

**EM BUSCA DA PALAVRA MAIS ENGENHOSA:  
COGNIÇÃO E PROCESSAMENTO AUTOMÁTICO  
DE LÍNGUA NATURAL COM CODENAMES**

ROBERTA VIOLA

EM BUSCA DA PALAVRA MAIS ENGENHOSA:  
COGNIÇÃO E PROCESSAMENTO AUTOMÁTICO  
DE LÍNGUA NATURAL COM CODENAMES

Dissertação apresentada ao Programa de Pós-Graduação em Linguística Teórica e Descritiva da Universidade Federal de Minas Gerais como requisito parcial para a obtenção do grau de Mestre em Linguística Teórica e Descritiva.

ORIENTADOR: HELIANA MELLO & ADRIANO VELOSO

Belo Horizonte  
Setembro de 2021

V795re Viola, Roberta.  
Em busca da palavra mais engenhosa [manuscrito] : cognição e processamento automático de língua natural com Codenames / Roberta Viola. – 2021.  
106 f., enc.: il, tab., color.

Orientadora: Heliana Mello.

Coorientador: Adriano Veloso.

Área de concentração: Linguística.

Linha de Pesquisa: Linguística de Corpus.

Dissertação (mestrado) – Universidade Federal de Minas Gerais, Faculdade de Letras.

Bibliografia: f. 80-86.

Apêndices: f. 88-90.

1. Linguística – Processamento de dados – Teses. 2. Processamento de linguagem natural (Computação) – Teses. 3. Linguística – Teses. I. Mello, Heliana. II. Veloso, Adriano. III. Universidade Federal de Minas Gerais. Faculdade de Letras. IV. Título.

CDD: 410



UNIVERSIDADE FEDERAL DE MINAS GERAIS  
FACULDADE DE LETRAS  
PROGRAMA DE PÓS-GRADUAÇÃO EM ESTUDOS LINGUÍSTICOS

## FOLHA DE APROVAÇÃO

**Em busca da palavra mais engenhosa: cognição e processamento automático de língua natural com Codenames**

**ROBERTA QUEIROZ LESSA VIOLA**

Dissertação submetida à Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em ESTUDOS LINGUÍSTICOS, como requisito para obtenção do grau de Mestre em ESTUDOS LINGUÍSTICOS, área de concentração LINGUÍSTICA TEÓRICA E DESCRITIVA, linha de pesquisa Estudos Linguísticos Baseados em Corpora.

Aprovada em 15 de outubro de 2021, pela banca constituída pelos membros:

Prof(a). Heliana Ribeiro de Mello - Orientadora  
UFMG

Prof(a). Adriano Alonso Veloso - Coorientador  
UFMG

Prof(a). Tommaso Raso  
UFMG

Prof(a). Valeria Correa Vaz de Paiva  
Topos Institute, Berkeley, California

Belo Horizonte, 15 de outubro de 2021.



Documento assinado eletronicamente por **Heliana Ribeiro de Mello, Professora do Magistério Superior**, em 15/10/2021, às 20:04, conforme horário oficial de Brasília, com fundamento no art. 5º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Tommaso Raso, Professor do Magistério Superior**, em 15/10/2021, às 20:10, conforme horário oficial de Brasília, com fundamento no art. 5º do [Decreto nº 10.543, de 13 de novembro de 2020](#).

Documento assinado eletronicamente por **Valeria de Paiva, Usuário Externo**, em 18/10/2021, às



10:42, conforme horário oficial de Brasília, com fundamento no art. 5º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Adriano Alonso Veloso, Professor do Magistério Superior**, em 03/11/2021, às 21:59, conforme horário oficial de Brasília, com fundamento no art. 5º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



A autenticidade deste documento pode ser conferida no site [https://sei.ufmg.br/sei/controlador\\_externo.php?acao=documento\\_conferir&id\\_orgao\\_acesso\\_externo=0](https://sei.ufmg.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0), informando o código verificador **0977899** e o código CRC **B96DAD1C**.

# Agradecimentos

aos guias acadêmicos brilhantes que tive o privilégio de encontrar: Heliana e Adriano. Muito obrigada pela troca intelectual, pela confiança e pelo cuidado atencioso; pelo *delicado essencial*, enfim.

ao meu pai, por me ensinar a desembolar a vida.

à minha mãe, pelo exemplo de bravura e coragem.

à minha avó, pelo melhor dos presentes: a literatura, que nos dá sempre mais possibilidades.

aos três pelo exemplo de paixão aos trabalhos a que se dedicaram.

à minha irmã, minha melhor parceria para a vida.

a todos que eu amo, por me permitirem o aprendizado e o exercício que o amor solicita.

ao Aikido, sem essa prática tudo seria menos harmonioso.

aos meus cachorros, pelo aconchego.

aos meus amigos, por seguirem firmes na crença de que brincadeira é coisa séria.

ao Felipe e ao Guilherme, pela troca durante toda a pesquisa, e pela colaboração ativa no desenvolvimento do modelo computacional.

à Tayane pela revisão e leitura generosa.

a todas as colegas da Faculdade de Letras, pela companhia nos cafés, nas risadas e conversas desinteressadas que nos aquecem o cotidiano.

à Maisa, pelo sonho, pela paciência, pelo amor, pela sabedoria, por ser a base de muito.

*“Outras palavras  
Para dizer certas coisas  
são precisas  
palavras outras  
novas palavras  
nunca ditas antes  
ou nunca  
antes  
postas lado a lado  
”*

(Marina Colasanti)

# Resumo

O jogo *Codenames* exige que um jogador forneça uma palavra como dica para outras duas ou três palavras no tabuleiro e, assim, os outros jogadores devem adivinhar as palavras-alvo. Essa dica deve ser *A* palavra: específica, de modo a apontar seus alvos e excludente, de modo a excluir as palavras não intencionadas. O nosso trabalho propõe um modelo computacional, adaptado de [Koyyalagunta et al., 2021], para a tarefa, completamente em português brasileiro. Nossos agentes, desenvolvidos para as funções de *Codemaster* e *Guesser*, alcançaram resultados competitivos e forneceram exemplos interessantes de dicas para o jogo. As pistas fornecidas e os processos cognitivos envolvidos no jogo também foram analisados por nós.

# Abstract

Codenames is a game that demands one player to give a word as a clue that relates to other two or three words on the board, and other players to guess the target words. This clue has to be THE word, an specific one that comprises the targets and excludes the bad words on the board. We propose an adapted model of [Koyyalagunta et al., 2021] for that task, in Brazilian Portuguese. Our agents, developed for the role of Codemaster and Guesser, achieved competitive win rates and interesting examples of clues for the game. The given clues and the cognitive processes involved on the game were analyzed by us.

# Lista de Figuras

2.1	Exemplo de tabuleiro para <i>Codenames</i> . . . . .	7
2.2	<i>Codenames Dueto</i> . Na carta à esquerda, as posições marcadas em verde indicam as palavras-alvo . . . . .	8
3.1	Fluxograma de Aprendizado Supervisionado para <i>Codenames</i> . . . . .	14
3.2	Tirinha Mafalda, fonte: <a href="https://linktr.ee/mafaldadigital">https://linktr.ee/mafaldadigital</a> acesso em 22 fev. 2021 . . . . .	15
3.3	Pseudocódigo para receita de bolo em <i>Python</i> . . . . .	16
3.4	Ilustração do Aprendizado de Máquina a partir de exemplos . . . . .	19
3.5	Ilustração simplificada de um processo de interpretação do português com um <i>output</i> que não foi explicitamente solicitado e que se adequa perfeitamente à fala . . . . .	21
3.6	Ilustração de Espaço Vetorial . . . . .	22

# Lista de Tabelas

5.1	Jogos entre dois jogadores humanos . . . . .	40
8.1	Exemplos Gerados Computacionalmente para 2 Palavras . . . . .	67
A.1	Jogos entre dois jogadores humanos . . . . .	89

# Lista de Abreviaturas

**pt-BR** Português brasileiro

**NLP** Processamento de Língua Natural (*Natural Language Processing*)

**ML** Aprendizado de Máquina (*Machine Learning*)

**IA** Inteligência Artificial

**ToM** *Theory of Mind*

# Sumário

Agradecimentos	v
Resumo	viii
Abstract	ix
Lista de Figuras	x
Lista de Tabelas	xi
Lista de Abreviaturas	xii
<b>1 Prefácio</b>	<b>1</b>
<b>2 Introdução</b>	<b>3</b>
2.1 Motivação . . . . .	5
2.2 <i>Codenames</i> . . . . .	6
2.2.1 <i>Codenames AI Competition</i> . . . . .	9
2.3 Cognição e <i>Codenames</i> . . . . .	9
2.4 Objetivos . . . . .	11
<b>3 <i>Background</i> e Trabalhos Relacionados</b>	<b>12</b>
3.1 <i>Background Computacional</i> . . . . .	12
3.1.1 Algoritmos . . . . .	13

3.1.2	Aprendizado de Máquina . . . . .	17
3.1.3	Processamento de Linguagem Natural . . . . .	20
3.1.4	<i>Word Embeddings: You shall know a word by the company it keeps!</i> [Firth, 1957] . . . . .	21
3.2	Trabalhos Relacionados . . . . .	22
<b>4</b>	<b>Pressupostos Teóricos</b>	<b>27</b>
4.1	Palavras x Léxico . . . . .	27
4.2	Associações Lexicais . . . . .	28
4.3	Insights . . . . .	31
4.4	Representações Mentais e <i>Mindreading</i> . . . . .	33
4.5	Protótipos Lexicais . . . . .	35
<b>5</b>	<b>Jogos entre Humanos</b>	<b>37</b>
5.1	Dados de jogos entre humanos . . . . .	38
5.2	Análises das Escolhas dos Jogadores . . . . .	41
5.2.1	Dica: Jataí . . . . .	42
5.2.2	Dica: Cinderela . . . . .	42
5.2.3	Dica: mistério . . . . .	44
5.2.4	Dica: climática . . . . .	44
5.2.5	Dica: arranha . . . . .	45
5.2.6	Dicas: água, polícia e acampamento . . . . .	45
5.2.7	Dica: coroa . . . . .	46
5.2.8	Dica: chocolate . . . . .	47
5.3	Conclusões sobre os resultados . . . . .	48
<b>6</b>	<b><i>Natural Language Understanding</i></b>	<b>50</b>
6.1	Algumas reflexões em NLU . . . . .	50
6.2	NLU em <i>Codenames</i> . . . . .	52

<b>7</b>	<b>Modelo Computacional</b>	<b>56</b>
7.1	Estrutura do Modelo Computacional . . . . .	57
7.2	Insumos Linguísticos . . . . .	58
7.2.1	Corpus para o Modelo Computacional . . . . .	58
7.2.2	Tabuleiro . . . . .	59
7.2.3	Possíveis Dicas . . . . .	59
7.2.4	Dict2vec . . . . .	60
7.2.5	DETECT . . . . .	62
7.3	Codemaster . . . . .	63
<b>8</b>	<b>Resultados do Modelo Computacional</b>	<b>65</b>
8.1	Exemplos . . . . .	66
8.1.1	Exemplos para duas palavras-alvo . . . . .	66
8.1.2	Exemplos para três palavras-alvo . . . . .	69
8.2	Análises dos Resultados . . . . .	70
8.2.1	Dica: Maria . . . . .	71
8.2.2	Dica: funil . . . . .	71
8.2.3	Dica: sacristia . . . . .	72
8.2.4	Dica: fagote . . . . .	73
8.2.5	Dica: centroavante . . . . .	73
8.2.6	Dica: biscoitos . . . . .	73
8.2.7	Dica: bigode . . . . .	74
8.2.8	Dica: viatura . . . . .	75
8.2.9	Dica: unicórnio . . . . .	76
8.2.10	Dica: linguista . . . . .	76
<b>9</b>	<b>Conclusões</b>	<b>77</b>
9.0.1	Contribuições . . . . .	78
9.0.2	Trabalhos Futuros . . . . .	79

<b>Referências Bibliográficas</b>	<b>81</b>
<b>Apêndice A Apêndice</b>	<b>89</b>
A.1 Exemplos de Jogos entre humanos . . . . .	89

# Capítulo 1

## Prefácio

Como costumamos dizer, no jargão dos jogadores, o *Codenames* é um jogo que “vê mesa sempre”. Tenho o jogo desde 2015, quando foi lançado, e de lá pra cá, já joguei com pessoas diversas: de idades distintas, habituadas ou não aos jogos de tabuleiro, de cidades ou países diferentes, de gêneros diversos etc. Não consigo me lembrar de alguma vez em que alguém que dividiu a mesa comigo tenha dito algo como: “esse jogo é chato, não gostei”. É verdade que algumas pessoas acham muito mais difícil que outras, mas de “chato”, nunca vi alguém adjetivar. Desconfio do viés que eu possa causar nas mesas em que participo - como gosto muito do jogo, poderia influenciar a avaliação dos meus companheiros - por isso, fiz uma rápida busca na Ludopedia<sup>1</sup> (uma enciclopédia da *web* sobre jogos) e verifiquei que as avaliações e as notas dada ao jogo pelos usuários, são condizentes com o que tenho testemunhado. A criatividade dos jogadores, as soluções improváveis que acabam funcionando muito bem e, principalmente, o quanto todos parecem se divertir, como linguista, me intrigaram. Um jogo tão simples: adivinhar palavras a partir de uma outra palavra que serviria de guia associativo, só isso, se mostra capaz de despertar tamanho potencial de soluções. As duas perguntas que me ocorriam também eram (aparentemente) simples: “qual é o mecanismo cognitivo que essas pessoas estão

---

<sup>1</sup><https://www.ludopedia.com.br/>

empregando aqui? é puramente linguístico, afinal?" e em seguida: "por que essa tarefa é tão recompensadora?", pois estava claro que a *tarefa* específica de *Codenames* era animadora, muito mais do que a competição ou a ludicidade presente em qualquer jogo.

Ao mesmo tempo, trabalhando com alguns modelos computacionais, desenvolvidos a partir de técnicas de Processamento da Linguagem Natural, no Laboratório de Inteligência Artificial da UFMG, eu me perguntava se de algum modo seria possível implementar um algoritmo capaz de simular o que os meus amigos faziam, e se esse modelo computacional poderia, quem sabe, se tornar tão bom quanto eles. Foi assim que propus aos meus orientadores, e tive sorte, porque eles toparam, embarcaram na viagem, apoiaram, e pudemos nos divertir também ao desenvolver este trabalho.

# Capítulo 2

## Introdução

O conhecimento de uma palavra está intimamente relacionado ao conhecimento sobre a posição que esta palavra ocupa na rede de associações com as outras palavras da língua. Este é também o conhecimento principal envolvido no jogo de tabuleiro *Codenames*<sup>1</sup>, que propõe que os agentes cooperem e, a partir de associações lexicais, produzam boas dicas para conectar palavras dispostas em um tabuleiro. O principal objetivo é fazer com que o outro jogador possa recriar associações formuladas pelo *Codemaster* (o jogador que dá a dica) ao tentar conectar diferentes conceitos, considerando que a dica fornecida procura ser tão precisa quanto possível e tão excludente quanto necessário. Ao fim das rodadas do jogo *Codenames* é frequente que os jogadores perguntem uns aos outros sobre os motivos e caminhos para se chegar à determinada dica ou palavra-alvo. O desafio cognitivo está muito evidente no jogo e, dentre os processos envolvidos, o processamento linguístico tem protagonismo, já que a principal tarefa consiste em fazer associações lexicais.

A estrutura do jogo apresenta diferentes processos que podem levar à melhor resposta, tornando-o um desafio relevante para modelos computacionais que se propõem a processar a linguagem natural. Sobretudo pela falta de contexto oferecido à interpretação, o modelo idealmente deveria ser capaz de lidar com problemas como

---

<sup>1</sup><https://codenames.game/>

ambiguidade, frequência de uso, protótipos lexicais, *background*, conhecimento sobre as experiências sensoriais, relações fonológicas, testemunho de textos multimodais, *insights*, *mindreading* etc, que, dentre outras, são algumas das estratégias observadas entre os jogadores humanos.

Na última década, contamos com grandes avanços na modelagem artificial da linguagem, mas ainda há muitos problemas e especificidades a serem tratados, principalmente para o português brasileiro (pt-BR), que carece de estudos e recursos na área. Hoje, os modelos computacionais de processamento de linguagem natural são capazes de gerar soluções satisfatórias e congruentes com as dos falantes em alguns casos, para outros, são geradas soluções divergentes das observadas entre os falantes, mas igualmente funcionais. No entanto, quando nos deparamos com alguns pontos chave, como quando há um fenômeno pragmático, metáforas ou inferências em questão, os modelos computacionais tendem a enfrentar alguns problemas.

Em nossa pesquisa, o estudo das associações lexicais se combina também a noções da linguística cognitiva, como os fenômenos pragmáticos envolvidos nos atos de fala conforme [Austin, 1962]. Ou seja, consideramos que as ações linguísticas alteram o estado no mundo e podem promover novas conexões e relações cognitivas entre conceitos que estavam antes desligados um do outro. Além disso, entendemos que, para estruturar e compreender o mundo, nos apoiamos em conceitos que têm uma estrutura prototípica comum, como postula [Cruse, 2014].

Tendo em vista os preceitos de [Sperber & Wilson, 1995], entendemos que os falantes, dada uma informação parcial, tomam decisões a partir do princípio da relevância.

Assim, no jogo, esse princípio seria essencial para o pacto entre os jogadores na tarefa de identificar o traço, o significado, ou o uso que as três palavras compartilham.

Consideramos ainda que, como nas interações de comunicação humana, os jogadores não minimizam o seu processamento pelo custo de demanda, já que estão submetidos a um contexto de competição, desafio e de cooperação entre os agentes. Assim, consideramos que os jogadores estão dispostos a disponibilizar todos os recursos da

atividade mental: pensamento, recuperação de informações na memória de longo prazo, formulação de conclusões etc, como observado por [Sperber & Wilson, 1995]. Essa dissertação se debruça sobre a tentativa de compreender alguns dos processos cognitivos e estratégias utilizadas pelo jogadores no jogo de tabuleiro *Codenames* para, em seguida, desenvolver um modelo computacional satisfatório para a tarefa do mesmo jogo, a fim de contribuir com recursos de NLP para o pt-BR.

## 2.1 Motivação

Como disciplina de métodos científicos e experimentais, a linguística é uma área de estudos relativamente nova, e pela complexidade de seu objeto de estudo, frequentemente, desliza sobre as pretensas fronteiras existentes entre os estudos da cognição.

Acreditamos que o jogo que utilizaremos como pretexto para a simulação e a análise dos resultados humanos e artificiais podem nos fornecer uma boa aproximação de alguns mecanismos cognitivos de entendimento crucial não só para os estudos linguísticos, como para um desenvolvimento mais satisfatório dos modelos de Aprendizado de Máquina (doravante ML), que costumam falhar nos limites de fenômenos linguísticos mais complexos e não lineares, como os que envolvem a pragmática, metáforas, ambiguidades etc. Assim, pretendemos, com o nosso trabalho, a partir da observação da interação humano x humano no jogo *Codenames*, e de alguns pressupostos teóricos, contribuir para o desenvolvimento de um modelo computacional que opere de forma automática a tarefa do jogo *Codenames*. Em seguida, verificamos o que funciona em nosso modelo e, principalmente, por que funciona. Em modelos de ML, a preocupação maior costuma convergir para melhorar o desempenho, centrando os esforços em quão eficiente, em termos quantitativos, o modelo é. Acreditamos que esta seja uma etapa crucial: desenvolver arquiteturas que executem bem a tarefa específica para a qual foram projetadas. No entanto, nos

parece ainda mais interessante investigar os caminhos e propósitos desse desempenho, em busca de intuições que nos digam algo sobre a cognição humana e possibilitem, conseqüentemente, o aprimoramento das técnicas de aprendizado de máquina.

Muitos jogos, principalmente os competitivos, de soma zero, fazem parte da história da computação, mas *Codenames* se diferencia por sua natureza cooperativa e por utilizar a língua como matéria: a interação harmônica entre os parceiros é crucial, o que lança interessantes desafios para a modelagem computacional e para a análise linguística.

## 2.2 *Codenames*

O jogo de tabuleiro *Codenames* foi publicado pelo *designer* Vlaada Chvátil em 2015. No jogo, o jogador denominado *Guesser* deve adivinhar um conjunto de duas ou mais palavras a partir da dica dada por outro jogador, chamado de *Codemaster*. Para dar a dica, o *Codemaster* deve se limitar a dizer uma só palavra, e esta deve ser divergente de todas as palavras que já se encontram dispostas no tabuleiro. O tabuleiro do jogo é composto por palavras em uma disposição de 5x5, totalizando 25. Cada um dos jogadores recebe 9 palavras-alvo, dentre as quais 3 delas são em comum com o outro jogador, e 1 palavra proibida, denominada *Assassino*. A figura a seguir ilustra um tabuleiro reduzido, com uma situação típica de jogo:



**Figura 2.1.** Exemplo de tabuleiro para *Codenames*

Na Fig.2.2, temos um exemplo simplificado de um tabuleiro para *Codenames*, em que as palavras em azul devem ser adivinhadas e as em vermelho, evitadas.

As três palavras a serem adivinhadas, ou palavras-alvo, estão em azul, as palavras em vermelho devem ser evitadas na conexão de significados pretendida, pois, caso figurem na tentativa dos jogadores em adivinhar, configuram um erro. As palavras-alvo podem ser adivinhadas todas de uma só vez ou, gradualmente, nas rodadas subsequentes. No entanto, quanto mais rodadas se leva para adivinhar as palavras, menor é a pontuação obtida ao final do jogo. Assim, quanto mais palavras a dica do *Codemaster* for capaz de fazer o *Guesser* adivinhar, mais vantajosa será essa dica. O *Guesser* é alertado sobre o número de palavras que deve adivinhar. No exemplo acima, o *Codemaster* poderia dizer algo como: *Casa* - 3, indicando que *Casa* é a dica para 3 palavras-alvo. Portanto, no exemplo, a tarefa do *Codemaster* consiste

em encontrar uma palavra que sirva como *input* para *almofada*, *madeira* e *sótão* e que, ao mesmo tempo, não ative as outras palavras dispostas no tabuleiro, de modo a minimizar as chances de erro.

Apresentamos aqui uma fotografia do jogo *Codenames Dueto*, que é a versão do jogo para dois jogadores. Há uma outra versão disponível para grupos de mais jogadores, a versão *dueto*, que é cooperativa, é a que utilizamos em nosso trabalho. A carta à esquerda na fotografia indica as palavras-alvo (em verde) e as palavras proibidas (*assassinos*) em preto.



**Figura 2.2.** *Codenames Dueto*. Na carta à esquerda, as posições marcadas em verde indicam as palavras-alvo

Chamamos atenção para a classificação indicativa do jogo: maiores de 11 anos, o que está bem alinhado com a competência lexical exigida. Ressaltamos que há alguns outros detalhes que aprimoram a experiência lúdica do jogo, mas para o nosso trabalho nos apoiaremos na dinâmica básica que descrevemos, com o intuito de simplificar a tarefa e gerar percepções mais explícitas sobre os resultados.

Disponibilizamos aqui um vídeo com a explicação mais detalhada das regras do jogo<sup>2</sup> e um link para o manual de instruções<sup>3</sup>.

<sup>2</sup><https://www.youtube.com/watch?v=C8NKRosF6Vo>

<sup>3</sup><https://czechgames.com/files/rules/codenames-duet-rules-en.pdf>

### 2.2.1 Codenames AI Competition

A *Codenames AI Competition*<sup>4</sup> testa a compreensão da linguagem e a comunicação humana por modelos de Inteligência Artificial (IA), utilizando o jogo *Codenames* como base de teste. Essa competição foi organizada, em 2019, por Adam Summerville, Andrew Kim, Maxim Ruzmaykin e Aaron Truong. Os organizadores justificam o interesse em lançar a competição pelo fato de o jogo oferecer diferentes facetas, as quais tornam o desafio instigante, além de ser um ótimo teste para os sistemas de processamento de linguagem natural: primeiramente pela falta de contexto para as dicas e palavras-alvo, o que significa que o sistema deve ser capaz de gerar um entendimento amplo das conexões lexicais. Além disso, enfatizam a natureza não simétrica do jogo, que requer dois tipos de agentes com habilidades distintas: *Guesser* e *Codemaster*, este que recebe a tarefa mais desafiadora, ao ter que escolher uma dica dentre todas as palavras disponíveis no vocabulário da língua, que faça o outro jogador interpretar bem suas intenções.

Todos os detalhes, regras, recursos e regulamento para a competição podem ser encontrados em seu site. Ressaltamos que todo o material disponível e os projetos inscritos utilizam a língua inglesa, para a qual há um número de recursos computacionais disponíveis imensamente maior do que temos para o pt-BR. Ainda assim, pretendemos trabalhar com o pt-BR, por acreditarmos ser necessário desenvolver e aprimorar recursos para outras línguas além do inglês.

## 2.3 Cognição e *Codenames*

[Cann et al., 2006] propõe uma visão da linguagem como um quebra cabeças: cada palavra e seus significados, combinada às palavras vizinhas e seus respectivos significados, formam peças cada vez maiores, que se encaixam para formar uma sentença e seu significado geral, que emerge da combinação das partes. Nesse sentido,

---

<sup>4</sup><https://sites.google.com/view/the-codenames-ai-competition>

considerando a lógica reversa para a definição, temos como postulado que jogar *Codenames* trata-se de desconstruir esse quebra-cabeças linguístico a que somos frequentemente expostos. No jogo, o que interessa é “desencaixar” as peças e hipotetizar sobre as várias possibilidades de “encaixe” e associação para o léxico. Ao interpretar sentenças, o falante está constantemente submetido à tarefa de selecionar, dentre os múltiplos significados de uma palavra, aquele que melhor se encaixa no contexto maior em que está inserido. Não havendo contexto, como em nosso jogo, a tarefa se torna desintegrativa, ou seja, não há uma sentença inteira que guiará o entendimento de cada uma das palavras que a compõe, mas são as próprias palavras que deverão ter seus significados expandidos, mapeados e, por fim, o jogador deverá fazer as associações mais oportunas. Um bom exemplo do que acabamos de descrever é a conexão, que em nossos dados coletados, os jogadores puderam fazer entre as palavras “oásis”, “labirinto” e “sono”: a partir da dica “confuso”, foi possível estabelecer a conexão pelo sentido de desordem, pelo caráter onírico que carregam os itens lexicais alvo. Assim, podemos observar que variadas relações linguísticas, como a meronímia, relações hierárquicas, como hipo-nímia e hiperonímia, sinonímia e antonímia, inferências e referências à cultura são muito relevantes no jogo, tornando crucial para a nossa pesquisa os conceitos da Teoria dos Protótipos Lexicais [Cruse, 2014].

Além disso, nos embasamos nos estudos do pesquisador John Kounais, que investiga o fenômeno dos *insights*. Em seu artigo, *The Cognitive Neuroscience of Insight and its Antecedents* [Kounios & Beeman, 2014], os autores apontam para o fato de que a maioria de nós está familiarizado com o fenômeno, que poderia ser descrito como uma solução ou compreensão repentina de um problema ou uma situação, sob um novo ponto de vista. E afirma que essas experiências são poderosas, justamente por surgirem inesperadamente e reorganizarem os pensamentos e percepções dos indivíduos. No entanto, apesar desses *insights* aparentarem uma desconexão com o *input*, ou seja, de aparentarem surgir sem uma motivação lógica, o autor desenvolveu

estudos de neuroimagem que puderam demonstrar que os *insights* são a culminação de uma série de processos e estados mentais.

As teorias aqui expostas serão exploradas em nossa pesquisa e nos guiarão na análise dos exemplos de jogos entre jogadores humanos que coletamos e, em seguida, na definição de alguns parâmetros para a modelagem computacional.

## 2.4 Objetivos

Com o presente trabalho, buscamos desenvolver um modelo de ML que desempenhe com qualidade a tarefa de associação demandada pelo jogo *Codenames* e que, quando analisado e comparado com o processo humano de soluções para tarefas semelhantes, nos dê a oportunidade de levantar hipóteses sobre os processos cognitivos envolvidos e os limites da modelagem artificial. Está claro que, fenômenos pragmáticos relacionados ao *background* dos jogadores, *insights*, associações por traços semânticos, entre outros, são frequentemente relacionados às soluções humanas para o jogo. Soluções estas que o estado da arte do ML hoje, não consegue apreender suficientemente. Aqui, os limites dos modelos computacionais esbarram nos limites do nosso entendimento sobre o conhecimento linguístico e, principalmente, sobre os mecanismos das associações cognitivas.

Em nosso trabalho, analisamos como as pequenas partes da linguagem, ou seja, as palavras isoladamente, operam seus significados sem a correlação contextual ou sentencial, considerando, ao mesmo tempo, que, se os falantes testemunham palavras diferentes se combinando de formas diferentes, há a criação de uma relação cognitiva entre os significados. A base para a nossa análise são os dados de jogos humanos e computacionais.

Em nossas análises, indicamos os pontos em que a comparação dos dados computacionais e humanos nos permitiu aprimorar o nosso modelo computacional e refletir sobre os processos que estão em jogo.

## Capítulo 3

### *Background* e Trabalhos

### Relacionados

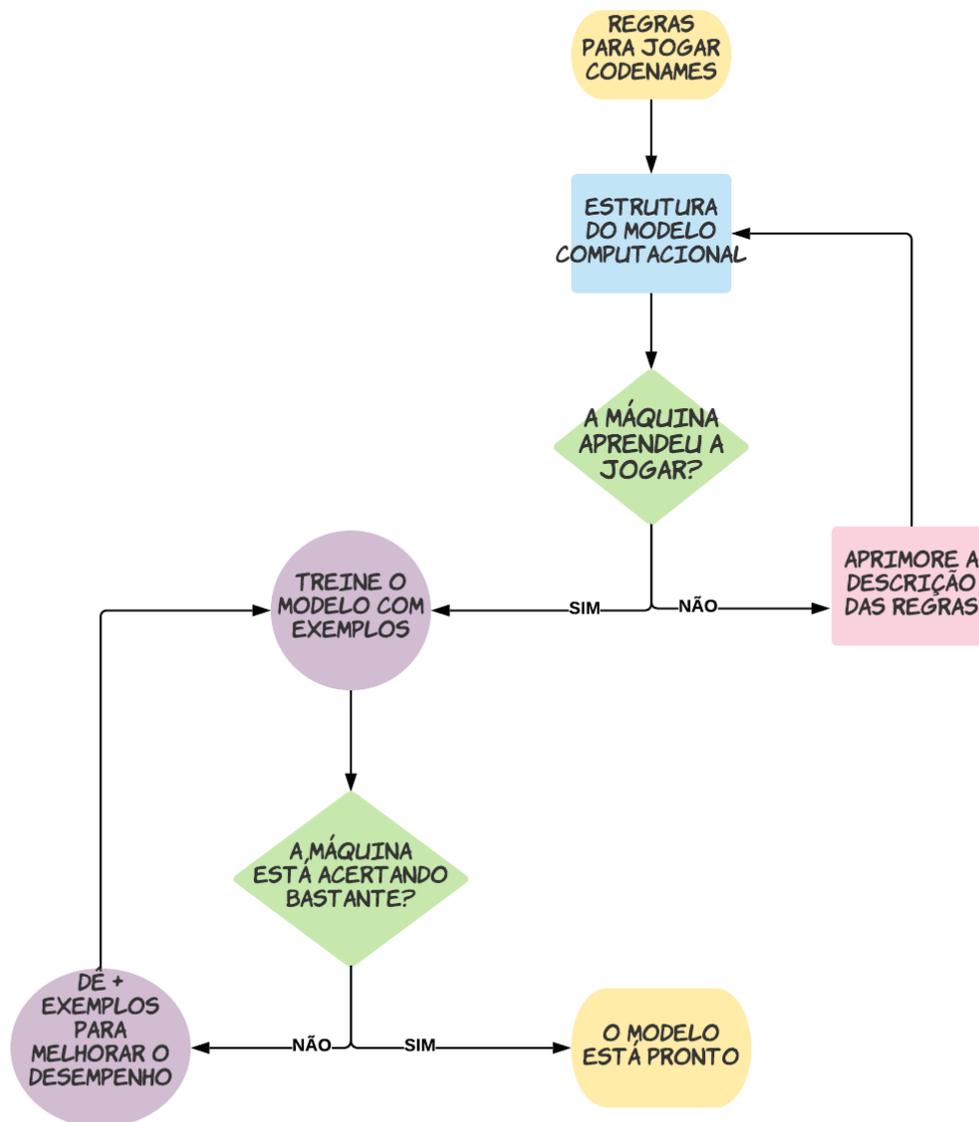
Nosso trabalho é essencialmente interdisciplinar, conectando pelo menos duas disciplinas: a Computação - mais especificamente a subárea da computação chamada de Aprendizado de Máquina - e a Linguística. Desse modo, trataremos de trabalhos que se relacionam ao nosso e que se subscrevem nessas disciplinas, além da psicologia e outros estudos interessados nos mecanismos da cognição humana.

#### 3.1 *Background Computacional*

Os passos para a criação de um modelo de Aprendizado de Máquina, como o que nos interessa aqui, incluem a aplicação de uma representação vetorial para as palavras, a definição de uma base de dados e de um algoritmo a ser implementado. A seguir, nos debruçaremos sobre a compreensão de alguns conceitos e ferramentas utilizadas na computação.

### 3.1.1 Algoritmos

Um algoritmo é uma sequência finita de regras, formuladas para executar uma tarefa específica. Cada uma dessas regras pode constituir uma expressão com condições determinadas por operadores lógicos, e decisões serão tomadas de acordo com os dados recebidos. Uma boa ilustração de um algoritmo é um fluxograma, em que, se determinadas condições são atendidas, o passo seguinte é executado, caso contrário, um passo diferente será executado. Abaixo, apresentamos um fluxograma com uma simplificação de como seria o processo de aprendizado de um modelo computacional supervisionado, ou seja, com exemplos de jogos, para *Codenames*.



**Figura 3.1.** Fluxograma de Aprendizado Supervisionado para *Codenames*

Todo processo com etapas estabelecidas pode ser expresso em um algoritmo, ou em um fluxograma. Esse modo de operar é ironizado pelo cartunista Quino:



**Figura 3.2.** Tirinha Mafalda, fonte: <https://linktr.ee/mafaldadigital> acesso em 22 fev. 2021

Um outro exemplo comum para ilustrar um algoritmo é uma receita de bolo, ela possui todos os ingredientes necessários (o que equivale aos dados) e uma sequência correta de ações para utilizá-los, de modo a alcançar o resultado final: o bolo - com boa aparência, consistência e sabor. Para que a receita tenha instruções mais completas, podemos pormenorizar os ingredientes e as etapas de execução. Podemos, ainda, alterar alguns processos ou ingredientes para mudar um pouco o sabor do bolo, sem que este perca o que o caracteriza como bolo.

Os algoritmos funcionam de forma análoga. Os algoritmos, para que uma máquina aprenda determinada tarefa, podem ser escritos em diversas linguagens de programação, cada uma com suas particularidades e finalidades. Neste trabalho, utilizamos uma linguagem de programação muito popular para trabalhar com ML: o *Python*. Cada linguagem de programação possui uma sintaxe própria para elaborar as ações e, também, ferramentas próprias. As ferramentas para implementar modelos de IA foram muito desenvolvidas em *Python*, popularizando diversos algoritmos, que estão prontos e disponíveis para a comunidade e que podem ser manipulados (adicionando mais instruções e alterando alguns passos) para obter diferentes resultados e adequar ao trabalho específico a que servirá.

Na figura seguinte, elaboramos um pseudocódigo (que pode ter seu sentido apreendido por um humano, mas por não estar totalmente em linguagem de programação, não

pode ser compilado pela máquina) para mostrar como funcionaria (hipoteticamente) uma receita de bolo, emulando a linguagem de programação *Python*.

```
: from armário_da_cozinha import ingredientes

farinha = Farinha('xícaras', 3)
leite = Leite('xícara', 1)
fermento_pô = Fermento('colher_sopa', 1)
açúcar = Açúcar('xícaras', 3)
ovos = Ovos(3)
manteiga = Manteiga('colher_sopa', 4)
forma = Forma('redonda')

# Passo 1: Pré-aqueça o forno
forno = Forno()
forno.set(180, graus='C')

# Passo 2: Adicione os ingredientes a um recipiente
ingredientes = Recipiente()
ingredientes.add(farinha)
ingredientes.add(fermento)
ingredientes.add(açúcar)
ingredientes.add(manteiga)
ingredientes.add(leite)
ingredientes.add(ovos)

# O passo 3 (while) é um loop

    while not ingredientes.homogeneo:
        ingredientes.mix()

        if ingredientes.muito_consistente:
            ingredientes.add(leite)
        else:
            forma.add(farinha) #para untar
            forma.add(ingredientes)

forno.assar(forma, minutos=40)

#teste se a massa está grudando com um garfo
#outro loop do algoritmo

while forma.grudar_garfo:
    forno.assar.minutos++
else:
    print(bolo_pronto)
```

Figura 3.3. Pseudocódigo para receita de bolo em *Python*

### 3.1.2 Aprendizado de Máquina

De modo geral, poderíamos definir um tipo de aprendizado qualquer, inclusive o aprendizado de máquina, como uma mudança adaptativa no comportamento, causada pela experiência. Nos interessa que os computadores aprendam tarefas que sejam difíceis de serem diretamente programadas, assim, com o aprendizado, a máquina é capaz de programar a si mesma a partir de exemplos de *inputs* e *outputs*. Ou seja, escrevemos um programa parametrizado, e permitimos que o algoritmo de aprendizado encontre a combinação de parâmetros que melhor se aproxima da função ou do comportamento desejados.

O cientista da computação Arthur Samuel<sup>1</sup> cunhou o termo *Machine Learning* em 1959, definindo-o como o campo de estudo que possibilita que os computadores aprendam sem que sejam explicitamente programados para as tarefas determinadas. Já em 2006, outro cientista da computação, Tom Mitchell, nos ofereceu uma definição mais atualizada e formal:

*Podemos dizer que um programa de computador aprende com a experiência  $E$  a desenvolver uma tarefa  $T$ , para a qual tem uma medida de desempenho  $P$ , se seu desempenho em  $T$ , medido por  $P$ , melhora com a experiência  $E$ .<sup>2</sup> [Mitchell, 2006]*

O aprendizado de máquina pode ser utilizado na resolução de diversas tarefas, mas é especialmente útil para a resolução de problemas complexos, como o reconhecimento automático de objetos tridimensionais. Para esta tarefa não saberíamos exatamente como escrever o nosso algoritmo, pois não sabemos o que acontece precisamente em nossa mente ao processarmos as imagens. Mas ainda que soubéssemos, provavelmente teríamos que escrever um algoritmo extremamente complexo. Para executar as tarefas envolvidas no jogo *Codenames*, nos deparamos com a mesma barreira: não

---

<sup>1</sup><http://infolab.stanford.edu/pub/voy/museum/samuel.html>

<sup>2</sup>*A computer program is said to learn from experience  $E$  with respect to some task  $T$  and some performance measure  $P$ , if its performance on  $T$ , as measured by  $P$ , improves with experience  $E$ .*

conhecemos todos os processos cognitivos e linguísticos envolvidos, portanto, com o intuito de desenvolver um modelo automático para o jogo, os algoritmos de aprendizado de máquina são mais adequados que os programas computacionais desenvolvidos passo a passo.

Ou seja, ao trabalharmos com algoritmos de aprendizado, não escrevemos as etapas detalhadas para se chegar aos resultados que almejamos. Em vez disso, alimentamos o nosso modelo com muitos exemplos que especificam a saída correta para uma determinada entrada. Assim, o nosso algoritmo é capaz de produzir um programa a partir desses dados. O programa produzido por um algoritmo de aprendizado geralmente se difere muito de um programa computacional típico, escrito à mão, em termos de estrutura e lógica computacional. Os dados utilizados como exemplos iniciais para o modelo são chamados de dados de treino. Uma vez treinado, este modelo deve ser capaz de fazer generalizações e de funcionar igualmente bem quando apresentarmos novos casos, os chamados dados de teste.

A ilustração abaixo representa um pouco do processo que acabamos de descrever.



**Figura 3.4.** Ilustração do Aprendizado de Máquina a partir de exemplos

A partir do treino com alguns exemplos de rostos de perfil, quando exposto a um novo rosto (que não estava no dado de treino), o modelo computacional deve ser capaz de responder se aquilo é um rosto de perfil ou não. Caso ele ainda não esteja pronto para responder com precisão, podemos fornecer mais alguns exemplos para melhorar a sua capacidade de fazer generalizações.

### 3.1.3 Processamento de Linguagem Natural

Problemas que não têm a dependência de dados ambíguos são bastante bem resolvidos computacionalmente. Já aqueles que dependem de qualquer linguagem ambígua - como é o caso das línguas humanas - figuram como problemas mais complexos para as máquinas. Isso é o *Processamento de Linguagem Natural*: a habilidade computacional de processar linguagens ambíguas.

O *Processamento de Linguagem Natural* (doravante, utilizaremos a sigla em inglês: NLP) é uma subárea da IA que utiliza algoritmos de aprendizado de máquina para suas representações, também chamadas de modelos de linguagem. Para a criação de um modelo de linguagem, é possível utilizar dados de texto ou de fala; em nosso trabalho trabalhamos com dados escritos. É importante pontuar que deve-se levar sempre em consideração o tipo de dado que está sendo utilizado no estudo, para que as técnicas, os algoritmos e os parâmetros dos modelos sejam melhor construídos. As pesquisas em NLP incluem a produção de *Corpus*, *parser*, léxicos etc. E suas aplicações são também muito vastas, podemos citar a tradução automática, assistentes virtuais, corretores automáticos, detectores de *fake news* etc.

Um bom modelo de NLP deve ser capaz de fazer previsões e gerar interpretações probabilísticas a partir dos dados de entrada. Como na ilustração a seguir.



**Figura 3.5.** Ilustração simplificada de um processo de interpretação do português com um *output* que não foi explicitamente solicitado e que se adequa perfeitamente à fala

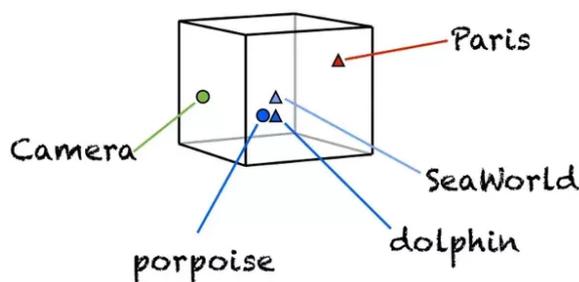
### 3.1.4 *Word Embeddings: You shall know a word by the company it keeps!* [Firth, 1957]

A abordagem de *Word Embeddings* parte da proposta teórica de *hipótese distributiva*, de Zellig Harris [Harris, 1954], que entende que há uma estrutura distribucional na língua e, assim, palavras que têm contextos similares terão significados similares. É uma abordagem que partilha, também, da concepção das teorias baseadas em uso, que preveem uma representação da linguagem a partir dos dados de uso.

Em geral, em modelos de NLP, as palavras são representadas pelo que chamamos de *Word Embeddings*, ou, simplesmente *embeddings*. Os *embeddings* são representações de palavras por vetores numéricos densos, que nos permitem mapear as palavras pelas características que carregam e pelas relações que estabelecem, a partir do *corpus* utilizado no treinamento. Uma boa representação vetorial das palavras de

uma língua deve ter uma boa generalização para, assim, funcionar como *input* para que outros modelos de NLP possam fazer boas previsões.

A figura abaixo ilustra um exemplo de um *espaço vetorial* ou *embedding space*, em que palavras com significados similares e com uma coocorrência alta se encontram mais próximas umas das outras e mais afastadas daquelas com que se relacionam com menor frequência.



**Figura 3.6.** Ilustração de um Espaço Vetorial hipotético, fonte:<https://www.kaggle.com/sbongo/do-pretrained-embeddings-give-you-the-extra-edge>

Em 2013, [Mikolov et al., 2013] desenvolveram um algoritmo de *word embeddings* chamado *Word2vec*, capaz de representar palavras como vetores a partir da previsão de contextos de ocorrência similares. Este é um algoritmo robusto, simples e eficaz, que serviu de ponto de partida para outros algoritmos ainda mais sofisticados e refinados que foram desenvolvidos em seguida, como o *BERT* [Devlin et al., 2019], ou o *GloVe* [Pennington et al., 2014], que será utilizado para a geração do nosso *embedding*.

## 3.2 Trabalhos Relacionados

Historicamente, para a ciência da computação, as competições de soluções e experimentos com jogos funcionam como testes importantes para avaliar a qualidade das decisões tomadas pelas máquinas e para promover inovações nos processos e técnicas. A performance em jogos simétricos e de soma zero, independentes de

processamento de linguagem natural e que, portanto, não lidam com ambiguidades, como Xadrez ou Go, é um problema resolvido para o aprendizado de máquina, em que os modelos superam enormemente o desempenho humano [Silver et al., 2016]. Para jogos que envolvem a cooperação entre os agentes, também foram desenvolvidos estudos e experimentos, embora com menor eficácia, como em [Bard et al., 2019], em que os agentes mostraram um bom desempenho quando treinados com a mesma IA, mas quando em confronto com agentes com estratégias diferentes, ou seja, "desconhecidos", têm um desempenho ruim, nos alertando para a natureza das relações de cooperação nos jogos [McCabe et al., 2001], que pressupõem, necessariamente, um grau de desconhecimento dos processos mentais entre os agentes e um esforço para maximizar o entendimento estratégico.

O coreano Lee Sedol, considerado o melhor jogador do milenar jogo de Go da atualidade, perdeu a série de partidas para o modelo computacional desenvolvido em 2015 pela *DeepMind* [Silver et al., 2016]. O jogador, em entrevista <sup>3</sup>, afirmou que, antes do jogo, pensou que seria fácil vencer, que no seu entendimento, o modelo seria baseado apenas em cálculos probabilísticos e estatísticos simples, mas que, quando se deparou com algumas jogadas, mudou o seu ponto de vista e afirmou perceber até mesmo criatividade em seu adversário. Embora possa ser precipitado afirmar algo sobre o potencial criativo de modelos de IA, entendemos que a maneira de jogar Go foi profundamente alterada depois do contato com as estratégias artificiais.

[Zunjani & Olteteanu, 2019] utilizaram *Codenames* em testes de criatividade e habilidade de associações. Nos inspiramos nesses exemplos para ousarmos nos propor a experimentar as influências que as soluções computacionais podem exercer sobre jogadores humanos, e de que modo as estratégias humanas podem se beneficiar do nosso modelo computacional. Além disso, investigamos os exemplos testemunhados em jogos entre humanos e como estes podem informar o desenvolvimento de um modelo computacional.

---

<sup>3</sup><https://www.alphagomovie.com/>

Consideraremos que, para jogos que envolvem algum componente linguístico, os modelos computacionais ainda têm um desempenho muito inferior ao dos jogadores humanos. Embora o desempenho seja inferior, os erros cometidos pelos humanos têm um caráter distinto dos erros cometidos pela máquina, estando ligados principalmente à menor capacidade de recuperação de itens ou associações pela memória. Em 2019, foi anunciada a primeira competição de *Codenames AI*<sup>4</sup>, para além do processamento da linguagem natural envolvido no jogo, a cooperação e o caráter epistêmico envolvidos desafiam as técnicas de IA. Muitas foram as soluções propostas para o problema durante a competição, como em [Kim et al., 2019], mas estas tendem a ser feitas somente a partir de *embeddings* de palavras e do cálculo das distâncias entre possíveis dicas e as palavras alvo. Uma arquitetura mais rebuscada, foi proposta por [Jaramillo et al., 2020], os autores utilizaram o algoritmo *Tf-idf*, que se baseia em uma medida estatística capaz de indicar a importância de um item lexical com relação a um determinado *corpus* linguístico. No artigo publicado, os pesquisadores relataram um melhor desempenho dos *bots* (agentes automáticos, diminutivo de *robot*) em comparação com os jogadores humanos para a tarefa do *Codemaster* de fornecer a dica. No entanto, o teste qualitativo apontou para a dependência humana de palavras relacionadas a conceitos abstratos e relações sutis, que os agentes não humanos têm mais dificuldade para identificar.

As autoras de [Koyyalagunta et al., 2021] apresentaram recentemente uma abordagem inovadora e eficiente para o problema. Elas propuseram o aprimoramento de dicas com o uso de sua função (*DETECT*) responsável por rebalancear os valores das palavras candidatas a dicas, de acordo com sua frequência de uso. Além disso, utilizam (em combinação com outras técnicas) um *embedding* elaborado a partir das definições dos dicionários de *Cambridge*, *Oxford Collins* e do dicionário online *dictionary.com*, segundo as autoras, a abordagem baseada em dicionários tem por intuito capturar melhor as ocorrências de meronímia, sinonímia e relações semânticas

---

<sup>4</sup><https://sites.google.com/view/the-codenames-ai-competition/home>

fundamentais. A técnica utilizada é nomeada *Dict2vec* [Tissier et al., 2017], publicada em 2017. Elencamos este modelo, que tem o melhor desempenho relatado até agora, como *baseline* para o nosso estudo. Ressaltamos que o nosso *baseline* foi completamente desenvolvido na língua inglesa, que dispõe de uma quantidade de recursos e adaptações técnicas imensamente maior que o pt-BR. Assim, um resultado similar para o nosso modelo em pt-BR já estabeleceria um avanço consideravelmente satisfatório.

Em nosso *baseline*, as autoras se atentam para o fato de que, diferentemente de outros jogos, competitivos e com dados mais formais que as línguas humanas, em *Codenames*, somente o *score* de jogos ganhos entre dois agentes artificiais não é uma boa métrica para a avaliação do bom desempenho nas tarefas do jogo. O *score* nos aponta apenas o quão bem as representações de palavras nos modelos de linguagem disponíveis para o *Codemaster* e *Guesser* estão alinhadas, apenas uma avaliação minuciosa das dicas fornecidas permite observar se algum tipo de inteligência está sendo gerada. É necessário uma análise qualitativa, humana e manual de exemplos gerados pelos modelos computacionais para validar as pistas fornecidas e os processos desenvolvidos, já que dicas fornecidas por modelos artificiais podem gerar respostas corretas para o seu par que adivinha, também artificial, mas serem péssimas dicas para jogadores humanos. Ou seja, um alto desempenho de vitórias, não significa que o modelo esteja gerando pistas significativas dentro das representações linguísticas humanas.

Assim, dentre os problemas encontrados em alguns exemplos de erro, o maior desafio para os modelos computacionais, ou *bots*, anteriormente desenvolvidos, parece ser a predição do entendimento, ou das associações possíveis que poderão fazer os outros jogadores, uma tarefa que entre humanos é, muitas vezes, trivial. Como trabalham por um objetivo comum, além de entender as palavras, quem dá a dica e quem a recebe, precisa ter uma compreensão, ainda que vaga, sobre o entendimento e associações possíveis que o parceiro de jogo possa dar às palavras. As experiências

compartilhadas, o *background*, o *input* fonológico, o *input* das outras palavras do tabuleiro etc podem contribuir com pistas sobre os caminhos do entendimento dos companheiros de jogo. Para tanto, o que postulam os autores em [Kounios & Beeman, 2014] sobre os *insights*, nos permite analisar as soluções encontradas pelos jogadores em algumas situações do jogo. Ademais, além dos fenômenos de meronímia, sinonímia, hiper e hiponímia, os conceitos de protótipo lexical e representação mental, presentes em [Cruse, 2014] e [Scott-Phillips, 2015] serão cruciais para elaborarmos um entendimento sobre a dinâmica entre os jogadores de *Codenames*, exploraremos um pouco de tais pressupostos teóricos no capítulo seguinte.

# Capítulo 4

## Pressupostos Teóricos

Para termos consistência nas decisões tomadas ao desenvolvermos o nosso modelo computacional para o jogo *Codenames*, é necessário que entendamos alguns pressupostos da linguística cognitiva, da análise lexical e que mergulhemos brevemente em algumas teorias cognitivas que podem nos trazer intuições sobre os processos envolvidos no jogo. Trazemos aqui uma conceituação e alguns apontamentos sobre o referencial teórico que nos guiou ao analisarmos alguns jogos entre dois jogadores humanos, e que foi crucial para a lapidação de nosso modelo computacional.

### 4.1 Palavras x Léxico

Em *Terminology: theory, methods and applications*, [Cabré, 1999] faz uma distinção entre: “palavras” e “léxico”, a autora propõe uma definição específica para cada um dos termos, que não devem se confundir. Segundo ela, o léxico consiste das palavras de uma língua e das regras que regem a criatividade associativa do falante. Já as palavras, seriam as unidades de referência da realidade empregadas pelos falantes. Assim, compõem a dimensão linguística mais estreitamente ligada ao mundo real. Com a tarefa proposta por *Codenames* os jogadores recorrem constantemente aos

seus conhecimentos linguísticos, às referências presentes na memória de suas experiências e ao *background* social e cultural.

Além disso, em *Codenames* as tarefas devem ser cumpridas utilizando apenas palavras únicas, isoladas de qualquer contexto. No entanto, muitas vezes, para conseguir associar as três palavras-alvo, a partir da dica dada, é preciso que o jogador acesse, além de outras palavras que poderiam se relacionar diretamente com a dica dada em determinados contextos, o léxico da língua de forma mais ampla, de modo a inferir palavras que não tenham uma coocorrência provável com a dica dada, mas que se relacionam a esta semântica ou pragmaticamente. Podemos perceber isto no exemplo a seguir: dada a dica *pote* o jogador adivinhou as palavras-alvo *jarro*, *amendoim* e *múmia* sem grandes dificuldades.

## 4.2 Associações Lexicais

Os usos e as combinações entre as palavras são mais do que apenas um conjunto de itens do acervo lexical de uma língua, cada palavra é um repositório de recursos complexos que têm grande relevância na organização mental do falante. De acordo com as pesquisadoras [Aitchison & Lewis, 1995], a aquisição lexical envolve, necessariamente, um entrelaçamento entre a estrutura fonológica, as classes de palavras, as formas flexionais, as derivadas e suas regras e os significados. Além disso, podemos incluir as relações que mantêm com outras palavras, assim, na aquisição de um novo item lexical, o seu significado pode ser mais bem compreendido a partir da explicação de tal item em termos da rede que existe entre os seus diferentes sentidos e sub-sentidos. O repositório de palavras testemunhadas ganha solidez em um léxico mental ao adquirir significado gradativamente, a partir da exposição contínua aos diferentes contextos de uso, como propôs o linguista J. R. Firth já em 1968 [Firth, 1968], quando postula o significados das palavras como equivalentes ao seu uso na língua.

Uma das marcas mais importantes da organização lexical mental está na frequência de uso de um item. Durante a produção, itens com alta ocorrência são recuperados mais rapidamente: é mais fácil e mais rápido para que o falante acesse a palavra *faca* do que *punhal*. Estudos de pausas e hesitações na fala demonstram que comumente ocorrem hesitações antes de palavras de baixa frequência [Levelt, 1983]. Há também, como já mencionado, uma organização por coocorrência, relações semânticas, fonológicas, pragmáticas ou até ortográficas, de modo que itens que se relacionam em algum desses aspectos são armazenados próximos uns dos outros. É importante notar que palavras opostas, antônimos ou complementares, como esquecer/lembrar, frio/quente, amor/ódio, compartilham muitos traços semânticos e têm referências similares, assim, apesar da oposição, são itens que ficam bem próximos no léxico mental e, em erros de fala, são frequentemente produzidos em lugar uns dos outros. Tarefas associativas fornecem importantes dados sobre a memória e a cognição, a partir do reconhecimento e das representações semânticas no processamento lexical. Para explorar tal conhecimento, dispomos de diferentes experimentos, no entanto, as tarefas de associações de palavras são consideradas o caminho mais direto para obter *insights* sobre o conhecimento semântico, já que os dados extraídos das associações de palavras são capazes de sinalizar diferentes tipos de relações entre os itens. No estudo desenvolvido por [Ghosh et al., 2010], são investigadas as diferentes áreas de ativação cerebral durante uma tarefa de associação lexical. A partir dos resultados encontrados, os autores afirmam haver um vínculo inequívoco entre o conteúdo semântico dos conceitos dos itens e a codificação gramatical, o que exige recursos substanciais de memória. Ainda segundo os autores, interpretações complexas envolvem o uso de informações pelas redes corticais, servindo não apenas aos mecanismos sensoriais, mas também às áreas de associação, essa capacidade não apenas nos ajuda a categorizar, organizar e integrar informações linguísticas, mas também a derivar novos conceitos a partir dos já existentes. Assim, segundo o estudo, a associação semântica entre palavras requer necessariamente que as informações

sejam processadas a partir da memória, mais especificamente, a partir de *traços* de memória dos itens lexicais testemunhados, que são manipulados usando redes hierárquicas de estruturas sintáticas e conceitos semânticos com base na associação de palavras, mesmo em uma tarefa relativamente simples, como a recuperação de palavras isoladas, o que nos indica que o processamento de conceitos de palavras no cérebro é essencialmente multimodal e provavelmente está distribuído em diferentes redes corticais.

Como apontam [Deyne et al., 2013], as redes de representação lexical (ou mesmo um *embedding* - como o que utilizamos neste trabalho) correspondem a uma representação idealizada de nossa rede lexical mental e podem apontar para relações entre os itens lexicais e as medidas de centralidade de determinados conceitos. Ainda de acordo com [Deyne et al., 2013], as propriedades derivadas desse tipo de rede foram instrumentais em três diferentes tradições de pesquisa, que têm se concentrado em três elementos, quais sejam: a força da associação direta, força de segunda ordem e similaridade distributiva e topologia de rede e medidas de centralidade. Os autores reafirmam o fato de que as três linhas de pesquisa correspondem a uma mudança de interações locais medidas como força direta, para interações dentro de uma sub-rede (medidas mediadas e distribucionais) para características globais da rede (topologia e centralidade da rede) e coincidem com os desenvolvimentos teóricos testemunhados à época da publicação - como o interesse renovado na teoria das redes e os avanços computacionais - mesmo ano em que tivemos a publicação de [Mikolov et al., 2013], que difundiu e consolidou o uso dos *embeddings*.

Já em [De Deyne et al., 2016], as redes de representações semânticas construídas a partir de *corpora* são investigadas e comparadas com uma forma mais direta de avaliar o léxico mental por meio de tarefas de associações de palavras. As comparações apresentadas indicam propriedades específicas destacadas por cada uma das abordagens e agrupamentos distintos, mas, principalmente, a pesquisa explícita uma considerável divergência no julgamento de relação (ou agrupamentos) entre

palavras, as redes de linguagem proposta pelo estudo têm certa analogia com os *embeddings* que aplicamos, e não são capazes de reproduzir com precisão todos os julgamentos humanos envolvidos no processo de relacionar os itens lexicais em uma rede semântica mental, ainda que sejam a melhor aproximação que tenhamos do modelo de representação lexical mental até hoje.

### 4.3 Insights

Algumas das soluções criadas pelos jogadores de *Codenames* se diferenciam daquelas produzidas de forma estritamente analítica. Em exemplos como: dada a dica *Pomarola* com o intuito de indicar as palavras-alvo *elefante*, *gôndola*, e *Itália*, os dois últimos itens lexicais estão relacionados no *common ground* dos falantes, com o modelo cultural de Itália, sua culinária, sua fama turística etc. Já para o item *elefante*, não podemos traçar qualquer relação linguística, não há uma lógica aparente e precisa, não identificamos grande coocorrência, similaridade semântica ou fonológica, ou qualquer outro traço que leve o jogador à resposta correta. E, no entanto, este exemplo faz parte dos dados que coletamos, e obteve uma taxa total de acerto. Por outro lado, o que os pesquisadores de psicologia cognitiva John Kounios e Mark Beeman, chamaram de "lógica difusa", presente nos *insights*, pode nos trazer algumas intuições sobre alguns dos processos presentes em *Codenames*. No exemplo dado, o *background* cultural dos dois jogadores, que conhecem a publicidade da marca Pomarola que utiliza um elefante, colaborou com a associação dos três itens de maneira não linear e possibilitou o *insight*.

Ter um *insight* é o equivalente a ter uma compreensão repentina de algo, resultando em uma nova interpretação de uma situação ou apontando a solução de um determinado problema [Sternberg & Davidson, 1995]. Geralmente, os *insights* resultam de uma reorganização ou reestruturação de elementos de um determinado problema ou situação, de modo que podem surgir sem que haja uma outra

compreensão anterior, que anteceda o momento *Aha!* dos *insights*, conforme [Kounios & Beeman, 2014].

[Kounios & Beeman, 2014] afirmam que este é um fenômeno de múltiplos domínios e que, além de prover soluções para alguns problemas, os *insights* são responsáveis pelo entendimento de piadas e metáforas, contrastando com estratégias analíticas para a solução de problemas e interpretações. Ao contrário das soluções analíticas, eles ocorrem inconscientemente, e emergem subitamente [Bowden, 2003]. Assim, os processos que levam às soluções por *insights* só podem ser reconstruídos após a sua emergência.

Pelas características não óbvias e de reorganização conceitual, alguns autores identificam os *insights* como uma manifestação da criatividade [Friedman & Förster, 2005]. [Subramaniam et al., 2009], apontam para a influência que o bom humor exerce sobre a capacidade cognitiva de solução de problemas, especialmente aqueles solucionados a partir de *insights*. Não podemos ignorar esse aspecto, já que o nosso estudo está focado em um ambiente lúdico. [Topolinski & Reber, 2010] salientam que quando a solução de um problema complicado surge na mente, o processamento de uma informação que era antes difícil e custosa, se torna fluente. Como consequência do impulsionamento de uma ideia inconsciente para a consciência, as soluções por *insights* parecem óbvias e certas ao sujeito, que a partir de um processamento mental altamente fluente, experiencia uma sensação de prazer e confiança na precisão da solução a que chegou. Além disso, [Kounios & Beeman, 2014] afirmam que a solução de problemas com *insights* é acompanhada de um intenso processo de recompensa neural, o que também nos ajuda a entender muito sobre a diversão, e principalmente a intensidade da satisfação, presente em alguns exemplos mais inusitados em nossos jogos entre humanos.

Mudanças nas representações mentais são cruciais para a resolução de problemas por *insights* [Davidson, 2003]. Diante de um desafio novo, para o qual não temos um ritual de procedimentos pré-definidos, buscamos em um espaço por caminhos

alternativos que levem à solução, diferentes processos mentais, nos ajudam a guiar essa busca e a alcançar uma mudança na representação interna dos dados e na relação entre os dados. Ressaltamos a consideração dos pesquisadores citados aqui, sobre a natureza não incremental, mas difusa, desses processos, quando resultam em uma solução por *insights*. Assim, tocamos em nosso próximo tópico de relevância teórica.

#### 4.4 Representações Mentais e *Mindreading*

Thom Scott-Phillips em *Speaking Our Minds* [Scott-Phillips, 2015], discorre sobre o fenômeno da representação mental, que incluem desde os conhecimentos e pressuposições mais básicas, a conceitos e crenças mais complexas. Segundo o autor, essas representações estão constantemente expostas a mudanças, especialmente quando influenciadas por *inputs* que sugerem a alteração de representações já existentes, ou a criação de novas representações. Como postulado pelo mesmo autor em [Scott-Phillips, 2010], a transferência de informação não é uma definição, mas uma consequência da comunicação, assim, em nosso estudo, nos beneficiamos do entendimento, de que a comunicação ocorre quando uma ação (sinal) produzido provoca uma mudança (reação), onde o sinal e a reação foram projetados para o mesmo propósito.

Em uma situação de interação construída e estimulada por um jogo, há um grau elevado de previsão e de expectativa para a comunicação, excluindo os ruídos habituais que interferem no surgimento e nas funções pretendidas pelos indivíduos, amplificando, assim, as características que nos interessam para a análise. Em *Codenames* o sucesso dos jogadores depende da tentativa de potencializar a precisão das declarações e o entendimento, em suma: maximizar o sucesso da comunicação e diminuir as lacunas do entendimento, é o que está em jogo em *Codenames*.

Desse modo, a observação da interação entre dois jogadores humanos nos leva à intuição de que há camadas de processamento da informação operando, de modo que

não é preciso produzir uma evidência explícita do que se quer comunicar, mas a evidência da intenção da expansão de significados por caminhos diversos é suficiente para que o outro jogador seja capaz de se esforçar e encontrar as conexões que convergem entre os seus modelos mentais. O êxito ao expressar e reconhecer intenções é gratificante, expandir as associações possíveis entre itens lexicais e compreender as representações mentais do outro pode ser ainda mais compensatório, temos aí uma outra base para a compreensão da diversão no jogo: ao explorar a capacidade humana de comunicação e seus desdobramentos.

Sob um ponto de vista representacional, os pressupostos da Teoria da Mente [Apperly, 2011] e da *Simulation Theory* de Alvin Goldman [Goldman et al., 2006] nos ajudam a entender alguns dos mecanismos empregados na leitura mental (*mindreading*) entre os jogadores de *Codenames*. A habilidade de *mindreading* é exclusivamente humana segundo [Penn & Povinelli, 2007], e isto caracteriza o cerne do processo cognitivo envolvido nas interações sociais e na própria comunicação, conforme aponta [Grice, 1989]. Além disso, o *mindreading* é crucial para gerar, manter e atualizar as representações do que é mutuamente sabido, e para entender o que é relevante no contexto em que se está inserido [Apperly, 2011]. As pessoas frequentemente relatam o que pensam e o que sabem sobre o pensamento dos outros, nas interações sociais, é importante reter essas informações e não confundi-las com os próprios conhecimentos e opiniões. Assim, mesmo que os jogadores se conheçam pouco ou nada, informações que estão disponíveis: como a faixa etária, o gênero, o sotaque (que indica o lugar de proveniência) etc, oferecem pistas relevantes para supor o entendimento de dicas que podem ser oferecidas, e inferências que podem ser feitas a partir das dicas recebidas. Se entendemos que a tarefa envolvida em nosso jogo depende de fenômenos como os que acabamos de descrever, é razoável postularmos, também, que um modelo computacional não poderá desempenhar satisfatoriamente a tarefa se a sua arquitetura não considerar estes aspectos e tentar fazer uma aproximação para a simulação de tais processos cognitivos.

## 4.5 Protótipos Lexicais

A Teoria dos Protótipos é explorada por diversas ciências cognitivas, e é especialmente útil para estudarmos o conhecimento lexical. A partir dos exemplos e estratégias testemunhadas entre os nossos jogadores, é razoável que façamos uma reflexão acerca do papel desempenhado pelo significado prototípico dos itens lexicais, e do quanto eles são considerados em um jogo como *Codenames*.

As teorias da linguística cognitiva pressupõe que a linguagem é um conhecimento em conexão com a totalidade da experiência humana. Assim, as unidades linguísticas devem ser analisadas não isoladamente, mas entrelaçadas em uma teia que abranja as capacidades cognitivas gerais, a organização conceptual, a categorização, as experiências sócio-culturais e individuais. Segundo D. A. Cruse [Cruse, 1986], podemos definir o significado de uma palavra como um padrão de afinidades e desafinidades que esta traça com todas as outras palavras da língua, com as quais pode contrastar relações semânticas. Assim, a Teoria dos Protótipos não atribui estabilidade e definições precisas às categorias, mas as define com traços gradientes e de limites difusos, como indica [Zadeh, 1965]. Para cada categoria, estes traços terão relevância conforme os itens lexicais gravitarem mais próximos ao centro cognitivo exemplar. Um bom exemplo para o que dissemos, seria *maçã* como um exemplar central da categoria *fruta*, e então, poderíamos citar *pêra* como um item mais próximo ao centro exemplar *maçã* do que uma outra fruta, como *tomate*.

Para a maioria das categorias, temos uma noção, mais ou menos bem definida de acordo com uma instância particular, de uma forma canônica apropriada. O autor sugere que possa haver mais de uma dimensão para essa centralidade do exemplar prototípico, com relações mais complexas, de modo que diferentes categoria e diferentes contextos podem ativar diferentes dimensões ou combinações de dimensões. De acordo com [Cruse, 2014], os exemplos prototípicos de uma categoria parecem ser acessados e processados mais rapidamente do que os exemplares mais periféricos. No

entanto, em nosso jogo, os exemplares prototípicos sozinhos não são suficientes para traçar as associações necessárias, é preciso traçar um acesso constante aos exemplos mais periféricos, ainda que o caminho para o acesso se abra pelo exemplar mais prototípico, já que o protótipo de uma categoria está fortemente implicado na representação mental desta categoria.

Conforme [Smith & Medin, 1981], a versão probabilística do modelo de protótipos propõe uma variância para as instâncias de um mesmo conceito, na medida em que compartilham certas propriedades e variam o grau de pertencimento à categoria.

Alguns modelos de protótipos, consideram também uma ordenação hierárquica das propriedades compartilhadas pelos itens de uma mesma categoria, assim, cada traço da definição de um item lexical é tido como um parâmetro.

Em nossas análises dos exemplos coletados, nos alinhamos ao pensamento de [Cruse, 2014], principalmente no que tange o princípio para a análise semântica, em que os conceitos e, portanto, os significados das palavras não têm limites nítidos: o status dos recursos para o pertencimento a determinada categoria lexical, pode mudar de necessário para provável ou possível.

## Capítulo 5

# Jogos entre Humanos

As palavras têm a capacidade de evocar *frames* na mente dos falantes. Os *frames* funcionam como uma compilação estruturada de conhecimento de mundo ou expectativas [Filmore, 2006]. Por exemplo, o *frame* de uma sala de aula produz uma representação mental para o falante que envolve objetos (caderno, lápis, borracha etc), pessoas (alunos e professores), situações comuns, memórias etc. Escritores entendem bem esse mecanismo cognitivo humano e, não raro, utilizam uma palavra, descrevem a presença de um objeto que é capaz de evocar uma cena completa - gerando suposições sobre a personalidade de uma personagem ou acontecimento anteriores à cena - ou evocam uma situação arquetípica da experiência humana a partir de descrições pontuais, como no trecho a seguir, de *Os Anos*, da escritora Annie Ernaux:

*Em meio à proximidade dos corpos, a mastigação e as brincadeiras, o pão, o foie gras e a recusa dos assuntos graves, outra vez, ia sendo construída a realidade imaterial das refeições em família. Dava para sentir a força e a espessura desta realidade quando saíamos alguns minutos para fumar um cigarro ou ir à cozinha conferir se o peru estava assado. Ao voltar, o burburinho da mesa já era outro, com uma nova conversa. Alguma coisa da nossa infância vinha até nós. Uma cena antiga e dourada, com pessoas*

*sentadas, os rostos misturados, em um rumor indistinto de vozes.* [Ernaux, 2021]

No trecho, a escritora consegue evocar no imaginário de cada leitor um *frame* íntimo, cada um de nós terá sua representação mental específica para semelhante experiência. No entanto, elementos chave do *frame* são compartilhados em nossas vivências sociais, e bastam algumas palavras da autora que se relacionam à essa experiência sensorial, alguns detalhes, para nos evocar a memória de uma cena completa. Acreditamos que os *frames* são fundamentais em uma tarefa associativa como a nossa. Dentro do *frame restaurante*, por exemplo, além de colher, garfo, faca, comida sobremesa e outros itens lexicais que poderiam ser ativados, podemos contar com ativações mais difusas: como sabores, aromas, temperaturas, tipos de pessoas que frequentam aquele lugar etc. Assim, as associações lexicais se amplificam, os significados das palavras são uma extensão de todas as nossas experiências culturais, capazes de gerar implicações e inferências, a partir de sua riqueza e sutilezas. Com o intuito de entender um pouco os processos de decisão dos jogadores humanos envolvidos em *Codenames* e, assim, beneficiar o desenvolvimento da arquitetura de nosso modelo computacional, produzimos um conjunto de exemplos de dicas e palavras-alvo testemunhadas em uma série de jogos. Além disso, esses dados nos possibilitarão fazer uma breve análise e servirão de base para a comparação com o nosso modelo.

## 5.1 Dados de jogos entre humanos

Devido à pandemia do novo Coronavírus, que se estende no Brasil desde Março de 2020, não tivemos a oportunidade de diversificar os informantes de nossos dados. Idealmente, deveríamos contar com uma base de exemplos de jogadores com *backgrounds* distintos, no entanto, acreditamos que uma boa quantidade de exemplos

de combinações entre palavras, atenderá aos propósitos deste trabalho. Ademais, a simplicidade das regras de *Codenames* faz com que a curva de aprendizado do jogo e de sua dinâmica, não seja tão significativa, tornando quase indiferente a experiência anterior dos jogadores com jogos de tabuleiro em geral, e isto não nos impõe, também, a necessidade de variar os informantes que poderiam dominar o jogo e encontrar caminhos mais curtos para a vitória com o passar das rodadas, como acontece em outros experimentos com jogos, não estamos expostos a este problema em *Codenames*. Seleccionamos dois informantes, que nos ofereceram 100 exemplos de conjuntos de combinações de palavras.

São nossos informantes: Maisa (doravante Jogadora 1), 28 anos, programadora, tem experiência com jogos de tabuleiro. Marcelo (doravante Jogador 2), 35 anos, jornalista, tem experiência com jogos de tabuleiro.

Inspirados pelo Método Protocolo Verbal, coletamos, também, alguns dados introspectivos qualitativos para os exemplos que nos pareceram mais proveitosos à análise, ou que tiveram uma solução menos óbvia. As respostas para essas perguntas foram coletadas de forma não estruturada, e têm o intuito de guiar as nossas análises e *insights* sobre os processos linguísticos envolvidos em cada exemplo. Pedimos aos nossos informantes que respondessem oralmente às seguintes perguntas:

Para o *Codemaster*:

- Você poderia descrever o caminho para chegar à dica proposta?
- Você pensou em outras opções de dicas?

Para o *Guesser*:

- Como você chegou às palavras-alvo?
- Você ficou com dúvidas entre outras palavras do tabuleiro?

Solicitamos aos nossos informantes que fornecessem sempre 1 dica para 2 ou 3 palavras-alvos, com o intuito de padronizar os dados. Apresentamos aqui uma lista

com 30 exemplos de situações de jogos que nos serão mais úteis para a análise posterior. Os exemplos relatados na lista seguinte contam com uma taxa de acerto de pelo menos 2 palavras-alvo para cada dica. A lista com todos os exemplos coletados pode ser encontrada no apêndice. Os exemplos apresentados servirão também para que o leitor possa compreender melhor a dinâmica do jogo.

**Tabela 5.1.** Jogos entre dois jogadores humanos

Palavras alvo	Dica	Acertos
abelha, porta, madeira	Jataí	2
corvo, destino, espada	morte	3
santa, julgamento, etiqueta	comportamento	2
elefante, gôndola, Itália	Pomarola	3
oásis, labirinto, sono	confuso	3
sono, marionete, caveira	pesadelo	3
vírus, paciente, pulso	corona	3
sapo, irmão, chapéu	príncipe	3
regra, carreira, artigo	ABNT	2
oásis, névoa, espírito	etéreo	2
testemunha, tinta, branco	parede	2
testemunha, gangster, espírito	assassinato	3
carruagem, desenho, sinal	Cinderela	3
caranguejo, confusão, torta	aperto	2
amendoim, televisão, sonho	sextou	2
amendoim, jarro, ordem	guardado	2
afoito, pimenta, tornado	agitação	2
cone, cruzado, labirinto	trânsito	3
raspar, guitarra, pimenta	arranha	3
sonho, pentágono, grade	apanhador	3
grade, pentágono, ferreiro	cerca	3
ferreiro, janela, esfinge	construção	2
esfinge, pentágono, lanterna	mistério	3
gota, páscoa, feriado	chocolate	3
ervilha, vapor, gota	cozido	2
tenda, pesca, cobra	acampamento	2
lágrima, chá, peixe	água	2
estábulo, pelotão, prensa	polícia	2
pluma, toque, pêssego	suave	3
rainha, bicicleta, pai	coroa	3
abelha, morte, Antártida	climática	3

## 5.2 Análises das Escolhas dos Jogadores

Está claro que em *Codenames* estamos diante de um problema que talvez não seja puramente linguístico (se considerarmos a linguística como uma disciplina isolada, que trata de processamentos específicos), em nosso estudo, esbarramos com estudos de pelo menos outras duas disciplinas: a psicologia e a neurociência. No entanto, é possível chamar o nosso problema de *problema de língua*, tendo como definição o que elabora Scott-Phillips em [Scott-Phillips, 2015]: a *língua* é o conjunto de características cognitivas que nos permite adquirir e usar as diversas línguas humanas. Analisamos aqui alguns exemplos que coletamos em nossos dados de jogadores humanos, lembramos que além das 3 palavras-alvo presentes nos exemplos, o jogador tem diante de si um tabuleiro com outras 22 palavras, que atuam como distratores e que, frequentemente, confundem o alvo de determinada dica. Observamos que a taxa de erro no jogo é muito alta, seja por algum equívoco no processo de *mindreading*, ou seja, os jogadores se enganam ao inferir as conexões que o outro quis estabelecer, ou os sentidos que são atribuídos a determinadas palavras são ignorados ou não estão ativos na mente de ambos os jogadores. Além disso, é comum que as palavras do tabuleiro que não são alvo, acabem por confundir os jogadores. Para exemplificar: a Jogadora 1 deu a dica *construção* para as palavras *ferreiro*, *janela* e *esfinge*, a dica nos pareceu muito boa e esperávamos uma taxa total de acerto, no entanto, havia a palavra *poeira* na mesa, que foi apontada pelo jogador 2 em lugar de *esfinge*. Ele parece ter acessado um sentido para o item lexical *construção* como o processo de atividade de construir, para o qual os itens que escolheu se conectam muito bem. Já a jogadora pensava em um outro sentido de construção: como a matéria final e estabelecida e não se atentou para a possibilidade dos caminhos tomados pelo Jogador 2. Já o exemplo de *sapo*, *irmão* e *chapéu* com a dica *príncipe* teve um acerto total. *irmão* é uma palavra que, a princípio não se relacionaria com a dica, no entanto, por ser a única palavra do gênero masculino no tabuleiro, pôde ser associada

a *príncipe*. Já no exemplo seguinte: *regra, carreira, artigo* que teve como dica *ABNT* o jogador não conseguiu adivinhar a palavra *carreira*, em seu lugar ele tentou a palavra *tabela*, que também estava na mesa e acabou por gerar um conflito associativo. Com *oásis, névoa, espírito* os jogadores tiveram um problema similar, a dica *etéreo* levou à associação com a palavra *sono*, em lugar de *oásis*. Já o grupo de palavras *elefante, gondola, Itália*, teve êxito com a dica *pomarola* porque ambos os jogadores tinham aproximadamente 30 anos e compartilhavam a memória dos comerciais da marca de molhos que utilizava um elefante verde em sua publicidade, ou seja, o *background* foi crucial para o sucesso. Veremos aqui alguns outros exemplos, com mais detalhes e respostas dos sujeitos às nossas perguntas.

### 5.2.1 Dica: Jataí

O Jogador 2 ofereceu a dica *Jataí* para as palavras *abelha, porta e madeira* de maneira muito rápida. A Jogadora 1 também foi impressionantemente rápida para desvelar as palavras-alvo. As respostas que forneceram às nossas perguntas deixam explícito que o êxito foi devido ao *background*, a partir de uma experiência compartilhada, com elementos bastante ativos: os jogadores haviam construído juntos uma casa de abelha, de madeira, com uma porta especial na semana anterior.

### 5.2.2 Dica: Cinderela

A Jogadora 1 foi perguntada sobre os seus processos para chegar à dica *Cinderela* para as palavras *carruagem, desenho, sinal*. Nos chamou a atenção que a dica tenha sido eficiente principalmente para a palavra *sinal*, já que este exemplo teve uma taxa total de acerto.

- Você poderia descrever o caminho para chegar à dica proposta?

“*Sinal* era uma das minhas palavras-alvo mais difíceis. Eu precisava conectá-la a algumas das outras para vencer o jogo, e nenhuma parecia possível de formar um bom grupo com ela. Me lembrei das badaladas que a personagem Cinderela escuta nas histórias, e achei que meu parceiro de jogo pudesse inferir o mesmo, considerando as palavras que estavam expostas no tabuleiro.”

- Você pensou em outras opções de dicas?

“Não.”

Perguntado sobre os seus processos para chegar às palavras-alvo corretas, o Jogador 2 nos relatou o seguinte.

- Como você chegou às palavras-alvo?

“As palavras *carruagem* e *desenho* foram bem fáceis e óbvias de acertar. A terceira palavra me deixou muito em dúvida, mas depois pensei: o sapatinho de cristal é um “sinal” de que ela é ela de fato, é muito simbólico na história.”

- Você ficou com dúvidas entre outras palavras do tabuleiro?

“Sim, em alguns momentos pensei que a terceira palavra-alvo poderia ser *extraterrestre*.”

Como podemos perceber a partir dos relatos dos nossos jogadores, a polissemia foi crucial para a resolução deste exemplo. Além disso, podemos pensar no papel que as associações fracas e a lógica difusa, descritas por [Kounios & Beeman, 2014], desempenham nos *insights* dos jogadores. As associações que os dois fizeram foram distintas, mas o processo de *insight* foi similar para ambos, e os levou à mesma resposta: *sinal*, que realmente não era uma palavra fácil de ser acertada, mas que pode figurar na categoria proposta a partir de parâmetros distintos.

### 5.2.3 Dica: mistério

A Jogadora 1 ofereceu a dica *mistério* para as palavras *esfinge*, *lanterna* e *pentágono*, para as duas primeiras palavras nos parece mais fácil estabelecer a conexão: a esfinge como um símbolo forte de enigma em nossa cultura, e a lanterna como uma ferramenta utilizada por detetives e investigadores também é um elemento marcante no imaginário ocidental. Questionamos a jogadora sobre a escolha da dica para a palavra *pentágono*.

- Você poderia descrever o caminho para chegar à dica proposta?

“Para mim é um mistério o que é o Pentágono (EUA), e o que acontece ali.”

- Você pensou em outras opções de dicas?

“Não.”

Está claro que a polissemia de *pentágono* como forma geométrica e como um órgão governamental norte americano foi crucial para o sucesso. Além disso, o entendimento do papel dos Estados Unidos no mundo, sua história pregressa com guerras e questões de segurança, espionagem etc, fazem também parte da conexão estabelecida entre a palavra e a dica. O curioso é que o Jogador 2 acertou todas as palavras-alvo, portanto não só teve o mesmo *insight* que a jogadora, como compartilha das suas percepções sobre política internacional e de seu *background* cultural.

### 5.2.4 Dica: climática

Para as palavras alvo *abelha*, *morte* e *Antártida* testemunhamos uma hierarquia de conceitos que foram se combinando e encaminharam a Jogadora 1 para as palavras alvo. Vejamos: a regra do jogo especifica que a dica deve ser composta de uma palavra única, assim certo de que *climática* ativaria *mudança* no léxico mental da Jogadora 1, o Jogador 2 deu a dica para palavras que se relacionam com as

consequências da mudança climática. Certamente é necessário que a Jogadora 1 tenha conhecimentos teóricos e conceituais sobre o fenômeno, como a ameaça à extinção das abelhas, a morte e extinção de espécies de plantas e animais, o derretimento de geleiras na Antártida etc. Observamos um caso emblemático de sucesso pelo compartilhamento de um determinado *background* cultural, conceitual e até mesmo ideológico.

### 5.2.5 Dica: arranha

Neste exemplo os jogadores tiveram a mesma percepção sinestésica para as palavras-alvo *raspar*, *guitarra* e *pimenta*. Perguntamos ao Jogador 2 sobre o seu processo para chegar à resposta correta.

- Como você chegou às palavras-alvo?

“*Raspar* é quase um sinônimo de arranhar pra mim, pra *guitarra* pensei na sensação de “arranhado” que o som dos instrumento por vezes nos causa, e sobre a *pimenta*, sabe aquele arranhado na gargante que ela às vezes dá?”

- Você ficou com dúvidas entre outras palavras do tabuleiro?

“Não, não tinha nenhuma outra palavra que pudesse ter qualquer ligação com a dica para mim.”

### 5.2.6 Dicas: água, polícia e acampamento

É notável que uma estratégia básica deve considerar minimizar a distância das associações entre a dica e as palavras-alvo e maximizar essa mesma distância em relação às outras palavras dispostas no tabuleiro, que atuam como distratores. No entanto, é comum, dada a complexidade envolvida em traçar essas associações e buscá-las na memória, que o jogador se concentre de tal modo nas palavras-alvo, que esqueça ou ignore o passo de maximizar a distância das outras palavras.

Para a dica *água* as palavras-alvo eram muito claras: *lágrima*, *chá* e *peixe*, mas a palavra *gota* também figurava no tabuleiro, de modo que o *Guesser* a escolheu de forma bem direta, sem também se preocupar em relativizar as distâncias associativas, o jogador 2, que era o *Guesser* neste momento, nos relatou que olhou para a palavra *gota* e esta lhe pareceu muito óbvia, dispensando qualquer consideração.

No exemplo da dica *polícia*, que nos pareceu uma boa dica, capaz de mapear uma associação criativa entre *pelotão*, *estábulo* e *prensa*. A associação mais fraca, com *estábulo* foi adivinhada, mas a palavra *revista* estava no tabuleiro e fez com que *prensa* fosse preterida.

Já para *acampamento* a palavra *feriado* gerou uma confusão com as palavras-alvo e provocou o erro. Evidenciando a importância de considerar as palavras que atuam como distratores e de maximizar as distâncias que as dicas podem estabelecer com elas. Isto pode parecer trivial a princípio, mas a quantidade de palavras dispostas (25), o desafio associativo entre as palavras-alvo, combinado ao tempo reduzido (não há uma regra que estabeleça o tempo para cada jogador, mas é razoável não se demorar em demasia), faz com que o cálculo de todas as distâncias entre os itens lexicais, os exemplares prototípicos e periféricos das categorias seja muito custoso para o cérebro humano, de modo que exemplos como os descritos acima acontecem com espantosa frequência em *Codenames*.

### 5.2.7 Dica: coroa

*Coroa* é uma palavra prototipicamente relacionada a *rainha*. Mas há aqui a exploração da polissemia presente na palavra *coroa* como um objeto que está muito bem associado à *rainha* e à *bicicleta*, já que é um componente indispensável em qualquer bicicleta, portanto há uma relação meronímica entre os itens lexicais. No entanto, essa relação não está presente quando o item é usado como gíria e como vocativo por uma geração ao se referir aos seus pais como “coroas”, pessoas abaixo

dos 20 anos hoje, talvez desconheçam este sentido possível.

Com o exemplo, percebemos que é necessário que os jogadores compartilhem uma representação mental específica e tenham um conhecimento comum sobre a polissemia da palavra e suas variadas acepções.

### 5.2.8 Dica: chocolate

A partir da dica *chocolate* a Jogadora 1 chegou com rapidez às palavras-alvo *Páscoa*, *feriado* e *gota*, nesta ordem. Perguntamos à ela:

- Como você chegou às palavras-alvo?

“*Páscoa* me levou de forma muito intuitiva a *feriado*, em seguida escolhi *gota* porque tem aquelas gotas de chocolate”

Esta é uma estratégia delicada, aqui serviu muito bem ao *Guesser* e ao *Codemaster* que pôde prever e traçar uma associação a partir não só do insumo da dica, mas da primeira palavra adivinhada. A dica escolhida pelo *Codemaster* representa bem o processo de *mindereading* na medida em que o jogador pôde prever um caminho associativo possível e provável de ser traçado pelo *Guesser*, que, em contrapartida apreendeu também muito bem a lógica associativa do *Codemaster*.

Mas esta é uma estratégia frágil em alguns casos, porque abre o leque de associações plausíveis, e pode levar a caminhos equivocados, já que, perder o foco da dica principal pode anular a possibilidade de polissemia do item lexical utilizado como dica primeiramente. Além disso, é preciso que, como neste caso, a palavra-alvo que servirá de insumo seja muito óbvia e fácil de ser acertada, e que não se confunda com as outras palavras dispostas no tabuleiro, pois o caminho associativo passa a ter uma linearidade, com uma ordem bem definida para ter sucesso.

### 5.3 Conclusões sobre os resultados

Não sendo possível modelar computacionalmente todo o processo cognitivo humano envolvido na tarefa e criar um algoritmo que espelhe o modo humano de jogar, nos interessa investigar como um modelo computacional pode se aproximar, ou até mesmo superar, o desempenho humano, a partir da otimização do modelo de linguagem e das estratégias utilizadas. Está claro que, apesar de não desempenhar muitas das operações descritas, um modelo de processamento de linguagem natural pode nos oferecer resultados interessantes para a tarefa a partir da similaridade semântica e da distância vetorial entre as palavras em um modelo de *embeddings*. Podemos tirar algumas conclusões sobre as estratégias necessária para o sucesso:

- É crucial a minimização da distância entre a dica e as palavras-alvo, mas também a maximização entre a dica e as outras palavras do tabuleiro. Isso se aplica a todos os casos.
- O acesso a um conjunto robusto de itens lexicais é importante para considerar a polissemia e as associações lexicais mais periféricas
- As palavras-alvo já adivinhadas podem ter o seu sentido e suas associações possíveis somadas à dica inicial, para servir de insumo para a descoberta das outras palavras. Esta estratégia não se aplica a todos os casos, mas serve bem a muitos exemplos.
- A lógica difusa e as associações fracas entre as palavras são cruciais para uma boa taxa de acerto

Experimentamos a modelagem computacional de acordo com as estratégias que pudemos mapear e, se por um lado os jogadores humanos se beneficiam das abstrações, inferências e fenômenos pragmáticos que podem emergir nas situações do jogo, por outro lado, é imprescindível que exploremos as vantagens que o modelo

artificial tem em comparação com os processos humanos: o acesso à memória e a uma quantidade de itens lexicais é imensamente maior, o cálculo preciso das distâncias que devem ser minimizadas e maximizadas não só é mais preciso como não será nunca negligenciado. Estas são as principais assimetrias que identificamos a partir dos dados e que nos beneficiaram na elaboração do modelo que apresentaremos no capítulo seguinte.

# Capítulo 6

## *Natural Language Understanding*

Depois de percorrermos um caminho teórico linguístico e de discutirmos as bases computacionais para o nosso trabalho, gostaríamos de propor um breve capítulo sobre *Natural Language Understanding*, ou NLU. Discutiremos algumas questões teóricas e os limites do entendimento de uma língua ambígua - como são as línguas naturais humanas - para uma máquina.

### 6.1 Algumas reflexões em NLU

Dentro das reflexões que podemos propor em ML, as relacionadas ao NLU estão dentre aquelas de maior complexidade. Afinal, não podemos afirmar que seja possível falar em entendimento, compreensão semântica ou pragmática por uma máquina. As autoras de [Bender et al., 2021] são bastante enfáticas e implacáveis ao afirmar que, pelo fato de a comunicação humana depender dos significados implícitos, os quais não estão presentes nos dados de treino, os textos gerados de forma automática, por modelos de linguagem, não têm intenção comunicativa, leitura de estado mental ou modelo conceitual. Se considerarmos modelos como o GPT-3 [Brown et al., 2020], estado da arte hoje, um gerador de textos extremamente verossímeis, muito próximo de produções humanas, o que as autoras colocam parece ser um argumento falho. No

entanto, elas pontuam a necessidade de levarmos em conta o fato de que nossa compreensão da linguagem é mediada por nossa própria competência linguística, portanto, temos uma predisposição para interpretar atos comunicativos como coerentes, mas o que os sistemas fazem é simplesmente costurar sequências de formas linguísticas observadas nos dados de treinamento, probabilisticamente, mas sem qualquer referência ao significado.

De fato, quando projetamos modelos de linguagem ou *embeddings*, o nosso esforço está em encontrar uma função ideal da linguagem humana para conseguirmos representar a forma dos significados e das relações lexicais, mas isso será sempre uma projeção, estamos a emular significados que só poderão ser estabelecidos na interação humana. Além disso, as autoras colocam em questão os modelos de linguagem e sua capacidade de amplificar vieses e outros problemas presentes nos dados de treino, além da tendência de se associar um ganho de performance do modelo de linguagem, comparado com suas próprias métricas, com um consistente entendimento de língua natural. Considerando os desafios encontrados em nossa tarefa específica, poderíamos acrescentar o apontamento de um outro viés dos modelos de linguagem mais populares e estado da arte em NLP: *Bert*, *Glove*, *word2vec* etc. Nos modelos pré-treinados com esses algoritmos, fica evidente a carência de dados de fala na elaboração dos *embeddings*. Os *datasets* utilizados são majoritariamente compostos de dados de escrita. Ora, sabemos que a escrita é uma modalidade secundária de manifestação da linguagem humana; não só produzimos mais dados orais, como a oralidade precede a escrita, cognitivamente, além do fato de que uma miríade de línguas humanas prescindem da escrita, então devemos nos colocar um questionamento: como pretendemos representar e dar conta de parte do conhecimento linguístico humano se prescindimos de dados de fala em nossos principais modelos? Acreditamos que incorporados aos *embeddings*, os dados de fala poderiam criar novos paradigmas e modelos mais sofisticados.

Em uma situação comunicativa, temos pelo menos 4 elementos envolvidos: o

enunciador, o enunciatário, o enunciado e a enunciação, segundo [Benveniste, 1970]. Mas, em um modelo de NLP o que temos de material para trabalhar é estritamente o enunciado, que é o único material linguístico disponível nos dados. Entendemos que neste ponto reside o limite dos nossos modelos, portanto a nossa investigação é: o que podemos extrair de conhecimento linguístico a partir desse único parâmetro disponível? De que maneira podemos combinar os nossos dados e processos para encontrar uma forma que se aproxime das interações humanas no jogo *Codenames* e resolva o problema com algum grau de precisão? Discutiremos um pouco mais sobre essa investigação no próximo tópico.

## 6.2 NLU em *Codenames*

Computadores são muito bons em resolver problemas com formas fixas, precisas, que não envolvem qualquer tipo de ambiguidade. Se abrirmos agora o nosso aplicativo *Maps* da *Google* e buscarmos por uma rota entre Belo Horizonte e Belém, em poucos segundos o algoritmo nos retornará um caminho ideal, considerando a otimização do tempo e da distância. Embora seja uma tarefa difícil, já que as duas cidades estão bastante distantes, há bilhões de ruas e vias entre elas, os caminhos possíveis para se chegar do ponto A ao B que determinamos também são muitos. No entanto há uma resposta certa para o problema que colocamos para o algoritmo: ou é possível chegar de Belo Horizonte a Belém de carro, ou não é. Tanto é assim que se buscarmos uma rota terrestre para Nova Delhi, o algoritmo do aplicativo não retornará um resultado. Além disso, a busca por uma função ideal para a tarefa não esbarra em ambiguidades, ao contrário, é precisa, numérica e, assim, com frequência, nos retorna não mais que duas ou três opções de rotas: uma com o trajeto mais curto, outra com o trajeto mais rápido e outra com menos pedágios, por exemplo.

A tarefa que propomos em *Codenames* não é um problema como o do *Maps*, não temos uma resposta precisa para a tarefa do *Codemaster*. Além da relação entre as

palavras, a proximidade que compartilham nos dados de *input* e as estratégias que estabelecemos com nosso algoritmo, nada mais serve de pista para o que seria uma dica ideal, ou seja: o problema não é discreto. Em suma, como deve ter ficado claro para o leitor ao longo desta pesquisa, a tarefa envolve uma considerável gradiência, e o que para os jogadores humanos não é trivial, para uma máquina é, de fato, uma tarefa colossal. Máquinas representam muitas coisas com extrema precisão: as formas dos números, das cores, dos caracteres utilizados em diferentes línguas, sons etc, afinal estas são representações de formas bem definidas, discretas. Para uma máquina o caractere *A* é representado como 01000001, o *B* como 01000010 e, a partir de tais representações binárias, temos na tela do nosso editor de texto o *output* do caractere que desejamos. Mas, quando o *output* que buscamos é uma abstração, a pergunta que se coloca é: como nos aproximar de uma boa representação dos significados possíveis dentro de um sistema linguístico?

O exemplo dos mapas se encontra no que conhecemos por *Weak AI*, que são os modelos computacionais que conseguem realizar tarefas com maior eficiência que os humanos em geral, mas que ainda estão sob alguma supervisão do programador. Para uma mesma entrada, a saída do algoritmo vai ser sempre a mesma. Já em modelos conhecidos como *strong AI*, ou AI geral, temos um sistema que tem certo tipo de iniciativa, de voluntariedade e não está sob completa supervisão. Assim, temos saídas imprevisíveis e muitas vezes diferentes para uma mesma entrada. Esse tipo de modelo é o que dispomos de melhor em AI hoje para representar fenômenos como as associações lexicais. Nos *embeddings*, dispomos de vetores que representam as distâncias entre as palavras; um formalismo para a distribuição semântica dentro de um sistema linguístico a partir de seus contextos de uso, observados nos dados de treinamento. Devemos, portanto, potencializar a nossa modelagem e processos, e escolher bem as palavras que estarão em nosso tabuleiro, para tentar encontrar uma heurística ideal para a tarefa.

Considerando que as propostas da *Theory of Mind* (ToM) [Carruthers & Smith, 1996]

descrevem características cognitivas e sociais elementares para a interação humana, nos permitindo fazer suposições sobre o estado mental de nossos interlocutores por meio de pistas gestuais ou verbais, pesquisadores como [Devin & Alami, 2016] e [Peters, 2005] experimentaram incorporar seus postulados em modelos computacionais, com a intenção de melhorar a qualidade de agentes conversacionais e robôs na interação com humanos. Já [Wang et al., 2021] sugerem uma análise das percepções de estudantes sobre um professor assistente virtual a partir da introdução de parâmetros linguísticos específicos, derivados também das propostas da ToM. A proposta dos pesquisadores é inovadora ao projetar uma estrutura que se adapta às mudanças da percepção do usuário a partir de marcas linguísticas. Um modelo simplificado de *Codenames* foi avaliado por [Shen et al., 2018], os pesquisadores elaboraram uma variedade de medidas de associação semântica e conduziram experimentos para testar qual a principal estratégia para extração de informação utilizada pelos humanos. Em seguida, combinaram essas medidas com o *Rational Speech Acts Model* (RSA) [Frank & Goodman, 2012] - um modelo que, apesar de sua projeção pouco realista das interações linguísticas humanas, faz uma tentativa de modelagem de alguns processos pragmáticos - para tentar derivar previsões sobre o comportamento pragmático envolvido na tarefa. Os achados de [Shen et al., 2018] afirmam a importância da alta frequência de ocorrência de palavras como preditores da escolha lexical, sugerindo que as estatísticas de coocorrência direta são especialmente salientes em ambientes associativos como o dos jogos. Os resultados apresentados, com preponderância de importância para a coocorrência, têm relação direta com a simplificação da tarefa que propuseram os autores. Em uma modelagem completa das tarefas envolvida no jogo, certamente a análise de estratégias linguística utilizadas aponta para a relevância de mecanismos de associação paralelos, como pudemos observar em nossa coleta de dados de humanos.

De forma geral, os estudos em NLU ressaltam a necessidade de uma composição por diferentes abordagens, que considere mais de uma característica como parâmetro para

as decisões algorítmicas. A semântica de uma língua estará sempre em interface com a pragmática, e os significados atribuídos aos itens lexicais são múltiplos e dinâmicos, exigindo uma abrangência maior na estruturação dos modelos de linguagem.

Explicitaremos as decisões tomadas em nosso trabalho e o *design* dos processos no capítulo que segue.

# Capítulo 7

## Modelo Computacional

A partir do que aprendemos com os dados de jogos entre humanos e das reflexões feitas com base em teorias linguísticas e cognitivas, arquitetamos um modelo computacional para *Codenames*, com foco especial no *Codemaster*. Colocamos o nosso foco sobre o *Codemaster*, pela criatividade que demanda a tarefa deste agente que deve - além de trabalhar com relações semânticas complexas - produzir dicas que se adequem ao contexto produzido pelo tabuleiro do jogo, e buscar uma única dica no espaço de todo o léxico disponível. O *Guesser* também executa um trabalho de *mindreading*, inferências e associações, considerável, no entanto, do ponto de vista de processamento linguístico computacional, a produção da dica ainda é algo mais exigente, considerando o tamanho do espaço de busca de cada agente, e a adição, por aquele que fornece a dica, de uma informação que não está previamente no tabuleiro. Os dados de nossos informantes nos levaram primeiramente ao entendimento de que um modelo de *embeddings*, sozinho, não nos levaria a resultados satisfatórios para o jogo. Para esta conclusão, nos embasamos também nos resultados dos modelos desenvolvidos por [Kim et al., 2019] e [Somers, 2020]. Percebemos que escolher a palavra que esteja mais próxima de duas ou três palavras-alvo no espaço vetorial do *embedding* como dica é uma boa estratégia e um passo importante para o sucesso no

jogo, mas utilizado isoladamente, não é suficiente para resolver, com uma boa taxa de acerto, todos os exemplos. Assim, em busca de melhores resultados, tomamos a decisão de sofisticar o nosso modelo com base na modelagem apresentada por [Koyyalagunta et al., 2021], que conseguiu os melhores resultados até agora para o jogo desenvolvido na língua inglesa.

Segundo [Divjak & Arppe, 2013], cognitivamente, nós, humanos, temos um sistema lexical simples, fácil de aprender, muito frequente e vago, limitado quantitativamente pela capacidade de memória, e um sistema contextual complexo, este sim: difuso e carregado de exceções. Como exemplo, os pesquisadores citam os estudantes de uma L2, para quem a apreensão lexical não é o maior desafio, mas sim o aprendizado do sistema contextual. Nos apoiamos em suas ideias e acreditamos que, também para os modelos automáticos, os desafios de aprendizado são similares, posto que os itens lexicais estão bem catalogados, estabelecidos no sistema, mas usá-los e combiná-los referencialmente é o principal desafio.

Entendemos ainda que todo modelo é, por definição, uma simplificação, portanto, buscamos combinar os elementos cognitivos que identificamos em nosso estudo com os jogadores humanos aos algoritmos e às técnicas que são o estado da arte em ML.

Como demonstramos em nossa pesquisa, o conhecimento linguístico se diferencia de outros tipos de conhecimento, talvez justamente por envolver processos diversos, que somam suas complexidades e criam soluções não lineares, difíceis de serem apreendidas e simuladas pelos algoritmos que conhecemos até hoje. Apesar disso, podemos afirmar que alcançamos resultados bastante satisfatórios, tendo em vista a complexidade da tarefa e os recursos disponíveis para o pt-BR.

## 7.1 Estrutura do Modelo Computacional

A estrutura de nosso modelo computacional se baseia no que foi apresentado em [Koyyalagunta et al., 2021], os dados e algoritmos utilizados pelas autoras podem

ser encontrados no *GitHub*<sup>1</sup>. Nossa proposta consiste em uma adaptação completa da arquitetura para o pt-BR, com alguns ajustes que consideramos relevantes, a disponibilização de novos recursos e uma análise qualitativa dos resultados gerados.

Detalharemos mais os processos de implementação neste capítulo.

Todos os códigos e processamentos para o modelo adaptado por nós estão disponíveis em nosso repositório, também no *GitHub*<sup>2</sup>. Todos os processos e dados disponibilizados por [Koyyalagunta et al., 2021] foram completamente adaptados para o pt-BR: as rotinas de pré processamento, os *embeddings* utilizados, os dados linguísticos utilizados como insumo para as funções do algoritmo, as palavras do tabuleiro e as dicas possíveis.

## 7.2 Insumos Linguísticos

### 7.2.1 Corpus para o Modelo Computacional

Para treinar o nosso modelo computacional utilizamos o *embedding* disponibilizado pelo Núcleo Interinstitucional de Linguística Computacional (NILC), o grupo possui um repositório destinado ao armazenamento e compartilhamento de vetores de palavras gerados para a Língua Portuguesa. O repositório traz vetores gerados a partir de um grande *corpus* do português brasileiro, de fontes e gêneros variados<sup>3</sup>. Os vetores disponibilizados são gerados a partir de diferentes algoritmos; os vetores utilizados em nosso trabalho foram gerados pelo algoritmo *Word2vec*, *Skip-Gram* de 300 dimensões. O *corpus* de treinamento para o *Word2vec* conta com um total de 1,395,926,282 *tokens*. As rotinas de pré-processamento dos *corpora* utilizados na geração dos *embeddings* podem ser verificadas no *GitHub*<sup>4</sup> do grupo. Elencamos o *Word2vec* como nosso *embedding* principal para o estudo, pelo fato de este ser uma

---

<sup>1</sup><https://github.com/divyakoyy/codenames>

<sup>2</sup><https://github.com/Robertaviola/codenames>

<sup>3</sup><http://nilc.icmc.usp.br/nilc/index.php/repositorio-de-word-embeddings-do-nilc>

<sup>4</sup><https://github.com/nathanshartmann>

técnica mais simples e eficaz para tarefas como a nossa, diferentemente de outros algoritmos como o *BERT* ou *GloVe* que, apesar de serem algoritmos mais novos e mais sofisticados na abordagem contextual, apresentaram um grau de instabilidade maior em nossos testes iniciais com o pt-BR. Além disso, os resultados de performance relatados por [Koyyalagunta et al., 2021] para o *Word2vec* foram, também, bastante robustos.

## 7.2.2 Tabuleiro

Seguimos a premissa das autoras em [Koyyalagunta et al., 2021], de que as cartas disponíveis para compor os tabuleiros do jogo analógico oficial *Codenames*, desenvolvido pelo designer Vlada Chvátíl, são idealmente as cartas que devem compor os tabuleiros para o modelo computacional. As palavras selecionadas pelo *designer* do jogo foram previamente estudadas e cuidadosamente selecionadas a partir do julgamento de possíveis inter-relações e multiplicidade de significados. Em nosso estudo, ampliamos a quantidade de palavras disponíveis e as igualamos às do jogo original. [Koyyalagunta et al., 2021] utilizam uma lista de 208 palavras em inglês para a geração de um tabuleiro com 20 palavras; seguindo o *design* do jogo original publicado no Brasil, propusemos uma lista de 400 palavras em pt-BR para a geração de tabuleiros compostos por 25 palavras.

## 7.2.3 Possíveis Dicas

A tarefa do *Codemaster* envolve uma busca em todas as palavras disponíveis para, assim, identificar possíveis dicas e, em seguida, classificá-las para selecionar a mais adequada. Considerando as estratégias dos jogadores humanos e o léxico mental disponível na memória e na ativação a partir dos insumos presentes no tabuleiro, propusemos uma nova abordagem de estruturação do inventário de palavras disponíveis para a dica.

Ora, entendemos que disponibilizar todas as palavras da língua como possíveis dicas para o *Codemaster* não seria eficiente, tampouco se assemelharia às operações dos jogadores humanos. Assim, em uma tentativa de emular as estratégias relatadas por nosso jogadores, propusemos uma expansão das palavras presentes nas cartas do tabuleiro combinadas a outras palavras, geradas de forma aleatória, mas com um grau de frequência controlado. Chegamos a uma lista de 4750 palavras para possíveis dicas da seguinte forma:

- Cálculo das distâncias (cosseno) e adição das 5 palavras mais próximas a cada uma das palavras do tabuleiro
- Inserção randômica de palavras que não constassem anteriormente em nossa lista, e verificação da plausibilidade com o algoritmo TF-IDF (*Term Frequency-Inverse Document Frequency*).

O primeiro passo do desenvolvimento de nossa lista de dicas possíveis tem o intuito de dar especificidade a elas e reduzir o espaço de busca. Já o segundo passo parte da intenção de agregar aleatoriedade às palavras, a fim de possibilitar a inserção de dicas com conexões mais difusas. E o controle de frequência de ocorrência na língua serve para não criar opções de dicas muito desviantes.

Ressaltamos que essa é uma inovação nossa, que acrescentamos ao modelo de [Koyyalagunta et al., 2021].

#### 7.2.4 Dict2vec

Em [Mollin, 2009], a autora demonstra que as coocorrências de palavras em *corpora* divergem significativamente quando comparados aos dados de tarefas de associações de palavras. Assim, entendemos que somente a coocorrência, ou o uso frequente de palavras em contextos similares, não são suficientes para tratar do problema envolvido em nosso jogo. Um exemplo mais preciso estaria entre os sinônimos: uma

boa dica para a palavra-alvo *computador* poderia ser *notebook*, por exemplo, no entanto, pelo grau de sinonímia que carregam, é pouco provável que tais palavras tenham grande coocorrência. Já em um fenômeno de meronímia como *cachorro* e *mamífero*, a probabilidade de captura desta relação pelos métodos de incorporação contextual é baixa, dado que a coocorrência entre os dois itens não é expressiva.

Uma boa estratégia para agregar outros fenômenos associativos, como a meronímia, hiperonímia, sinonímia e as relações semânticas elementares, está na incorporação da lógica de verbetes dos dicionários aos dados. [Koyyalagunta et al., 2021] reconhecem esta importância e dissertam sobre as limitações dos *embeddings*; segundo as autoras, as abordagens que utilizam o contexto de uso para a composição das distâncias vetoriais entre as palavras não têm nenhuma restrição ao colocar palavras muito raras próximas a palavras muito frequentes, e alguns relacionamentos importantes como a meronímia e hiperonímia são difíceis de serem capturados contextualmente.

Ao elaborarmos o nosso modelo computacional, fizemos uma busca por um *embedding* feito a partir de dicionários do pt-BR, não encontramos esses dados disponíveis e, até a data de elaboração deste trabalho, não é do nosso conhecimento algum outro estudo que os tenha disponibilizados. Assim, entendemos que o desenvolvimento de nosso *embedding* de dicionário, feito a partir do algoritmo *Dict2Vec* [Tissier et al., 2017], figura como uma contribuição para os recursos disponíveis para o pt-BR em NLP. Para a elaboração de nosso *Dict2Vec*, utilizamos os dados abertos em português do *Wikcionário*<sup>5</sup>. Nosso *embedding* está disponível <sup>6</sup> e pode ser baixado para uso em futuras pesquisas.

---

<sup>5</sup><https://pt.wiktionary.org/>

<sup>6</sup>[https://drive.google.com/drive/folders/1hAmRYGp\\_GfTr3QuZpiTiU9Gnu8rudCvw?usp=sharing](https://drive.google.com/drive/folders/1hAmRYGp_GfTr3QuZpiTiU9Gnu8rudCvw?usp=sharing)

### 7.2.5 DETECT

Nos experimentos de produção de dicas a partir das distâncias vetoriais, [Koyyalagunta et al., 2021] identificaram três tipos frequentes de problemas: dicas muito obscuras, pistas excessivamente genéricas e a ausência de pistas com relações entre palavras a partir do conhecimento de mundo. Em busca de melhorar o modelo nesses aspectos, as autoras desenvolveram uma métrica de pontuação nomeada *DETECT*. Pistas obscuras são aquelas que, apesar de não estarem incorretas, não emulam um conhecimento comum, são dicas muito especializadas: como dar a dica *fonologia* para as palavras *professor* e *linguística* - esta seria uma excelente dica, mas só os especialistas compreenderiam e obteriam sucesso no jogo. Já as dicas excessivamente genéricas, são aquelas que, apesar de também parecerem bastante boas à primeira vista, são palavras que podem se relacionar a muitos distratores presentes no tabuleiro e induzir ao erro. As relações mais expandidas entre as palavras que não têm alta coocorrência no *corpus* podem ser buscadas entre itens lexicais que figuram respectivamente em suas definições de dicionário. Como exemplo, as autoras citam *carro* e *veículo*, apesar de estarem relativamente distantes nos *embeddings*, estes dois itens formam um par forte, a partir de suas definições no dicionário, já que *veículo* é um item presente na definição de *carro* e vice-versa. A qualidade das pistas pode ser aprimorada, e os problemas citados bastante reduzidos a partir da implementação da métrica de pontuação *DETECT*, tanto para o modelo desenvolvido em língua inglesa, quanto em nossa adaptação para o pt-BR. A métrica é composta por dois passos: um em que a função *FREQ* utiliza a frequência de palavras em um *corpus* para excluir aquelas palavras que sejam ou muito raras, ou muito frequentes. Utilizamos um *corpus* da *Wikipédia* em pt-BR com 50 milhões de palavras, fizemos um pré-processamento desse dado e a remoção das *stopwords*. Em seguida, a função *FREQ* faz a contagem das ocorrências da palavra e atribui uma penalização maior para aquelas palavras muito raras, uma penalização um pouco

menor para palavras comuns, e uma penalização também mais relevante para aquelas palavras extremamente comuns. As autoras ressaltam que em [Jaramillo et al., 2020] a frequência de ocorrência foi utilizada como um método direto de seleção da dica, mas aqui a abordagem é mais sofisticada, na medida em que a frequência é incorporada como um dos parâmetros de uma função maior para seleção da dica. A métrica *DICT* atribui um peso maior às palavras que tenham forte relação em nosso *embedding* de dicionário. A sua definição se estabelece pela distância do cosseno entre os vetores de palavras em nosso *Dict2Vec*. Portanto, combinadas, as métricas *FREQ* e *DICT*, formam a função de pontuação *DETECT*, que permite produzir dicas semanticamente relacionadas de maneiras que o contexto dos *embeddings* sozinho não é capaz de capturar.

### 7.3 Codemaster

[Koyyalagunta et al., 2021] elaboraram um algoritmo chamado *ClueGiver* para gerar dicas a partir da similaridade entre palavras. O algoritmo tem duas etapas principais: a primeira, em que calcula as palavras vizinhas mais próximas de cada palavra-alvo no tabuleiro e as adiciona como candidatas à dica. Na segunda etapa, à cada palavra candidata à dica é atribuída uma pontuação, essa pontuação é calculada a partir de uma função que atribui um alto valor para aquelas pistas que têm proximidade com as palavras-alvo e ao mesmo tempo uma boa distância das outras palavras dispostas no tabuleiro.

A decisão entre priorizar as dicas que estão mais perto de palavras azuis ou priorizar as dicas que estão longe de palavras vermelhas na escolha da dica final foi tomada empiricamente pelas autoras, a partir da função de proximidade entre palavras-alvo = 1 e proximidade com outras palavras do tabuleiro = 0,5 que foi a que se mostrou mais efetiva nos testes que produziram. A principal candidata à dica é a palavra com maior pontuação para a função *ClueGiver*. O agente que representa o *Codemaster*,

enfim, fornece as dicas a partir da combinação das funções *ClueGiver* e *DETECT*.

No capítulo seguinte, apresentaremos os resultados gerados pelo modelo computacional para o pt-BR.

## Capítulo 8

# Resultados do Modelo Computacional

As pesquisas de modelos computacionais para *Codenames* têm se concentrado em avaliar os agentes desenvolvidos a partir de métricas quantitativas, como em [Jaramillo et al., 2020] e [Kim et al., 2019]. Em geral, avalia-se a performance pelo número de jogos vencidos e pelo número de rodadas até o fim do jogo. Esse tipo de abordagem, pode acabar indicando somente o quão bem alinhados estão os *embeddings* utilizados e os agentes desenvolvidos. Do ponto de vista linguístico, entendemos que uma avaliação mais interessante para o desempenho do nosso modelo deve ser qualitativa, para que possamos observar se há algum tipo de inteligência sendo gerada, principalmente a partir das dicas fornecidas pelo *Codemaster*. [Koyalagunta et al., 2021] também propuseram uma avaliação qualitativa de exemplos computacionais, as pesquisadoras validaram o modelo computacional que desenvolveram e a qualidade das dicas a partir do julgamento por avaliadores humanos de 1400 exemplos.

Como enfatizamos ao longo de nossa pesquisa, todo episódio de interação linguística é um processo, não envolve só a recuperação de um item lexical na memória, já que a

formação de conceitos exige processos extra linguísticos relativos à cognição e à percepção geral, que é mediada pela pragmática e pelo contexto. Habitualmente, a sintaxe é a responsável por codificar as conexões relacionadas com aquilo sobre o que se fala, mas quando não há o intermédio da sintaxe no insumo que utilizamos e temos apenas os itens lexicais isoladamente, essas conexões devem ser inferidas, imaginadas e, assim, se dá a expansão dos significados e as possibilidades associativas, pois que, se o inventário lexical na memória de um falante é limitado, as expressões e combinações possíveis são ilimitadas e renovadas a cada novo evento linguístico. Se mesmo os humanos - apesar da produção de exemplos interessantes e criativos - muitas vezes não têm boas métricas de resultados quantitativos nos jogos e chegam mesmo a perder várias partidas, nos interessa mais avaliar, como dito pela pesquisadora Dagmar Divjak em palestra virtual<sup>1</sup>, não o melhor modelo em termos de desempenho, mas o que seja cognitivamente mais plausível. Apresentamos aqui alguns exemplos gerados pelo nosso modelo computacional para *Codenames*. Os exemplos apresentados serão proveitosos para a análise qualitativa e comparativa com os exemplos produzidos por jogadores humanos.

## 8.1 Exemplos

### 8.1.1 Exemplos para duas palavras-alvo

A tabela que apresentamos aqui traz exemplos de dicas para duas palavras-alvo. Nosso modelo gerou uma alta quantidade de dicas satisfatórias para este número de palavras-alvo. Os resultados para o modelo desenvolvido em língua inglesa por [Koyyalagunta et al., 2021], também apresentou bons resultados para exemplos com duas palavras-alvo, alcançando 50% de acerto das palavras-alvo na validação dos exemplos por humanos, com a dica fornecida pelo modelo utilizando *Word2vec*, mas

---

<sup>1</sup><https://www.youtube.com/watch?v=jTF5ochiNtU>

sem a função *DETECT*, e 60% de precisão ao se incorporar a função *DETECT* ao modelo. Foram avaliados, no total para diferentes *embeddings*, uma amostra de 1400 dicas.

Apresentamos aqui uma tabela com alguns exemplos que consideramos mais relevantes, gerados pelo nosso modelo computacional. Disponibilizamos, também, um arquivo *.csv*<sup>2</sup> com mais de um milhão de exemplos de dicas para duas palavras-alvo geradas pelo nosso modelo computacional em pt-BR.

**Tabela 8.1.** Exemplos Gerados Computacionalmente para 2 Palavras

Palavras-alvo	Dica
cientista, lua	Nasa
alfabeto, agente	FBI
princesa, morte	Diana
flauta, computador	piano
estrela, cassino	Hollywood
igreja, enfermeira	Maria
pincel, cola	verniz
dragão, brilhante	unicórnio
pata, elefante	rinoceronte
vaso, triângulo	funil
igreja, fechadura	sacristia
morte, cólera	agonia
moscou, cerca	leningrado
cupim, piano	fagote
sorvete, banana	iogurte
poder, olimpo	Hades

<sup>2</sup>[https://drive.google.com/drive/folders/1hAmRYGp\\_GfTr3QuZpiTiU9Gnu8rudCvw?usp=sharing](https://drive.google.com/drive/folders/1hAmRYGp_GfTr3QuZpiTiU9Gnu8rudCvw?usp=sharing)

braço, lutador	boxeador
cobre, vidro	estanho
veneno, verme	parasita
aprendiz, fantasma	feiticeiro
modelo, rascunho	layout
veneno, comprimido	letal
ouro, diamante	cravejado
magia, herói	bruxo
seda, parede	forrada
trave, volante	centroavante
cientista, ninja	ciborgue
luz, onda	infravermelha
trave, bola	chutar
dinossauro, lesma	salamandra
furo, carro	pneu
recreio, chocolate	biscoitos
polícia, carro	viatura
roma, europa	crisandade
aprendiz, hospital	internato
alma, anjo	imortalidade
água, cola	pepsi
esquerda, alemanha	rfa
pedra, magia	filosofal
servidor, arquivo	html
violão, concerto	recital
gato, terno	bigode
saturno, magia	luas

paraquedas, terno	colete
vento, neve	granizo
peixe, espinha	barbatana
política, cientista	linguista
mato, Amazonas	Xingu
cobertura, paraquedas	aterragem
cruz, bruxa	feiticeira

### 8.1.2 Exemplos para três palavras-alvo

A combinação de três palavras-alvo para uma dica torna a tarefa exponencialmente mais complexa. Os experimentos realizados com nosso modelo em pt-BR demonstram que a capacidade de gerar boas dicas ao adicionarmos mais uma palavra-alvo é drasticamente reduzida. No entanto, conseguimos, ainda assim, coletar alguns exemplos razoáveis, que talvez possam gerar algum tipo de associação difusa na mente do falante, mas ressaltamos que a sua ocorrência é rara e muitas vezes as associações lexicais são bastante questionáveis quanto à capacidade de levar ao acerto. Devemos sempre lembrar que o processamento de uma dica está restrita ao contexto fornecido pelo tabuleiro, reduzindo o campo de busca associativa, assim, uma conexão que à primeira vista possa nos parecer improvável, se torna possível e adequada dado o contexto do jogo.

Em uma coleta de 100 exemplos, conseguimos coletar uma média de 9 exemplos que apresentam alguma conexão associativa. [Koyyalagunta et al., 2021] não apresentaram experimentos para a combinação de três palavras-alvo.

Palavras-alvo	Dica
terno, paraquedas, pistola	colete
vaso, lava, centauro	hércules
medida, polícia, política	judiciária
parada, hotel, banheiro	motel
dedo, prato, pedra	garfo
vela, mastro, boca	proa
prato, sorvete, doce	bolinho
água, churrasco, cano	balde
balanço, som, fluxo	medidor

## 8.2 Análises dos Resultados

Consideramos ter alcançado resultados bastante satisfatórios; temos uma boa amostra de exemplos em que é possível observar uma plausibilidade cognitiva para as associações, podemos afirmar que geramos algum tipo de aprendizado que levou a muitos exemplares com resultados similares aos coletados com jogadores humanos. Analisaremos aqui alguns exemplos e os compararemos com os dados coletados entre os jogadores humanos.

Enfatizamos que a explicabilidade dos processos dos algoritmos de ML é um campo de pesquisa vasto e complexo. Não podemos afirmar precisamente sobre os pormenores e as motivações da tomada de decisões desse tipo de algoritmo.

Entretanto, a nossa intuição como linguistas, combinada ao que conhecemos sobre a estrutura dos dados linguísticos fornecidos como insumos, pode nos levar a algumas hipóteses na análise da estruturação das associações propostas e da escolha das dicas fornecidas pelo modelo computacional.

Ainda que tenhamos uma boa amostra de associações lexicais bem sucedidas para o nosso modelo em pt-BR, observamos em nossa pesquisa uma certa desvantagem em relação ao número de exemplos ótimos gerados para a língua inglesa. Colocamos esta observação como um adendo à nossa pesquisa, e ressaltamos que é uma avaliação empírica, que deve ser checada e explorada em estudos futuros, já que não dispomos

de recursos para a validação e comparação de uma amostra robusta entre modelos de diferentes línguas por avaliadores humanos.

### 8.2.1 Dica: Maria

Nosso modelo computacional forneceu a dica *Maria* para as palavras-alvo *igreja* e *enfermeira*. *Maria* traça uma associação clara com *igreja*, já que a *Santa Maria*, ou *Maria Mãe de Deus*, é uma figura central no catolicismo. Com *enfermeira* a relação associativa é um pouco mais difusa, mas o gênero gramatical feminino apontado pelos dois itens acaba por favorecer o acerto, caso não haja competição com outro item lexical no tabuleiro, algum que também aponte, necessariamente, para o gênero gramatical feminino. A dica nos parece muito eficiente para levar ao acerto no jogo, mas, acima de tudo, nos chama atenção não só por ter uma associação absolutamente plausível entre os itens lexicais, mas por termos coletado um exemplo entre os nossos jogadores humanos em que foi utilizada uma estratégia muito similar.

A dica *príncipe*, fornecido por nosso Jogadora 1, para as palavras alvo *sapo*, *irmão* e *chapéu*, pode ser associada a *irmão* por ser a única palavra de gênero gramatical masculino presente no tabuleiro. No exemplo computacional, *enfermeira* também era o único item no tabuleiro que apontava para um sujeito de gênero feminino e, dado o contexto (não só das três palavras, mas de todo o tabuleiro) a dica se tornou oportuna.

### 8.2.2 Dica: funil

Como dissemos, a análise dos exemplos gerados é sempre pautada em nossa interpretação e abordagem como linguistas. Não podemos afirmar categoricamente sobre os processos que levaram o nosso algoritmo à dica *funil* para as palavras-alvo *triângulo* e *vaso*. Todavia, os três objetos têm uma forma geométrica bastante similar, a dica nos parece altamente eficaz e inusitada e provavelmente levaria a um

alto grau de acerto entre jogadores humanos. Podemos supor sobre o que em nosso algoritmo colaborou para a geração desse exemplos e, neste caso, acreditamos que possa ter algo relacionado às definições de dicionário de cada um dos itens, mas ainda assim, isso não nos parece muito óbvio quando verificamos especificamente as entradas de cada um dos itens. Vejamos as definições dos itens em nosso dicionário:

*funil: utensílio com a forma de pirâmide ou de cone invertido em cujo vértice há um tubo que serve para transvasar líquidos*

*triângulo: (Geometria) polígono de três lados*

*vaso: recipiente para líquidos, sólidos, flores etc.*

Entre *funil* e *triângulo* podemos supor uma expansão da definição e uma transferência de conceitos entre itens que tenham alguma ligação com os dois primeiros, como *cone*. Mas *vaso* nos parece deveras impressionante, só uma experiência de mundo levaria à atribuição de similaridade na forma do objeto. É verdade que o bom exemplo pode se dever ao acaso, mas observando a qualidade de outros exemplos gerados, podemos perceber que a probabilidade de haver uma boa lógica que sustente dicas como essa também é alta, todavia ainda não conseguimos destrinchar e explicar completamente os processos de muitos dos algoritmos de ML.

### 8.2.3 Dica: sacristia

A dica *sacristia* é auspiciosa na associação dos itens lexicais *igreja* e *fechadura*, que poderiam nos parecer bastante desconectados.

Em nosso dicionário temos a definição para *sacristia*: *cômodo de uma igreja onde se guardam objetos sagrados e vestimentas*. A ideia de *guardar* objetos em um cômodo anexo, que deve ser fechado para a preservação dos pertences da igreja, combinada com os dados de uso presentes em nosso *embedding*, pode ter colaborado na construção associativa proposta pelo algoritmo nesse exemplo. Entendemos a conexão estabelecida aqui como a uma associação conceitual sutil e bastante apropriada.

#### 8.2.4 Dica: fagote

Para as palavras-alvo *cupim* e *piano*, nosso modelo forneceu a dica *fagote*. A relação deste item é forte e clara com *piano*, no entanto, a relação da dica com *cupim* tem a ver com o material utilizado na construção do instrumento, que é a madeira.

Portanto, entendemos que o processo de construção dessa dica é riquíssimo, na medida em que não trabalha só as relações diretas possíveis, mas constrói os significados a partir de uma complexidade que envolve também a realidade concreta dos objetos a que se refere.

#### 8.2.5 Dica: centroavante

O item lexical *volante* é polissêmico: pode significar o *volante* de veículos, ou o *volante* do futebol, que é a posição de um jogador em campo. Aqui a polissemia foi bem resolvida e utilizada de forma eficaz pelo nosso modelo que estabeleceu uma boa relação entre a dica *centroavante*, que também indica a posição de um jogador em campo, e as palavras-alvo *trave* e *volante*, a partir do *frame* do futebol.

Esta é também uma estratégia que pode ser observada entre os jogadores humanos. É comum e bastante eficaz evocar um *frame* a partir da combinação das palavras intencionadas e de combiná-las de forma construtiva, de modo que a primeira a ser adivinhada - geralmente a que tem a relação mais forte com a dica, como em *trave* e *centroavante*, sirva como informação para contribuir na intensificação e aponte para a segunda palavra a ser acertada.

#### 8.2.6 Dica: biscoitos

Nosso modelo produziu a dica *biscoitos* para as palavras-alvo *recreio* e *chocolate*.

Essa dica nos parece muito curiosa, como já foi dito, podemos apenas supor os caminhos tomados por nosso algoritmo, dado o caráter obscuro dos processos dos algoritmos de ML, tidos como verdadeiras caixas pretas pelos pesquisadores da área.

A explicabilidade da função modelada pelos algoritmos de ML não pode ser totalmente alcançada e, em nosso modelo, isso não é diferente. Contudo, o nosso viés interpretativo para esse exemplo nos mostra uma situação que, se entre jogadores humanos, demandaria uma relação de *background* cultural e experiência compartilhada forte.

Nos anos 90, no Brasil, era comum as crianças levarem para a escola um biscoito de uma marca chamada *Recreio*, da *Turma da Mônica*, de sabor chocolate. Nesse caso, para ter sucesso no jogo com a dica fornecida, os jogadores precisariam ter, pelo menos, mais de 25 e ter vivido no Brasil no período a que a dica se refere. Além disso, seriam necessárias outras duas coisas: a recuperação desse *frame* específico na memória e, seria necessário ainda, que o jogador fosse falante de uma das variedades (ou ao menos tivesse o conhecimento) de variedades do pt-BR que utiliza o item *biscoito*, já que em algumas variedades o termo corrente é *bolacha*.

É verdade que um outro caminho associativo para esse exemplo, seria possível. O jogador poderia evocar o *frame* de um recreio escolar infantil, em que as crianças costumam levar lanches com biscoitos de chocolates; esse caminho associativo, embora possua conexões mais difusas entre os itens lexicais, demanda um *background* cultural compartilhado mais simples.

### 8.2.7 Dica: bigode

A relação entre as palavras-alvo *gato* e *terno* é bastante difícil de ser traçada. Como jogadores, precisando relacioná-las de alguma maneira, seria-nos demandada uma boa quantidade de criatividade. No entanto, o nosso modelo computacional parece ter encontrado um caminho possível para resolver o problema. A dica *bigode*, apesar de traçar uma relação bastante difusa, é algo que ativa sem muita dificuldade o item *gato*, e que tem uma probabilidade de ativar o estereótipo de um homem comum no século XIX, por exemplo, de terno e bigode.

É frequente observamos entre os jogadores humanos de *Codenames* estratégias como essa, que envolvem a busca de conexões inusitadas e difusas, que acabam por se revelar muito criativas e eficientes no contexto do tabuleiro dos jogos. Assim, acreditamos que uma dica como essa representa um grande passo para o modelo computacional.

### 8.2.8 Dica: *viatura*

O conceito de *viatura* é muito claro para nós: uma *viatura* é um *carro de polícia*. Este é o significado mais forte para os falantes. Entretanto, não há uma grande ocorrência dos itens *carro* e *viatura* nos mesmos contextos, já que, por carregarem um alto grau de sinonímia nos contextos de uso possíveis, os falantes acabam por selecionar um em detrimento do outro. Ainda, na definição de nosso dicionário, o item *viatura*, está definido da seguinte forma:

*vi.a.tu.ra: designação genérica de qualquer veículo de transporte, de carga ou de passageiros. Veículo oficial utilizado por órgão governamental (Brasil).*

Não há qualquer menção ao item *polícia* ou *carro*, que são as palavras-alvo. Mas temos o item *veículo*, que também está presente na definição de *carro*. Já nos dados de nosso *embedding*, temos a informação sobre o alto uso contextual de *viatura* e *polícia*.

Ou seja, inicialmente, não conseguiríamos traçar essa relação conceitual - que é fortemente compartilhada pelos falantes da língua - a partir dos insumos linguísticos fornecidos isoladamente e sem o tipo de processamento que é proposto pelo algoritmo que utilizamos. Assim, a dica *viatura*, ilustra bem o funcionamento da combinação dos *embeddings* contextuais de uso e de dicionário, e a potencialização da capacidade associativa que emerge desse conjunto.

### 8.2.9 Dica: unicórnio

A dica *unicórnio*, apesar de não compartilhar qualquer grau de sinonímia, hiponímia ou hiperonímia, ou ter uma definição de dicionário compartilhada com *brilhante* e *dragão*, compartilha com esses itens a presença em universos similares: da fantasia, do conto, da magia. Assim, os itens compartilham uma coocorrência em contextos similares na língua, mas, a associação não é absolutamente direta, desta forma consideramos um grande acerto do nosso modelo ao captar a sutileza dessa relação; que é ainda mais interessante por envolver dois substantivos e um adjetivo, adjetivo esse que pode acompanhar ambos os substantivos em seus contextos de ficção.

### 8.2.10 Dica: linguista

Consideramos essa dica bastante auspiciosa e, não só por afeto ao tema, vamos analisá-la.

A dica *linguista* traça uma interessante relação com as palavras-alvo *cientista* e *política*, na medida em que, de fato a linguística é uma disciplina científica e que em tantas das suas vertentes esbarra nas fronteiras das ciências políticas e, na medida em que, suas teorias e pressupostos tem forte influência na implementação de políticas públicas. No entanto, esse não nos parece um caminho associativo muito direto a ser traçado, os itens não têm forte coocorrência, tampouco estão presentes em suas respectivas entradas de dicionário, assim, podemos considerar esse um bom exemplo de sucesso do nosso modelo computacional, ainda que não possamos precisar os processos que garantiram a boa associação alcançada.

# Capítulo 9

## Conclusões

As línguas humanas não são formais, os significados não se encontram inteiramente na palavra, no texto ou na voz, mas na interação, na combinação entre a produção e a recepção, no processo comunicativo, enfim. Em um modelo formal de NLP, o que escapa, os limites do que pode ser aprendido ou executado estão justamente na falta de limites que a língua humana pressupõe, com suas infinitas possibilidades significantes.

O estudo da tarefa de associação lexical presente no jogo *Codenames* foi um facilitador que nos permitiu a exploração de uma visão aprofundada da natureza de representação presente nos modelos linguagem. Nosso estudo corrobora a noção de um léxico em constante evolução, na medida em que, além da inserção de novos itens lexicais, a proposição de novas relações entre itens produz novos significados e possibilidades dentro dos sistemas linguísticos, a partir da criatividade dos falantes. Na busca pela produção de um modelo computacional interessante para a análise, ou mesmo para ser um bom adversário no jogo, é esse tipo de processo que estamos tentando emular.

Nosso trabalho nos possibilitou a reafirmação de uma fluidez e flexibilidade, em um espalhamento uniforme da carga semântica por todo o léxico, perpassando diferentes

categorias lexicais. Entendemos que os itens lexicais compartilham significados, se completam e se expandem em combinação, assim, o significado de uma palavra não esta só nela mesma, tampouco no contexto, mas na relação geral que traça com o todo o léxico do sistema linguístico em que se insere.

Considerando as reflexões e modelagem que propusemos para os processos linguísticos envolvidos no jogo *Codenames*, criamos soluções interessantes para a simulação de conceitos abstratos e conexões difusas, que são frequentemente traçadas por humanos, mas difíceis de simular em um modelo automático. Chegamos a uma simulação muito satisfatória para o pt-BR, a partir do algoritmo desenvolvido por [Koyyalagunta et al., 2021].

### 9.0.1 Contribuições

Acreditamos ter iniciado uma discussão importante em pt-BR sobre a modelagem computacional de um jogo que envolve um forte componente linguístico, e alcançamos resultados promissores com a adaptação de recursos antes disponíveis somente para o inglês. Como observamos ao longo do estudo, a tarefa proposta envolve camadas complexas de processamento linguístico e criatividade, o que nos permitiu avançar nas discussões sobre as estruturas formais adequadas à modelagem de jogos de palavras e em um aprofundamento no entendimento das possibilidades e limitações de representação da linguagem por um *embedding*.

Disponibilizamos os dados e algoritmos utilizados em todo o desenvolvimento de nosso modelo para *Codenames* em nosso GitHub<sup>1</sup> e contribuímos com:

- embedding *Dict2vec* em pt-BR
- agente treinado como *Codemaster*
- agente treinado como *Guesser*

---

<sup>1</sup><https://github.com/Robertaviola/codenames>

- *dataset*: conjunto de um milhão de exemplares de associação de itens lexicais

O *dataset* de exemplos para o jogo que disponibilizamos é uma contribuição relevante, por ser um dado com um alto custo computacional de produção. Esperamos que esse dado possa colaborar com pesquisas futuras, que avancem em outras direções na investigação semântica, ou que proponham inovações e melhorias ao modelo computacional que apresentamos. Agradecemos ao Laboratório de Inteligência Artificial da UFMG (LIA), que disponibilizou as máquinas e toda a estrutura necessária para a execução de toda a pesquisa e coleta dos dados.

## 9.0.2 Trabalhos Futuros

Trabalhos futuros contribuirão enormemente ao utilizar pressupostos da ToM [Carruthers & Smith, 1996] para considerar os erros ou acertos do jogador adversário no desenrolar da partida, sejam esses oponentes humanos ou outros agentes computacionais. Como enfatizam os criadores da *Codenames AI Competition*, o próximo grande passo é desenvolver agentes com uma compreensão de seus companheiros na interação, ao longo de vários jogos, isso seria crucial, também, para o desenvolvimento de agentes que se comunicam melhor com as pessoas para realizar qualquer tipo de tarefa compartilhada. Essa abordagem se apropriaria do que fazem os falantes de uma língua, ao adequar o discurso de acordo com o interlocutor e com o desenvolvimento da interação, que é atualizada a todo instante.

Ademais, a manipulação mental das unidades lexicais é reveladora das arquiteturas mentais de representação linguística, assim, experimentos e modelagens que envolvam pessoas de idades, *backgrounds* e culturas diversas poderiam trazer avanços para o modelo e para as análises, considerando que os protótipos utilizados levam em conta questões culturais e de melhor exemplar. Acreditamos que um modelo computacional como o nosso poderia se beneficiar, ainda, de *datasets* como o que foi coletado e organizado com o *JeuxDeMots*, por [Lafourcade, 2007] e dos dados de outros tipos de

dicionários de associações, ou tesouros, como o *Dicionário Analógico da Língua Portuguesa* [Azevedo, 2016]. Além disso, como sugerido pela pesquisadora Dra. Valéria de Paiva em nossa banca de Mestrado em 15/10/2021, será interessante agregar o uso de ontologias na arquitetura do nosso modelo computacional, alinhado aos experimentos propostos por [Koyyalagunta et al., 2021] para *Codenames* com *BabelNet* na língua inglesa, já que trazem consigo a vantagem de interpretabilidade dos resultados, pela estrutura de grafo que constitui esse tipo de modelo.

Esperamos que outros pesquisadores possam se beneficiar de nossa pesquisa e de nossos recursos para a produção de pesquisas que contribuam com o NLP em pt-BR e que proponham uma interface agregadora entre a linguística e a computação.

# Referências Bibliográficas

- [Aitchison & Lewis, 1995] Aitchison, J. & Lewis, D. (1995). How to handle wimps: Incorporating new lexical items as an adult. 29(1-2):7--20.
- [Apperly, 2011] Apperly, I. (2011). *Mindreaders: The Cognitive Basis of Theory of Mind*. Psychology Press.
- [Austin, 1962] Austin, J. L. (1962). *How to do things with words*. Cambridge: Harvard University Press.
- [Azevedo, 2016] Azevedo, F. F. (2016). *Dicionário Analógico da Língua Portuguesa*. Lexikon.
- [Bard et al., 2019] Bard, N.; Foerster, J. N.; Chandar, S.; Burch, N.; Lanctot, M.; Song, H. F.; Parisotto, E.; Dumoulin, V.; Moitra, S.; Hughes, E.; Dunning, I.; Mourad, S.; Larochelle, H.; Bellemare, M. G. & Bowling, M. (2019). The hanabi challenge: A new frontier for AI research. *CoRR*, abs/1902.00506.
- [Bender et al., 2021] Bender, E. M.; Gebru, T.; McMillan-Major, A. & Shmitchell, S. (2021). On the dangers of stochastic parrots: Can language models be too big? . FAccT '21, p. 610–623, New York, NY, USA. Association for Computing Machinery.
- [Benveniste, 1970] Benveniste, E. ([2006] 1970). *O aparelho formal da enunciação*. Pontes.

- [Bowden, 2003] Bowden, E.M., . J.-B. M. (2003). Normative data for 144 compound remote associate problems.
- [Brown et al., 2020] Brown, T. B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; Agarwal, S.; Herbert-Voss, A.; Krueger, G.; Henighan, T.; Child, R.; Ramesh, A.; Ziegler, D. M.; Wu, J.; Winter, C.; Hesse, C.; Chen, M.; Sigler, E.; Litwin, M.; Gray, S.; Chess, B.; Clark, J.; Berner, C.; McCandlish, S.; Radford, A.; Sutskever, I. & Amodei, D. (2020). Language models are few-shot learners.
- [Cabr e, 1999] Cabr e, M. T. (1999). *Terminology: Theory, methods, and applications*. Amsterdam: J. Benjamins Pub. Co.
- [Cann et al., 2006] Cann, R.; Kempson, R. & Marten, L. (2006). The dynamics of language: An introduction. syntax and semantics. volume 35. *Academic Press*.
- [Carruthers & Smith, 1996] Carruthers, P. & Smith, P. K. (1996). *Theories of Theories of Mind*. Cambridge University Press.
- [Cruse, 1986] Cruse, D. A. (1986). *Lexical Semantics*. Cambridge University Press.
- [Cruse, 2014] Cruse, D. A. (2014). Prototype theory and lexical semantics. pp. 382–402. Routledge.
- [Davidson, 2003] Davidson, J. E. (2003). *Insights about Insightful Problem Solving*. Cambridge University Press.
- [De Deyne et al., 2016] De Deyne, S.; Verheyen, S. & Storms, G. (2016). *Structure and Organization of the Mental Lexicon: A Network Approach Derived from Syntactic Dependency Relations and Word Associations*, volume 99, pp. 47–79.
- [Devin & Alami, 2016] Devin, S. & Alami, R. (2016). An implemented theory of mind to improve human-robot shared plans execution. In *2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pp. 319–326.

- [Devlin et al., 2019] Devlin, J.; Chang, M.-W.; Lee, K. & Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding.
- [Deyne et al., 2013] Deyne, S. D.; Navarro, D. J. & Storms, G. (2013). Better explanations of lexical and semantic cognition using networks derived from continued rather than single-word associations. *Behavior Research Methods*.
- [Divjak & Arppe, 2013] Divjak, D. & Arppe, A. (2013). Extracting prototypes from exemplars what can corpus data tell us about concept representation? *Cognitive Linguistics*, 24.
- [Ernaux, 2021] Ernaux, A. (2021). *Os Anos*. Fósforo, São Paulo.
- [Filmore, 2006] Filmore, C. (2006).
- [Firth, 1957] Firth, J. R. (1957). A synopsis of linguistic theory 1930-55. 1952-59:1–32.
- [Firth, 1968] Firth, J. R. (1968). Review of selected papers of j. r. firth, 1952-59, ed. by frank l. palmer. 47.
- [Frank & Goodman, 2012] Frank, M. C. & Goodman, N. D. (2012). Predicting pragmatic reasoning in language games. *Science*, 336(6084):998--998. ISSN 0036-8075.
- [Friedman & Förster, 2005] Friedman, R. & Förster, J. (2005). Effects of motivational cues on perceptual asymmetry: Implications for creativity and analytical problem solving. *Journal of personality and social psychology*, 88:263–75.
- [Ghosh et al., 2010] Ghosh, S.; Basu, A.; Kumaran, S. S. & Khushu, S. (2010). Functional mapping of language networks in the normal brain using a word-association task. *The Indian journal of radiology imaging*.

- [Goldman et al., 2006] Goldman, A.; Goldman, B. & Press, O. U. (2006). *Simulating Minds: The Philosophy, Psychology, and Neuroscience of Mindreading*. Philosophy of Mind Series. Oxford University Press, USA. ISBN 9780195138924.
- [Grice, 1989] Grice, P. (1989). *Studies in the way of words*. Harvard University Press.
- [Harris, 1954] Harris, Z. S. (1954). Distributional structure. *WORD*, 10(2-3):146–162.
- [Jaramillo et al., 2020] Jaramillo, C.; Charity, M.; Canaan, R. & Togelius, J. (2020). Word autobots: Using transformers for word association in the game codenames. *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, 16:231–237.
- [Kim et al., 2019] Kim, A.; Ruzmaykin, M.; Truong, A. & Summerville, A. (2019). Cooperation and codenames: Understanding natural language processing via codenames. In *AIIDE*.
- [Kounios & Beeman, 2014] Kounios, J. & Beeman, M. (2014). The cognitive neuroscience of insight. *Annual Review of Psychology*, 65(1):71–93.
- [Koyyalagunta et al., 2021] Koyyalagunta, D.; Sun, A.; Draelos, R. & Rudin, C. (2021). Playing codenames with language graphs and word embeddings. *J. Artif. Intell. Res.*, 71:319–346.
- [Lafourcade, 2007] Lafourcade, M. (2007). Making people play for Lexical Acquisition with the JeuxDeMots prototype. In *SNLP'07: 7th International Symposium on Natural Language Processing*, p. 7, Pattaya, Chonburi, Thailand.
- [Levelt, 1983] Levelt, W. J. (1983). Monitoring and self-repair in speech. *Cognition*, 14(1):41–104. ISSN 0010-0277.

- [McCabe et al., 2001] McCabe, K.; Houser, D.; Ryan, L.; Smith, V. & Trouard, T. (2001). A functional imaging study of cooperation in two-person reciprocal exchange. *Proceedings of the National Academy of Sciences*, 98(20):11832--11835.
- [Mikolov et al., 2013] Mikolov, T.; Chen, K.; Corrado, G. & Dean, J. (2013). Efficient estimation of word representations in vector space. In *International Conference on Learning Representations, ICLR*.
- [Mitchell, 2006] Mitchell, T. M. (2006). The discipline of machine learning.
- [Mollin, 2009] Mollin, S. (2009). Combining corpus linguistic and psychological data on word co-occurrences: Corpus collocates versus word associations. 5(2):175--200.
- [Penn & Povinelli, 2007] Penn, D. C. & Povinelli, D. J. (2007). On the lack of evidence that non-human animals possess anything remotely resembling a 'theory of mind'.
- [Pennington et al., 2014] Pennington, J.; Socher, R. & Manning, C. D. (2014). Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532--1543.
- [Peters, 2005] Peters, C. (2005). Foundations of an agent theory of mind model for conversation initiation in virtual environments.
- [Scott-Phillips, 2010] Scott-Phillips, T. (2010). Animal communication: insights from linguistic pragmatics. *Animal Behaviour*, 79(1).
- [Scott-Phillips, 2015] Scott-Phillips, T. C. (2015). *Speaking our minds: Why human communication is different, and how language evolved to make it special*. Palgrave Macmillan.
- [Shen et al., 2018] Shen, J. H.; Hofer, M.; Felbo, B. & Levy, R. (2018). Comparing models of associative meaning: An empirical investigation of reference in simple language games.

- [Silver et al., 2016] Silver, D.; Huang, A.; Maddison, C. J.; Guez, A.; Sifre, L.; van den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; Dieleman, S.; Grewe, D.; Nham, J.; Kalchbrenner, N.; Sutskever, I.; Lillicrap, T.; Leach, M.; Kavukcuoglu, K.; Graepel, T. & Hassabis, D. (2016). Mastering the game of go with deep neural networks and tree search. *Nature*, 529:484--503.
- [Smith & Medin, 1981] Smith, E. & Medin, D. (1981). *Categories and concepts*. Harvard University Press.
- [Somers, 2020] Somers, J. (2020). Automatically finding codenames clues with glove vectors. <https://gist.github.com/jsomers/1bb5e197dec221714df250e72265a301>.
- [Sperber & Wilson, 1995] Sperber, D. & Wilson, D. (1995). Relevance: Communication and cognition. *Blackwell, Oxford.*, 2nd edn.
- [Sternberg & Davidson, 1995] Sternberg, R. J. & Davidson, J. E. (1995). *The nature of insight*. Cambridge, MA: MIT Press.
- [Subramaniam et al., 2009] Subramaniam, K.; Kounios, J.; Parrish, T. B. & Jung-Beeman, M. (2009). A brain mechanism for facilitation of insight by positive affect. *Journal of Cognitive Neuroscience*, 21(3):415–432.
- [Tissier et al., 2017] Tissier, J.; Gravier, C. & Habrard, A. (2017). Dict2vec : Learning word embeddings using lexical dictionaries. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 254–263, Copenhagen, Denmark. Association for Computational Linguistics.
- [Topolinski & Reber, 2010] Topolinski, S. & Reber, R. (2010). Gaining insight into the "aha" experience. *Current directions in Psychological Science*, 19:402–405.

[Wang et al., 2021] Wang, Q.; Saha, K.; Gregori, E.; Joyner, D. & Goel, A. (2021).

Towards mutual theory of mind in human-ai interaction: How language reflects what students perceive about a virtual teaching assistant. pp. 1–14.

[Zadeh, 1965] Zadeh, L. A. (1965). Fuzzy sets. *information and control*, vol. 8 (1965), pp. 338–353.

[Zunjani & Olteteanu, 2019] Zunjani, F. H. & Olteteanu, A.-M. (2019). Towards reframing codenames for computational modelling and creativity support using associative creativity principles. In *Proceedings of the 2019 on Creativity and Cognition*, CC '19, p. 407–413, New York, NY, USA. Association for Computing Machinery.



# Apêndice A

## Apêndice

### A.1 Exemplos de Jogos entre humanos

Tabela A.1. Jogos entre dois jogadores humanos

Palavras alvo	Dica	Acertos
madeira, dente, bacon	mastigar	3
cone, brinde, balão	aniversário	2
ficha, argentina	passaporte	3
abelha, porta, madeira	Jataí	2
corvo, destino, espada	morte	3
santa, julgamento, etiqueta	comportamento	2
elefante, gôndola, Itália	Pomarola	3
segundo, estetoscópio	emergência	2
portal, hino, avental	França	2
vikings, branco, doca	nórdico	3
oásis, labirinto, sono	confuso	3
sono, marionete, caveira	pesadelo	3
vírus, paciente, pulso	corona	3
sapo, irmão, chapéu	príncipe	3
regra, carreira, artigo	ABNT	2
oásis, névoa, espírito	etéreo	2
testemunha, tinta, branco	parede	2
testemunha, gangster, espírito	assassinato	3
vampiro, capa, desenho	Drácula	3
carruagem, desenho, sinal	Cinderela	3
caranguejo, confusão, torta	aperto	2
cego, sonho, terremoto	escuro	2
amendoim, televisão, sonho	sextou	2
amendoim, jarro, ordem	guardado	2
jarro, amendoim, múmia	pote	3
afoito, pimenta, tornado	agitação	2

cone, cruzado, labirinto	trânsito	3
brinde, tarde, guitarra	aniversário	3
raspar, guitarra, pimenta	arranha	3
roda-gigante, balão, ficha	parque	3
sonho, pentágono, grade	apanhador	3
grade, pentágono, ferreiro	cerca	2
ferreiro, janela, esfinge	construção	2
esfinge, pentágono, lanterna	mistério	3
golfe, televisão, bicicleta	entretenimento	3
gosto, cara, feijão	sentido	3
gota, páscoa, feriado	chocolate	3
ervilha, vapor, gota	cozido	2
tenda, pesca, cobra	acampamento	2
lágrima, chá, peixe	água	2
estábulo, pelotão, prensa	polícia	2
tigre, lobisomem, pesca	predatório	2
pluma, toque, pêssego	suave	3
nave, elástico, golfe	lançamento	3
relógio, boneca, natal	presente	3
papagaio, mosquito, seda	bicho	3
sono, tarde, namorada	domingo	3
labirinto, névoa, cruzado	embaraçado	1
exército, armadura, tambor	guerra	3
banana, caramelo, brinde	bala	3
névoa, guitarra, raspar	show	2
couro, cabra, flecha	caça	2
Bigbang, sinal, Éden	começo	2
sinal, ternura, suor	paquera	3
caramelo, afoito, tornado	cachorro	2
açúcar, balão, roda-gigante	criança	3
chifre, cabra, café	fazenda	3
beliche, secador, pote	casa	3
doca, branco, espada	marinha	3
tabela, ficha, série	bingo	2
crystal, desenho, carruagem	Cinderela	3
chifre, aço, capacete	resistente	3
líquido, receita, amazônia	água	2
filme, chave, mágico	cinema	3
pamonha, samba, tamanduá	brasileira	2
língua, batata, filme	queima	3
Amazônia, Maracanã, Berlim	turístico	3
cangaço, cabelo, sapato	couro	2
ovo, manteiga, leite	vegetariano	3
controle, marcha, cangaço	militar	2
Islândia, costa, trovão	mar	2
ouriço, elfo, alienígena	criatura	2
loja, rapadura, açúcar	mercearia	3
pulso, escama, saci	corpo	2

relógio, parada, múmia	tempo	3
barco, bandeirante, Amazônia	colonização	3
hambúrguer, guaraná, pamonha	quitutes	3
esfinge, cérebro, cruzado	enigma	2
brigadeiro, rapadura, futebol	brasil	2
estádio, skate, medalha	olimpíada	3
feijoada, salsa, família	almoço	2
cemitério, pirâmide, toupeira	catacumba	2
samba, cachaça, órgão	música	3
rosto, saturno, volante	redondo	3
exame, manicure, revolução	vermelho	2
linha, ponte, saudade	ligar	2
maracanã, massa, classe	futebol	2
cavalo, volta, futebol	corrida	2
Egito, Amazônia, mergulhador	rio	2
Amazônia, curupira, verde	mata	3
bicicleta, chocolate, sol	diversão	3
giz, russo, bandeirante	branco	3
capivara, jarro, lenhador	lama	2
bater, helicóptero, nervo	acidente	2
guaraná, lata, limão	refrigerante	3
caminho, cego, faixa	calçada	3
suor, pizza, futebol	mancha	2
pote, biscoito, fazenda	nata	3
testemunha, bebida, prova	lei	3
pé, futebol, faixa	capitão	3
cerca, guitarra, tubo	metal	2
cozinheiro, feijoada, afoito	cru	2
raiz, hino, tijolo	nação	2
festa, hotel, imperador	luxo	2
negativo, túnel, segredo	escuro	3
âncora, capacete, casa	segurança	3