriormente, o *chatbot* fosse integrado com o Facebook da empresa.

### 7.1 MODELAGEM UML DO CHATBOT

Antes de começar o desenvolvimento, é importante realizar a modelagem do *chatbot* por meio da linguagem UML. Foram desenvolvidos os Diagramas de Caso de Uso e Atividades através da ferramenta Astah Community para representar seu funcionamento. Como a base de conhecimento não está armazenada em um banco de dados, não foi necessário um Diagrama de Classes.

A Figura 15 representa como o *chatbot* interage com o usuário e a base de conhecimento através do Diagrama de Caso de Uso. Há dois atores (usuário e *chatbot*) e dois casos de uso.

Figura 15 - Diagrama de Caso de Uso do chatbot

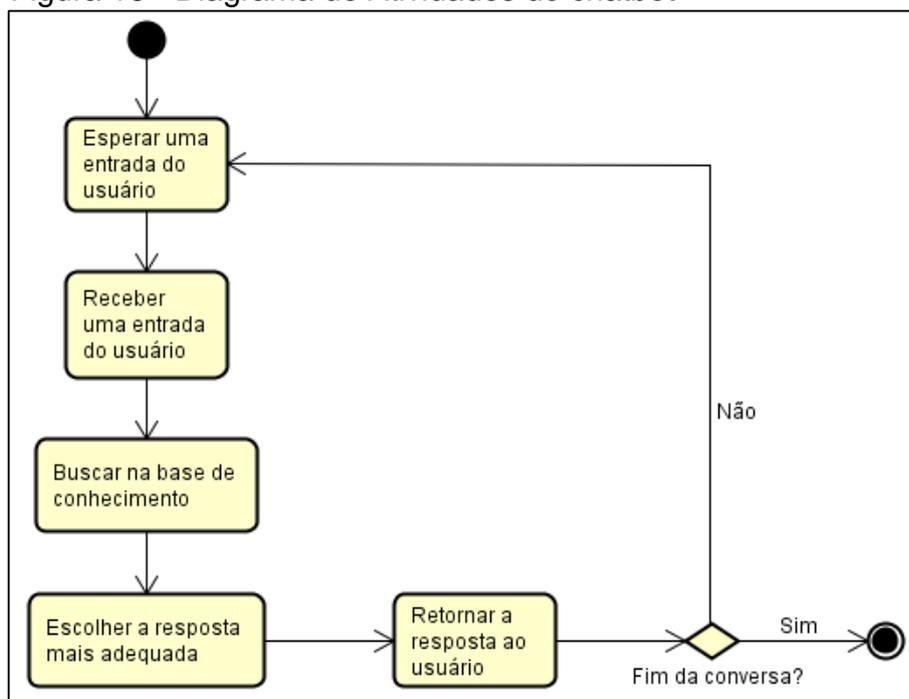


Fonte: Elaborada pelo autor.

Primeiramente, o usuário se comunica com o *chatbot*, perguntando sobre algum assunto relacionado à empresa. Após isto, ele responde o usuário, buscando as informações necessárias na base de conhecimento.

A Figura 16 representa a sequência de ações do *chatbot*, através do Diagrama de Atividades. Primeiramente, o *chatbot* espera uma entrada do usuário, para depois buscar na base de conhecimento a resposta mais adequada. Conseqüentemente, a resposta é mostrada ao usuário. Este ciclo acaba quando o usuário termina a conversa.

Figura 16 - Diagrama de Atividades do chatbot



Fonte: Elaborada pelo autor.

## 7.2 BASE DE CONHECIMENTO

A empresa estudada é do setor funerário, com produtos voltados para a Tanatopraxia (técnica de conservação para velórios ou funerais). Neste contexto, há muitos produtos químicos, equipamentos e instrumentais relacionados com a área funerária. Além disto, a empresa trabalha com assessoria e ministra cursos teórico-práticos nesta área.

A construção da base de conhecimento foi realizada através do serviço *Conversation* do IBM Watson. Muitas informações importantes da empresa foram coletadas com o setor de atendimento, estando relacionadas a produtos, cursos e serviços da mesma.

O IBM Watson trabalha com Intenções, Entidades e Diálogo para a criação da base de conhecimento. Estes itens são abordados nas seções a seguir.

### 7.2.1 Intenções

As intenções representam o propósito da entrada do cliente, ou seja, o que ele está buscando na conversa. Para que o *chatbot* entenda a intenção do cliente na

sentença, é necessário cadastrar vários exemplos. Através deles, o *chatbot* consegue detectar uma intenção.

Por meio de conversas com o setor de atendimento, foi possível descobrir quais as principais dúvidas dos clientes em relação à empresa. Com base nisto, houve o cadastro das intenções, abordando os principais assuntos procurados. A Figura 17 mostra algumas intenções cadastradas no Watson.

Figura 17 - Exemplos de intenções cadastradas

>	<b>#Pedidos</b> Ainda não chegou.	7
>	<b>#UtilizaçãoProdutos</b> Como devo usar?	19
>	<b>#Quantidade</b> A caixa vem com quantas unidades?	6
>	<b>#AplicaçãoProdutos</b> Como aplicar?	5

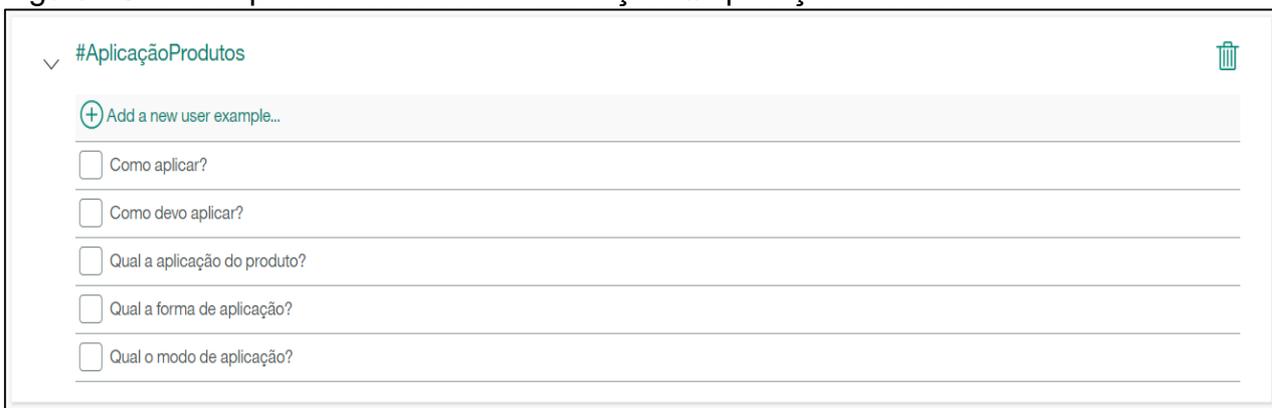
Fonte: Elaborada pelo autor.

Como exemplo, as intenções cadastradas representam algumas dúvidas relacionadas à empresa. Muitos clientes perguntam sobre utilização de determinado produto ou sua aplicação. Também há questões envolvendo a quantidade de produtos em determinada embalagem, ou a situação de pedidos.

Os números do lado direito representam os exemplos cadastrados em determinada intenção. Quanto maior a quantidade de exemplos, melhor será a detecção do *chatbot* em relação à intenção.

A Figura 18 demonstra os exemplos cadastrados na intenção “#AplicaçãoProdutos”.

Figura 18 - Exemplos cadastrados na intenção “#AplicaçãoProdutos”



Fonte: Elaborada pelo autor.

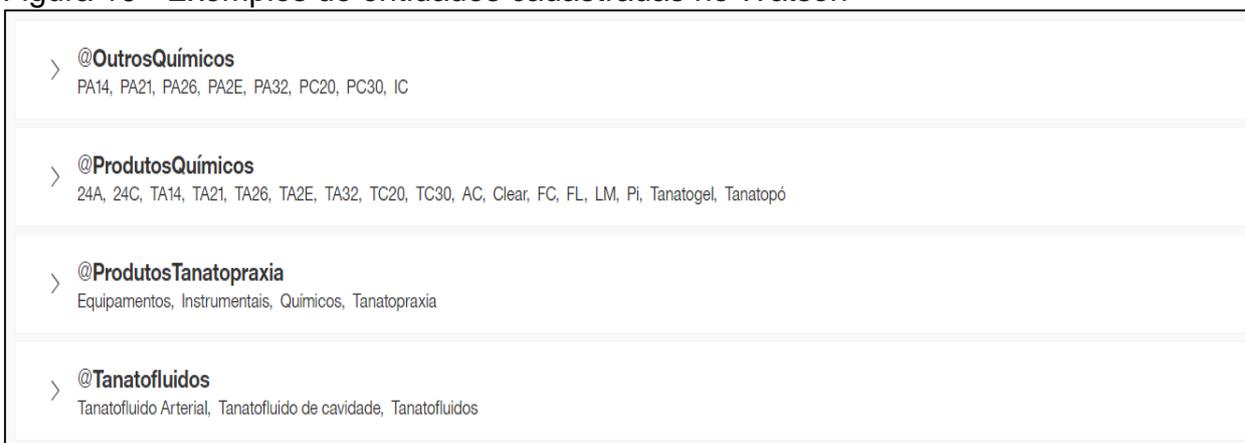
Através dos exemplos cadastrados, o *chatbot* entende a intenção. Para isto, não é necessário que a pergunta seja exatamente igual aos exemplos cadastrados. Ele é capaz de entender perguntas parecidas e concluir que se trata da intenção “#AplicaçãoProdutos”.

## 7.2.2 Entidades

Uma entidade representa um termo ou objeto que é relevante para seus propósitos e que fornece um contexto específico para uma intenção. Portanto, é importante para que o *chatbot* responda uma dúvida específica.

No contexto da empresa, os clientes fazem perguntas constantemente sobre os produtos químicos, equipamentos e instrumentais presentes no catálogo. A Figura 19 mostra alguns exemplos de entidades cadastradas no Watson.

Figura 19 - Exemplos de entidades cadastradas no Watson



Fonte: Elaborada pelo autor.

Neste caso, as entidades representam os produtos químicos da empresa. Cada entidade tem um conjunto de nomes de produtos, que estão presentes no catálogo. A Figura 20 demonstra os nomes de alguns produtos cadastrados na entidade “@ProdutosQuímicos”.

Figura 20 - Produtos cadastrados na entidade “@ProdutosQuímicos”

<input type="checkbox"/>	TC20	Add synonyms...	(Synonyms)
<input type="checkbox"/>	TC30	Add synonyms...	(Synonyms)
<input type="checkbox"/>	AC	Tanatosolvente A..	(1Synonym)
<input type="checkbox"/>	Clear	Desinfetânatus Tô...	(1Synonym)
<input type="checkbox"/>	FC	Cor Facial	(1Synonym)

Fonte: Elaborada pelo autor.

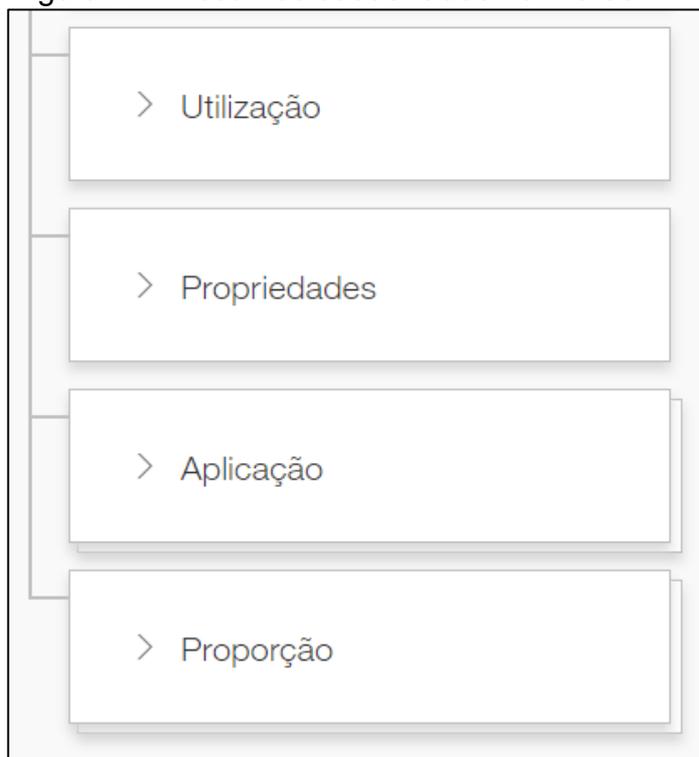
Através dos exemplos cadastrados, o *chatbot* consegue detectar um produto específico mencionado pelo cliente. Muitos produtos têm nomes diferentes, portanto é necessário cadastrar os sinônimos, para que sejam detectados na conversa.

### 7.2.3 Diálogo

Por meio das intenções e entidades, é possível criar as respostas na seção de Diálogo. Aqui o fluxo da conversa é definido, abordando assuntos gerais e específicos.

Em cada ramo de diálogo é adicionado um assunto procurado pelos clientes. A Figura 21 mostra alguns assuntos cadastrados.

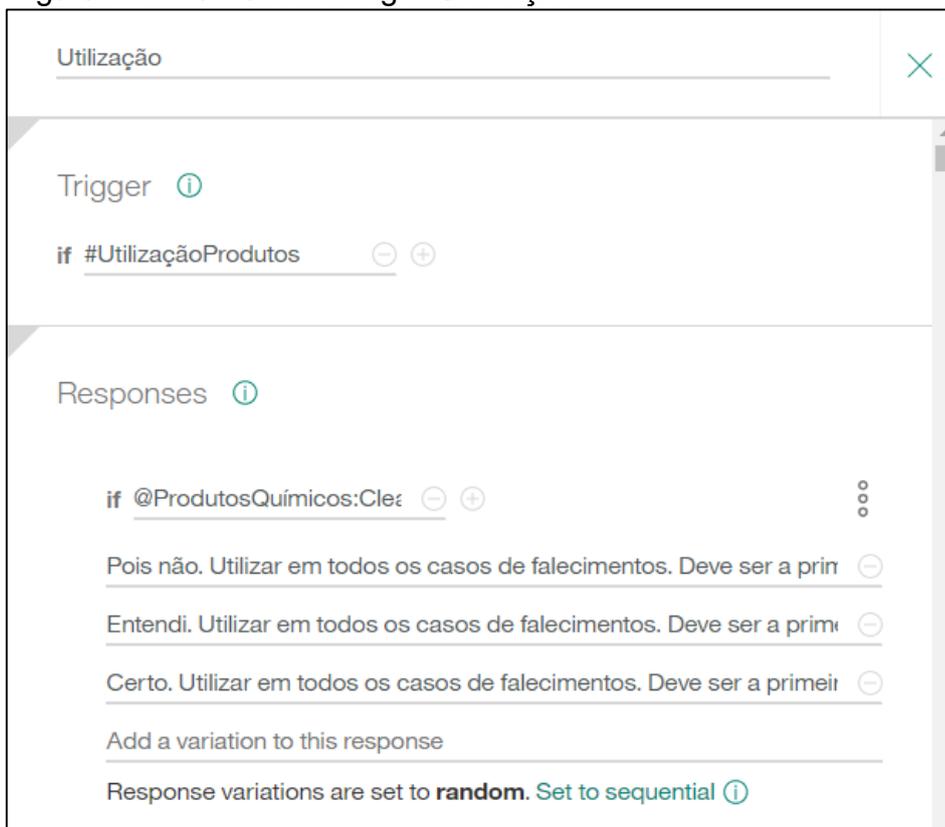
Figura 21 - Assuntos cadastrados no Watson



Fonte: Elaborada pelo autor.

Muitos clientes buscam informações sobre a utilização, propriedades, aplicação e proporção de produtos químicos, equipamentos e instrumentais da empresa. Neste caso, as respostas devem ser específicas para cada item do catálogo, utilizando intenções e entidades. A Figura 22 mostra o ramo de diálogo “Utilização”.

Figura 22 - Ramo de diálogo “Utilização”



Fonte: Elaborada pelo autor.

No ramo de diálogo é necessário que a intenção seja chamada no campo “Trigger”. No caso, a intenção “#UtilizaçãoProdutos” foi adicionada. Para gerar uma resposta específica, é necessário chamar a entidade que corresponde ao produto no campo “Responses”. Foi adicionada a entidade “@ProdutosQuímicos:Clear”, representando um produto químico do catálogo da empresa. Com isto, o *chatbot* consegue entender que o cliente busca a utilização do produto “Clear”, permitindo o retorno de uma resposta adequada.

Também foram adicionadas variações de resposta, que podem aparecer para o cliente de forma aleatória ou sequencial. Com isto, o *chatbot* consegue responder de formas diferentes sobre o mesmo assunto.

Além de respostas específicas, também foram criados ramos de diálogo para respostas gerais. Nestes casos, são utilizadas apenas intenções, sem a necessidade de entidades. A Figura 23 demonstra um ramo de diálogo para um assunto geral.

Figura 23 - Ramo de diálogo “Localização da Empresa”



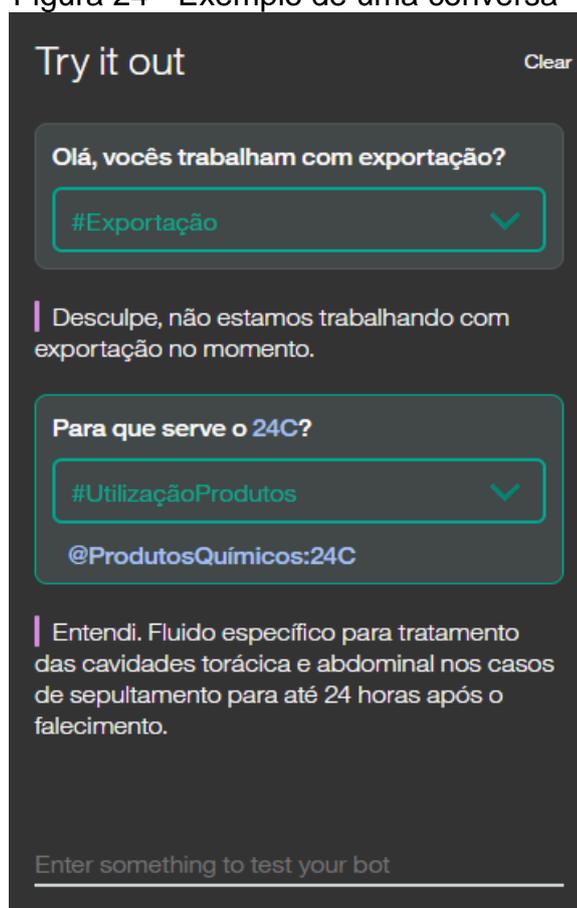
Fonte: Elaborada pelo autor.

Neste caso, foi adicionada somente a intenção “#LocalizaçãoEmpresa” e uma única resposta. Sempre que o cliente perguntar sobre a localização da empresa, receberá esta resposta com o endereço da mesma.

### 7.3 TESTES E MELHORIAS

Foram realizados testes com o *chatbot*, através de conversas com o mesmo. O Watson possui uma seção para conversa, onde foi possível testar a detecção de intenções e entidades, e verificar se a resposta foi apropriada. A Figura 24 mostra um exemplo de conversa com *chatbot*.

Figura 24 - Exemplo de uma conversa



Fonte: Elaborada pelo autor.

No exemplo mostrado, o *chatbot* detectou corretamente a intenção “#Exportação” e gerou a resposta apropriada. Depois identificou a intenção “#UtilizaçãoProdutos” e a entidade “@ProdutosQuímicos:24C”, retornando uma resposta específica adequada.

Os testes foram feitos constantemente com funcionários da empresa, possibilitando um feedback imediato. Quando houve respostas incorretas, a base de conhecimento foi atualizada para corrigir o problema, permitindo um processo de refinamento da mesma.

Além disto, há o histórico de conversas, que é responsável por registrar todas as mensagens. No entanto, os testes realizados através da interface do *Conversation* não são registrados pelo mesmo, que armazena somente conversas com clientes através do Messenger. Portanto, só foi possível utilizá-lo após o processo de integração com o Facebook.

## 7.4 INTEGRAÇÃO DO CHATBOT

Após o refinamento da base de conhecimento, o *chatbot* foi integrado com o Facebook da empresa, a fim de se comunicar com os clientes. Primeiramente, foi necessário acessar o portal de desenvolvedores do Facebook, com o objetivo de criar um novo Aplicativo ou *Application* (App) para a página do Facebook da empresa. A Figura 25 demonstra a área de criação de um novo App.

Figura 25 - Área de criação de um novo App



Crie um novo número de identificação do aplicativo

Comece integrando o Facebook ao seu aplicativo ou site

Nome de exibição

O nome que você quer associar a este número de identificação do aplicativo

Email de contato

Usado para comunicados importantes sobre o seu aplicativo

Ao prosseguir, você concorda com as [Políticas da Plataforma do Facebook](#)

Cancelar Crie um número de identificação do aplicativo

Fonte: Elaborada pelo autor.

Posteriormente foi realizado o processo de criação do Orquestrador, que é responsável pela comunicação do Messenger com o serviço *Watson Conversation* da IBM. Para isto, foi necessário acessar o IBM Bluemix (plataforma da nuvem da IBM), e criar um novo App da categoria “Cloud Foundry” chamado “SDK for Node.JS”. A Figura 26 mostra a área de criação do App “SDK for Node.JS”.

Figura 26 - Área de criação do App “SDK for Node.JS”



Nome do app:

Inserir um nome exclusivo

Nome do host:

Inserir um nome exclusivo

Domínio:

mybluemix.net

Selecionar região para implementação:

Sul dos EUA

Escolha uma organização:

ProjetoUSC

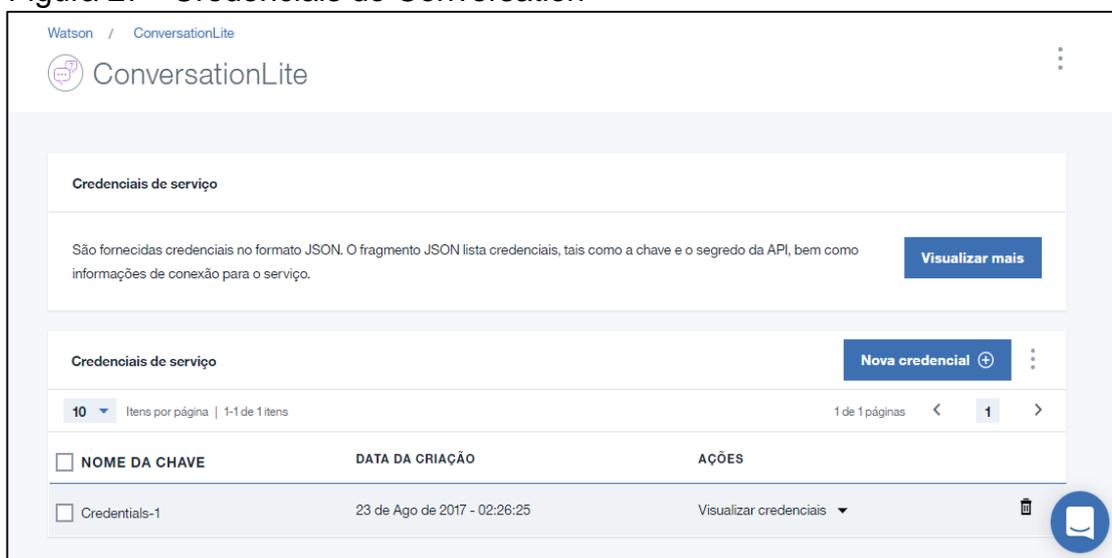
Escolha um espaço:

dev

Fonte: Elaborada pelo autor.

Para efetuar a comunicação entre os serviços, o Orquestrador utiliza o ID do Workspace (contêiner com as Intenções, Entidades e Diálogo), usuário e senha do serviço *Watson Conversation*, assim como o token gerado pelo Facebook. No serviço *Conversation*, ao clicar no campo “View details” do Workspace, é possível visualizar o ID do mesmo. Por outro lado, o usuário e a senha são obtidos ao acessar as credenciais do *Conversation*, clicando em “Visualizar credenciais”, como mostra a Figura 27.

Figura 27 - Credenciais do Conversation



Fonte: Elaborada pelo autor.

Para se obter o token do Facebook, é preciso acessar o campo “Messenger” no portal de desenvolvedores e, depois, “Geração de token” (Figura 28). Nesta área foi selecionada a página do Facebook da empresa e o token correspondente foi gerado.

Figura 28 - Campo de Geração de token

**Geração de token**

É necessário o token da Página para começar a usar as APIs. O token desta Página terá todas as permissões do Messenger mesmo se o seu aplicativo ainda não estiver aprovado para usá-las, contudo, neste caso, você poderá enviar mensagens apenas para os administradores do aplicativo. Você também pode gerar tokens para Páginas que não são suas usando o Login do Facebook.

Página Token de acesso da Página

Selecionar uma página ▾ You must select a Page to generate an access token.

[Criar uma nova Página](#)

Fonte: Elaborada pelo autor.

Neste contexto, foi definido o Webhook no portal de desenvolvedores, que é responsável por fazer a conexão e enviar mensagens quando algum cliente interagir com o *chatbot*. Foram selecionadas as opções “messages”, “messaging\_optins”, “message\_deliveries” e “messaging\_postbacks”, que são as funcionalidades de chat utilizadas. Após isto, a página do Facebook foi selecionada no campo “Select a page to subscribe your webhook to the page events”. A Figura 29 demonstra o campo Webhooks.

Figura 29 - Campo Webhooks

**Webhooks** Editar eventos

Para receber mensagens e outros eventos enviados pelos usuários do Messenger, o aplicativo precisa ativar a integração de webhooks. ✔ Concluído

Eventos selecionados: messages, messaging\_postbacks, messaging\_optins, message\_deliveries

Select a page to subscribe your webhook to the page events Selecione uma Página ▾

Páginas com inscrição: Tanatus Indústria e Comércio

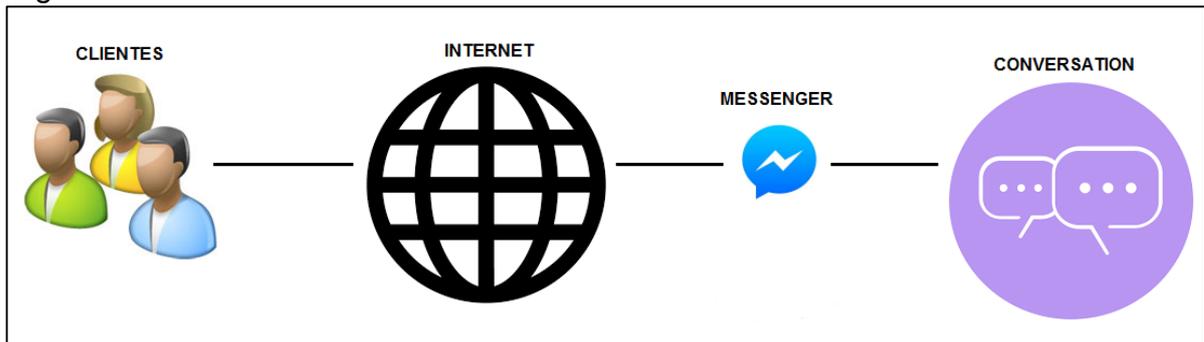
Fonte: Elaborada pelo autor.

Com o App configurado e as credenciais adicionadas ao Orquestrador, é necessário solicitar uma avaliação do Facebook, para que o App seja aprovado e fique disponível para o público. No campo “Avaliação do aplicativo para Messenger”, é preciso adicionar a opção “pages\_messaging”, selecionar a página do Facebook e adicionar alguns exemplos de respostas do *chatbot*, que serão avaliados. Após a aprovação, o *chatbot* se tornou público para todos os usuários.

## 7.5 ESTRUTURA DO SISTEMA

Os clientes podem interagir com o *chatbot* por meio do Messenger. Para isto, é necessária uma conexão com a Internet. As entradas dos clientes são enviadas para o serviço *Conversation* do IBM Watson, que retorna uma resposta adequada. A Figura 30 mostra como funciona a estrutura do sistema.

Figura 30 - Estrutura do sistema



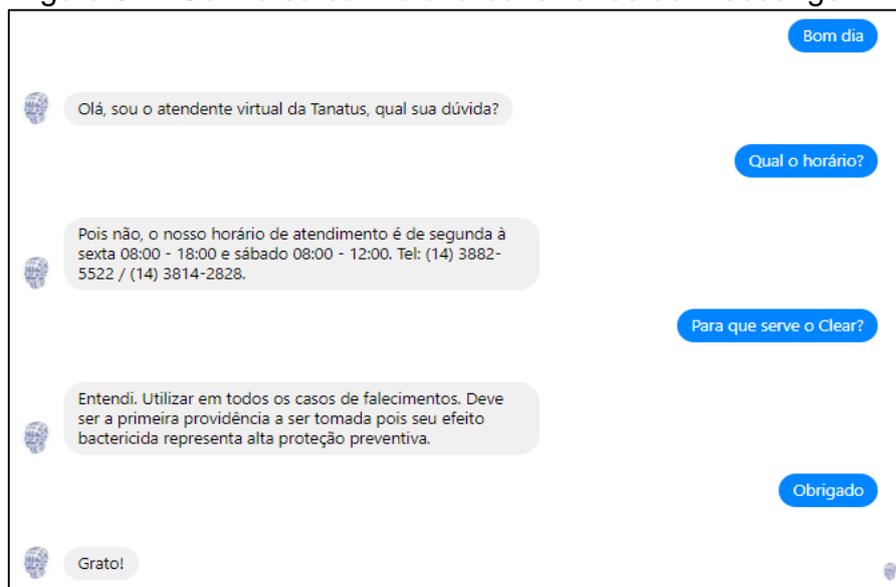
Fonte: Elaborada pelo autor.

## 8 RESULTADOS E DISCUSSÕES

Após o processo de integração e aprovação do Facebook, o *chatbot* ficou disponível para tirar dúvidas dos clientes através do Messenger. Todas as mensagens foram registradas no histórico de conversas do *Conversation*, que foi utilizado para verificar se o *chatbot* estava detectando as Intenções e Entidades corretamente, e retornando uma resposta apropriada. Novos assuntos abordados pelos clientes puderam ser detectados e adicionados à base de conhecimento, melhorando o desempenho do *chatbot*.

Foram efetuadas várias conversas, para verificar se o *chatbot* estava respondendo de forma apropriada. A Figura 31 demonstra um exemplo de uma conversa realizada através do Messenger.

Figura 31 - Conversa com o chatbot através do Messenger



Fonte: Elaborada pelo autor.

Neste exemplo, o *chatbot* detectou corretamente as Intenções e Entidades, respondendo de forma adequada. Primeiramente, detectou o início da conversa e se apresentou para o usuário. Posteriormente, identificou uma dúvida geral e uma específica, respondendo sobre o horário de atendimento e a utilização de um produto presente no catálogo da empresa. Por fim, detectou o agradecimento e respondeu de forma apropriada. No histórico de conversas, as mensagens ficaram registradas, assim como mostra a Figura 32.

Figura 32 - Mensagens registradas

The screenshot displays a chatbot interface with a list of recorded messages. At the top, there is a 'Filter' section with 'Intents' and 'Entities' dropdown menus, a search bar for 'Search user statements...', and a 'Refresh data' button. The date is set to '29 de Outubro de 2017'. Below the filter, it shows 'Showing 1 through 4 of 4 results' and a 'Newest first' sort order. The messages are as follows:

Message	Intent	Entities
Obrigado	#FimConversa	No entities
Para que serve o Clear?	#UtilizaçãoProdutos	@ProdutosQuímicos:Clear
Qual o horário?	#Horário	No entities
Bom dia	#InicioConversa	No entities

Fonte: Elaborada pelo autor.

No histórico de conversas, é possível filtrar as mensagens por dia, meses e anos, além de serem mostradas as Intenções e Entidades detectadas. Neste caso, o *chatbot* identificou corretamente as Intenções “#InicioConversa”, “#Horário”, “#UtilizaçãoProdutos” e “#FimConversa”. Quando houve uma dúvida específica sobre um produto químico, também foi identificada a Entidade “@ProdutosQuímicos:Clear”. Além disto, também é possível abrir a conversa através do campo “Open Conversation”, para visualizar as respostas do *chatbot*, assim como demonstra a Figura 33.

Figura 33 - Conversa registrada

The screenshot shows a chatbot conversation. The user asks: "Para que serve o Clear?". The bot responds: "Entendi. Utilizar em todos os casos de falecimentos. Deve ser a primeira providência a ser tomada pois seu efeito bactericida representa alta proteção preventiva." The user replies: "Obrigado". The bot replies: "Grato!". There are '+ Show classifications' links next to both the user and bot messages.

Fonte: Elaborada pelo autor.

Muitos clientes conversaram com o *chatbot*, perguntando sobre vários assuntos relacionados a empresa, como endereço, produtos, cursos, entre outros. A Figura 34 demonstra algumas mensagens recebidas de um cliente.

Figura 34 - Mensagens de um cliente

qual endereço da empresa	#LocalizaçãoEmpresa	No entities
29/10/2017 @ 20:10	<a href="#">Open conversation</a>	
gostaria de saber sobre o curso de tanatopraxia	#PerguntasCurso	@ProdutosTanatopraxia:Tana...
29/10/2017 @ 20:09	<a href="#">Open conversation</a>	
Qual função TA14 ?	#UtilizaçãoProdutos	@ProdutosQuímicos:TA14
29/10/2017 @ 20:09	<a href="#">Open conversation</a>	
Olá boa noite	#InicioConversa	No entities
29/10/2017 @ 20:09	<a href="#">Open conversation</a>	

Fonte: Elaborada pelo autor.

Neste exemplo, o *chatbot* detectou corretamente as Intenções e Entidades, respondendo de forma adequada sobre a utilização do produto químico TA14, o curso de tanatopraxia e o endereço da empresa. A Figura 35 mostra outras dúvidas dos clientes.

Figura 35 - Outras dúvidas dos clientes

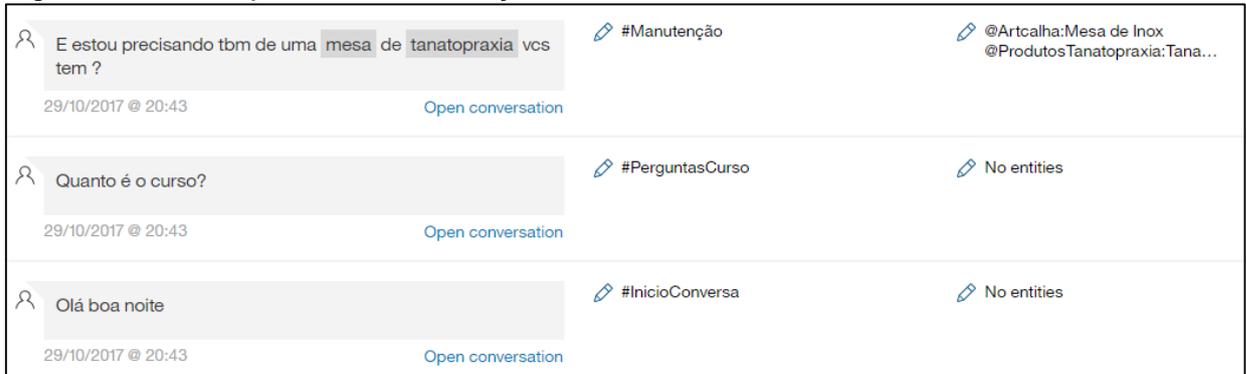
Qual o preço do ta14 ?	#PerguntasPreço	@ProdutosQuímicos:TA14
29/10/2017 @ 20:10	<a href="#">Open conversation</a>	
Vcs trabalham com Whatsapp?	#Whatsapp	No entities
29/10/2017 @ 20:10	<a href="#">Open conversation</a>	
Vcs tem ta26 ?	#VerificaçãoProdutos	@ProdutosQuímicos:TA26
29/10/2017 @ 20:10	<a href="#">Open conversation</a>	

Fonte: Elaborada pelo autor.

Neste caso, o *chatbot* identificou de forma correta as Intenções e Entidades, respondendo sobre a disponibilidade do produto químico TA26, o uso de Whatsapp da empresa e o preço do produto químico TA14.

Também há casos em que o *chatbot* não detecta a Intenção ou Entidade corretamente, como está demonstrado na Figura 36.

Figura 36 - Exemplo de uma detecção incorreta



Fonte: Elaborada pelo autor.

Neste exemplo, o *chatbot* detectou corretamente as Intenções “#InicioConversa” e “#PerguntasCurso”, mas a Intenção “#Manutenção” está incorreta. Portanto, é necessário realizar um processo de refinamento da base de conhecimento, clicando no ícone de “lápis”. Com isto, é aberta uma área para editar a Intenção, como mostra a Figura 37.

Figura 37 - Área para editar a Intenção



Fonte: Elaborada pelo autor.

Neste caso, foi selecionada a Intenção correta, que é “#VerificaçãoProdutos”. Assim, o Watson efetua um treinamento, adicionando a mensagem como um exemplo à Intenção “#VerificaçãoProdutos”. Com isto, há o processo de refinamento e a detecção do *chatbot* é melhorada.

## 9 CONSIDERAÇÕES FINAIS

Este trabalho demonstrou o processo de desenvolvimento de um *chatbot* empresarial de relacionamento com o cliente, a fim de tirar dúvidas sobre os assuntos importantes referentes à empresa como detalhes de produtos, informações sobre cursos, localização, horário de atendimento, assessoria, exportação, dúvidas técnicas, entre outros. Com isto, o processo de atendimento foi melhorado e foi promovido um contato maior da empresa com os clientes.

O IBM Watson é uma ferramenta atual excelente para o desenvolvimento de *chatbots*, com inúmeras funcionalidades e possibilidades. Além disto, o mesmo pode ser integrado com várias plataformas, permitindo que as empresas tenham um contato maior com os clientes. A capacidade de aprendizado do IBM Watson é um ponto a ser destacado, já que o mesmo pode se adaptar a inúmeros contextos, além de entender muitos idiomas, possibilitando que clientes estrangeiros possam se comunicar com as empresas.

Neste contexto, concluiu-se que a Inteligência Artificial pode ser aplicada em muitas áreas, facilitando inúmeros processos. No contexto empresarial, os *chatbots* podem auxiliar no setor de atendimento, respondendo aos clientes de forma rápida e eficiente. A capacidade de adaptação e aprendizado dos mesmos é outro fator relevante, já que adquirem mais conhecimento conforme as conversas com os clientes são realizadas.

Outro ponto importante é que a base de conhecimento do *chatbot* foi desenvolvida de acordo com as informações presentes no site da empresa. Também houve uma pesquisa sobre os assuntos mais procurados pelos clientes, com o setor de atendimento. Com isto, o *chatbot* foi capaz de responder as dúvidas dos clientes de forma eficiente. Assim, foi possível perceber que o contato com o cliente é essencial, para que o projeto seja desenvolvido de acordo com suas necessidades.

Além disto, o processo de refinamento da base de conhecimento aconteceu antes e após a integração com o Messenger do Facebook. Com isto, o *chatbot* pôde responder às dúvidas com eficiência quando foi integrado, além de expandir seus assuntos, com o surgimento de novas dúvidas dos clientes. Conseqüentemente, o atendimento da empresa foi melhorado, aprimorando muitos processos.

Como trabalhos futuros, os estudos poderão ser ampliados, algumas atividades que poderão ser desenvolvidas incluem:

- a) Integrar o *chatbot* com o novo site da empresa: para que os clientes possam interagir através de outra plataforma, além do Messenger do Facebook;
- b) Integrar o *chatbot* com o Telegram e outras plataformas: isto permite que os clientes possam se comunicar de várias formas, aumentando seu contato com a empresa;
- c) Acrescentar os idiomas inglês e espanhol: alguns clientes da empresa são de outros países, portanto, é interessante adicionar estes idiomas, permitindo a interação com novos clientes estrangeiros;
- d) Utilizar ferramentas de voz: o IBM Watson tem a possibilidade de trabalhar com voz, sendo algo atrativo. Com isto, os clientes teriam outra forma de comunicação;
- e) Criar mais interações do cliente com o *chatbot*: isto permite que a interação do cliente com o *chatbot* seja ampliada. Poderia ser fornecido um menu de opções referentes a algum assunto, que seria solicitado pelo cliente, para facilitar algum processo.

Por meio destas atividades, serão promovidas novas possibilidades de interação com o *chatbot*, expandindo o contato da empresa com os clientes e gerando um aumento na quantidade dos mesmos. Com isto, poderão surgir novas dúvidas, que deverão ser respondidas pelo *chatbot*. Neste contexto, será necessário dar continuidade ao processo de refinamento da base de conhecimento, incluindo novos assuntos abordados pelos clientes.

## REFERÊNCIAS

- AGUIAR, M. L. G. de. **Implementação de um atendente virtual especializado em licenciamento ambiental utilizando AIML**. 2013. 87 f. Trabalho de Conclusão de Curso (Graduação em Ciência da Computação) – Universidade do Sagrado Coração, Bauru, 2013.
- ALICE seguros. **Botsbrasil.com.br**, 2016. Disponível em: <<http://www.botsbrasil.com.br/alice-seguros/>>. Acesso em: 6 maio 2017.
- ANA. **Algartelem.com.br**, 2017. Disponível em: <<https://www.algartelem.com.br/para-voce/celular/4g>>. Acesso em: 7 maio 2017.
- BEN. **Botsbrasil.com.br**, 2016. Disponível em: <<http://www.botsbrasil.com.br/ben/>>. Acesso em: 7 maio 2017.
- CLOUDIA. **Botsbrasil.com.br**, 2016. Disponível em: <<http://www.botsbrasil.com.br/cloudia/>>. Acesso em: 6 maio 2017.
- COPPIN, B. **Inteligência artificial**. Rio de Janeiro: LTC, 2010.
- DIAS DA SILVA, B. C. et al. Introdução ao processamento das línguas naturais. **ICMC-USP** - Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo, 1999. Disponível em: <<http://www.nilc.icmc.usp.br/nilc/index.php/publications>>. Acesso em: 22 abr. 2017.
- ESTÚDIO ABC. 6 exemplos de como usar computação cognitiva. **Exame.abril.com.br**, 2016. Disponível em: <<http://exame.abril.com.br/tecnologia/6-exemplos-de-como-usar-computacao-cognitiva/>>. Acesso em: 14 maio 2017.
- FACELI, K. et al. **Inteligência artificial: uma abordagem de aprendizado de máquina**. Rio de Janeiro: LTC, 2011.
- FERREIRA, L. P.; UCHÔA, J. Q. Desenvolvimento de um chatbot para auxiliar o ensino de espanhol como língua estrangeira. **Bazar**, Lavras, n. 1, p. 21 – 32, mar. 2006. Disponível em: <[http://repositorio.ufla.br/bitstream/1/9629/1/ARTIGO\\_Desenvolvimento%20de%20um%20chatbot%20para%20auxiliar%20o%20ensino%20de%20espanhol%20como%20%C3%ADngua%20estrangeira.pdf](http://repositorio.ufla.br/bitstream/1/9629/1/ARTIGO_Desenvolvimento%20de%20um%20chatbot%20para%20auxiliar%20o%20ensino%20de%20espanhol%20como%20%C3%ADngua%20estrangeira.pdf)>. Acesso em: 5 maio 2017.
- GONZALEZ, M.; LIMA, V. L. S. de. Recuperação de informação e processamento da linguagem natural. **PUCRS** - Pontifícia Universidade Católica do Rio Grande do Sul, 2003. Disponível em: <<http://www.inf.pucrs.br/~gonzalez/docs/minicurso-jaia2003.pdf>>. Acesso em: 22 abr. 2017.
- GUDWIN, R. R. Engenharia de software: uma visão prática. **UNICAMP** – Universidade Estadual de Campinas, 2015. Disponível em: <<http://faculty.dca.fee.unicamp.br/gudwin/sites/faculty.dca.fee.unicamp.br/gudwin/files/ea975/ESUVP2.pdf>>. Acesso em: 13 maio 2017.

IBM. Watson conversation. **Ibm.com**, 2017. Disponível em: <<https://www.ibm.com/watson/developercloud/doc/conversation/index.html>>. Acesso em: 15 maio 2017.

IBM. Watson documentation. **Ibm.com**, 2017. Disponível em: <<https://www.ibm.com/watson/developercloud/doc/index.html>>. Acesso em: 15 maio 2017.

I FOOD pizzeria. **Botsbrasil.com.br**, 2016. Disponível em: <<http://www.botsbrasil.com.br/ifood-pizzaria/>>. Acesso em: 6 maio 2017.

LANDSTEINER, N. Eliza. **Masswerk.at**, 2005. Disponível em: <<http://www.masswerk.at/elizabot/>>. Acesso em: 6 maio 2017.

LATTARO, A. O que é o Watson e quais aplicações os desenvolvedores podem aproveitar. **Imasters.com.br**, 2016. Disponível em: <<https://imasters.com.br/desenvolvimento/o-que-e-o-watson-e-quais-aplicacoes-os-desenvolvedores-podem-aproveitar/?trace=1519021197&source=single>>. Acesso em: 14 maio 2017.

LAVEN, S. Chatterbot central. **Simonlaven.com**, 2003. Disponível em: <<http://www.simonlaven.com/>>. Acesso em: 5 maio 2017.

LEONEL, J. S. O deepQA. **Computacaocognitiva.wordpress.com**, 2015. Disponível em: <<https://computacaocognitiva.wordpress.com/2015/01/07/o-deepqa/>>. Acesso em: 14 maio 2017.

LEONHARDT, M. D. **Doroty**: um chatterbot para treinamento de profissionais atuantes no gerenciamento de redes de computadores. 2005. 110 f. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal do Rio Grande do Sul, Porto Alegre, 2005. Disponível em: <<http://www.lume.ufrgs.br/bitstream/handle/10183/5659/000473673.pdf>>. Acesso em: 6 maio 2017.

MONARD, M. C.; BARANAUSKAS, J. A. Conceitos sobre aprendizado de máquina. In: REZENDE, S. O. **Sistemas inteligentes**: fundamentos e aplicações. São Paulo: Manole, 2003. p. 39-56. Disponível em: <<http://dcm.ffclrp.usp.br/~augusto/publications/2003-sistemas-inteligentes-cap4.pdf>>. Acesso em: 13 maio 2017.

MR joboto. **Botsbrasil.com.br**, 2016. Disponível em: <<http://www.botsbrasil.com.br/mr-joboto/>>. Acesso em: 7 maio 2017.

MÜLLER, D. N. Processamento de linguagem natural. **UFRS** - Universidade Federal do Rio Grande do Sul, 2003. Disponível em: <<http://www.inf.ufrgs.br/~danielnm/docs/pln.pdf>>. Acesso em: 22 abr. 2017.

OLIVEIRA, V. T. de. **Desenvolvimento e implementação web de um chatterbot para educação ambiental utilizando AIML e PROLOG**. 2016. 110 f. Trabalho de

Conclusão de Curso (Graduação em Ciência da Computação) – Universidade do Sagrado Coração, Bauru, 2016.

PRESSMAN, R. S. **Engenharia de software**: uma abordagem profissional. 7. ed. Porto Alegre: AMGH, 2011.

RICH, E.; KNIGHT, K. **Inteligência artificial**. 2. ed. São Paulo: Makron Books, 1994.

ROSA, J. L. G. **Fundamentos da inteligência artificial**. Rio de Janeiro: LTC, 2011.

RUSSEL, S.; NORVIG, P. **Inteligência artificial**: tradução da terceira edição. 3. ed. Rio de Janeiro: Elsevier, 2013.

SILVA, A. B. da. **Um chatterbot em AIML plus que conversa sobre horóscopo**. 2002. 46 f. Trabalho de Conclusão de Curso (Graduação em Ciência da Computação) – Universidade Federal de Pernambuco, Recife, 2002. Disponível em: <<http://www.cin.ufpe.br/~tg/2002-1/>>. Acesso em: 5 maio 2017.

SILVA, H. da. **TeO**: um chatterbot para telessaúde. 2014. 102 f. Trabalho de Conclusão de Curso (Graduação em Ciência da Computação) – Universidade do Sagrado Coração, Bauru, 2014.

SIQUEIRA, R. de A. Robôs com inteligência artificial. **Ed.conpet.gov.br**, 2005. Disponível em: <[http://www.ed.conpet.gov.br/materias/2005\\_comciencia.php](http://www.ed.conpet.gov.br/materias/2005_comciencia.php)>. Acesso em: 5 maio 2017.

WAYPOINT. **Botsbrasil.com.br**, 2016. Disponível em: <<http://www.botsbrasil.com.br/waypoint/>>. Acesso em: 7 maio 2017.

WINSTON, P. H. **Artificial intelligence**. MA: Addison-Wesley, 1992. Disponível em: <<http://courses.csail.mit.edu/6.034f/ai3/rest.pdf>>. Acesso em: 11 maio 2017.