UNIVERSIDADE FEDERAL DO AMAZONAS
INSTITUTO DE COMPUTAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

IDENTIFICAÇÃO E DESAMBIGUAÇÃO DE MENÇÕES A PRODUTOS EM CONTEÚDO GERADO POR USUÁRIOS - UM ESTUDO DE CASO NO DOMÍNIO DE JOGOS

Diego de Azevedo Barros

Julho de 2016

Manaus - AM

UNIVERSIDADE FEDERAL DO AMAZONAS
INSTITUTO DE COMPUTAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

IDENTIFICAÇÃO E DESAMBIGUAÇÃO DE MENÇÕES A PRODUTOS EM CONTEÚDO GERADO POR USUÁRIOS - UM ESTUDO DE CASO NO DOMÍNIO DE JOGOS

Diego de Azevedo Barros

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Informática, Instituto de Computação - IComp, da Universidade Federal do Amazonas, como parte dos requisitos necessários à obtenção do título de Mestre em Informática.

Orientador: Altigran Soares da Silva

Julho de 2016

Manaus - AM

Ficha Catalográfica

Ficha catalográfica elaborada automaticamente de acordo com os dados fornecidos pelo(a) autor(a).

Diego de Azevedo, Barros
D559i Identificação e desambiguação de menções a produtos em conteúdo gerado por usuários : um estudo de caso no domínio de jogos / Barros Diego de Azevedo. 2016
81 f.: il. color; 31 cm.

Orientador: Altigran Soares da Silva
Dissertação (Mestrado em Informática) - Universidade Federal do Amazonas.

1. Ferramenta GameSpotter. 2. Domínio de Jogo. 3. Regras de Desambiguação. 4. Método de Desambiguação. I. Silva, Altigran Soares da II. Universidade Federal do Amazonas III. Título

IDENTIFICAÇÃO E DESAMBIGUAÇÃO DE MENÇÕES A PRODUTOS EM CONTEÚDO GERADO POR USUÁRIOS - UM ESTUDO DE CASO NO DOMÍNIO DE JOGOS

Diego de Azevedo Barros

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO PROGRAMA DE PÓS-GRADUAÇÃO DO INSTITUTO DE COMPUTAÇÃO DA UNIVERSIDADE FEDERAL DO AMAZONAS COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM INFORMÁTICA.

Aprovado por:

Prof. Altigran Soares da Silva, Icomp/UFAM, Doutor

Prof. Viviane Pereira Moreira, INF/UFRGS, Doutora

Prof. João Marcos Bastos Cavalcanti, Icomp/UFAM, Doutor

Prof. David Braga Fernandes de Oliveira, Icomp/UFAM, Doutor

JULHO DE 2016

MANAUS, AM – BRASIL

Aos meus pais, Gilberto (in memoriam) e Cirene, razão de tudo em minha vida; aos meus irmãos, Gildázio e Cimone, pela amizade, incentivo e companheirismo; e aos meus queridos sobrinhos, Felipe e Ayla, por tornarem os meus dias mais alegres.

Agradecimentos

A Deus, por me permitir realizar meus sonhos e objetivos, por estar comigo todos os dias de minha vida, amparar-me nos momentos de dificuldade dando-me forças para não desistir e por eu ter a ciência de que sem Ele nada disso seria possível.

Aos meus pais Gilberto (in memoriam) e Cirene, pelo amor, carinho, cuidado, apoio e incentivos que sempre me deram; a quem sou eternamente grato. Mesmo nos momentos difíceis, nunca deixaram de acreditar em mim.

Aos meus queridos irmãos Gildázio e Cimone, pelo amor, carinho, companheirismo, incentivos de todos os dias e, na ausência de nosso pai, fizeram-me sentir o quanto sou amado.

Aos meus cunhados Isaque e Angélica, dois novos irmãos que Deus me deu.

Aos professores Altigran Soares e João Marcos pelos ensinamentos, dedicação e paciência com que sempre me orientaram.

À Dheniffer Souza, que desde o começo do mestrado sempre me apoiou nos estudos para que eu não perdesse o foco.

Aos meus amigos Leonardo Nascimento, Henry Vieira e Tiago de Melo, por estarem comigo desde o começo do mestrado e por me ajudarem na realização deste trabalho. Assim como, aos demais amigos e familiares que direta ou indiretamente contribuíram para a conclusão de mais essa etapa da minha vida.

A todos vocês, muito obrigado.

"Como é feliz o homem que acha a sabedoria, o homem que obtém entendimento, pois a sabedoria é mais proveitosa do que a prata e rende mais do que o ouro. É mais preciosa do que rubis; nada do que você possa desejar se compara a ela." (Provérbios 3:13-15)

Resumo

Um problema bastante relevante para a análise de comentários postados por usuários em redes sociais é a identificação das entidades que são o alvo destes comentários. No entanto, identificar corretamente as entidades mencionadas em textos produzidos pelos usuários é uma tarefa desafiadora, visto que uma mesma entidade pode ser mencionada de várias maneiras diferentes, dependendo do usuário e de como a menção está sendo feita. Além disso, esses comentários são caracterizados por texto com baixa qualidade de escrita, erros ortográficos, gramaticais, etc. Neste trabalho, apresentamos um estudo de caso sobre o problema de identificação e desambiguação de menções a entidades em conteúdo gerado por usuários, voltado para o domínio de jogos. A escolha deste domínio deve-se à importância econômica e cultural deste tipo de conteúdo e também ao fato de a maioria dos trabalhos na literatura relacionada recente abordar este problema no contexto de produtos eletrônicos (televisores, smartphones, etc.). Como estratégia para a realização deste estudo de caso, desenvolvemos uma ferramenta chamada *GameSpotter*, que utiliza métodos de reconhecimento de entidades nomeadas (*named entity recognition - NER*) e de desambiguação de entidades nomeadas (*named entity disambiguation - NED*) para identificar e desambiguar as menções a jogos nos comentários postados em um fórum real da Web. Para tanto, desenvolvemos dois métodos alternativos NER e um método de NED voltados ao domínio de jogos. Nossos resultados experimentais mostraram que nossos métodos de NER e NED são efetivos, tendo alcançado em média uma precisão de 0,93 e 0,83 em relação ao reconhecimento e desambiguação de menções a jogos, respectivamente.

Abstract

A very important issue for the analysis of comments posted by users in social networks is the identification of the entities that are the target of these comments. However, correctly identifying the entities mentioned in texts produced by users is a challenging task, since the same entity can be mentioned in several different ways, depending on the user and on how the mention is being made. In addition, these comments are characterized by text with low-quality writing, misspellings, grammatical errors, etc. In this work, we present a case study on the problem of identification and disambiguation of mentions to entities in user-generated content, focused on the domain of games. The choice of this domain is due to the economic and cultural importance of this type of content and also because most of the work in recent literature related to this problems focuses on the context of electronics (televisions, smartphones, etc.). As a strategy for carrying out this case study, we have developed a tool called GameSpotter, which uses methods of named entity recognition - NER and named entity disambiguation - NED to identify and disambiguate mentions to games in comments posted on a real Web forum. Therefore, we have developed two alternative NER methods and one NED method focused on the domain of games. Our experimental results showed that our NER and NED methods are effective, achieving an average precision of 0.93 and 0.83 in the recognition and disambiguation mentions of games, respectively.

Sumário

Lista de Figuras

Lista de Tabelas

Capítulo 1

Introdução

Ao longo dos últimos anos observou-se o crescimento acelerado das mídias sociais ao ponto de se tornarem parte do cotidiano das pessoas. Através delas, os usuários trocam informações que eles mesmos geram, utilizando mecanismos de comunicação altamente acessíveis e escaláveis [Kaplan & Haenlein, 2010]. Exemplos de mídias sociais são: redes sociais, blogs, microblogs, enciclopédias colaborativas, sites de compartilhamento de conteúdo multimídia, fóruns, etc. Tais mídias sociais são caracterizadas por conteúdo diversificado, produzido por amadores, onde o conteúdo gerado pode ser compartilhado, discutido, comentado, transformado, citado, etc [Lee & Pang, 2008].

As mídias sociais são definidas como um grupo de aplicações para a internet construídas com base nos fundamentos ideológicos e tecnológicos da Web 2.0, e que permitem a criação e troca de conteúdo gerado pelo usuário [Kaplan & Haenlein, 2010]. Elas podem ser agrupadas conforme o foco em que foram idealizadas, como por exemplo as mídias sociais destinadas ao compartilhamento multimídia (Youtube), fotos (Instagran), músicas (Last.fm), voltadas à comunicação (Blogs), Microblogs (Twitter), Redes Sociais (Facebook, Fóruns) e as Colaborativas (Wikipedia) [Kietzmann et al., 2011].

Atualmente, as mídias sociais têm grande influência nas decisões de consumo dos usuários. Neste contexto, cada vez mais usuários difundem e confiam em opiniões publicadas por outros usuários em tais mídias sociais sobre produtos e serviços [Kaplan & Haenlein, 2010]. Essas opiniões são veiculadas de diferentes formas, como por exemplo, através de notícias, mensagens em redes sociais, postagens e respostas em fóruns, etc. Segundo o Wall Street Jornal[1], os usuários confiam mais em informações

[1] http://www.wsj.com/articles/SB123144483005365353 - acesso em: 04.11.2015

sobre produtos e serviços obtidas em mídia social do que as fornecidas por vendedores. De acordo com trabalho realizado por [Dang et al., 2010], os consumidores que fazem compras online acreditam quase quatro vezes mais em *reviews* online de estranhos do que a sugestão de um amigo.

Um problema bastante relevante para a análise de comentários postados por usuários em mídias sociais é a identificação das entidades que são o alvo desses comentários, ou seja, identificar sobre que entidades do mundo real os comentários se referem. Esse problema tem sido amplamente estudado na literatura recente e, para sua solução, técnicas de Reconhecimento de Entidades Nomeadas (*Named Entity Recognition - NER*) têm sido aplicadas [Vieira & da Silva, 2015, Zhang & Liu, 2011, Wu et al., 2012, Yao & Sun, 2014].

A identificação correta dessas entidades mencionadas no texto são a primeira etapa para outras tarefas que dependem que a entidade alvo desses comentários tenha sido identificada, como por exemplo: desambiguação de entidades nomeadas, análise de sentimentos, extração de aspectos, etc [Yao & Sun, 2015, Hu & Liu, 2004]. Todas essas tarefas podem ser combinadas e utilizadas em sites de *e-commerce* com o intuído de enriquecer essas entidades com conteúdo extraído de mídia social.

Neste sentido, atualmente está em desenvolvimento, no Grupo de Pesquisa em Bancos de Dados e Recuperação de Informação da UFAM, um método chamado *ProdSpot* para identificação de menções a produtos eletrônicos, tais como televisores e smartphones, feitas em comentários de fóruns na Web [Vieira, 2016]. Os fóruns Web foram escolhidos como alvo deste trabalho, pois, dentre os vários tipos de mídias sociais existentes, eles são fontes de opiniões importantes e diversificadas. Ao contrário de outras fontes como por exemplo, os *reviews* de produtos, os fóruns contêm textos com maior liberdade de expressão, pois não focam em produtos específicos, em uma mesma postagem, vários produtos são geralmente mencionados. Além disso, os textos postados em fóruns não sofrem limitações de tamanho como no Twitter [Liu, 2012].

1.1 Objetivos

Nessa dissertação de mestrado, nosso principal objetivo é estudar os problemas de identificação e desambiguação de menções a produtos em um domínio diferente do de pro-

dutos eletrônicos, domínio este que tem sido abordado não somente em [Vieira, 2016], mas também em vários outros trabalhos na literatura recente [Vieira & da Silva, 2015, Yao & Sun, 2014, Wu et al., 2012]. Procuramos com isso, contribuir com a extensão da aplicabilidade dos métodos citados anteriormente e, em particular, com o método proposto por [Vieira, 2016].

Assim, para esta dissertação, escolhemos como alvo o domínio de jogos eletrônicos. Os jogos são atualmente bastante populares e possuem uma grande importância no mercado de entretenimento mundial. Segundo o Yahoo[2], a indústria de jogos cresceu muito nos últimos anos e tornou-se uma das maiores indústrias no ramo de entretenimento, sendo hoje maior do que as indústrias de cinema e música. De acordo com uma notícia publicada na Forbes[3], é previsto que as receitas obtidas com o mercado de jogos deve crescer nos próximos anos, ultrapassando o volume de 67 bilhões de dólares alcançado em 2012, para 82 bilhões de dólares em 2017.

Como estratégia para a realização deste estudo de caso, decidimos por desenvolver uma ferramenta para identificação e desambiguação de menções a jogos em comentários de um fórum real da Web chamado *Gamespot*[4]. Este fórum foi escolhido neste trabalho por ser bastante popular e possuir uma grande variedade de tópicos de discussões sobre diversos tipos de jogos. Nossa ferramenta, chamada *GameSpotter*, é importante em nosso estudo de caso, pois, através dela pudemos demonstrar de forma prática uma aplicação real de técnicas de reconhecimento e desambiguação de entidade nomeadas voltadas ao domínio de jogos para lidar com um grande volume de conteúdo gerado por usuários em um fórum. Como uma de suas funcionalidades, a nossa ferramenta disponibiliza ao usuários uma interface web em que eles podem pesquisar por menções a jogos de seu interesse, e obter os comentários relacionados a sua busca, sem ter a necessidade de verificar manualmente os comentários postados no fórum. Tal tarefa seria custosa e sujeita a erros em um contexto de um fórum com milhões de comentários.

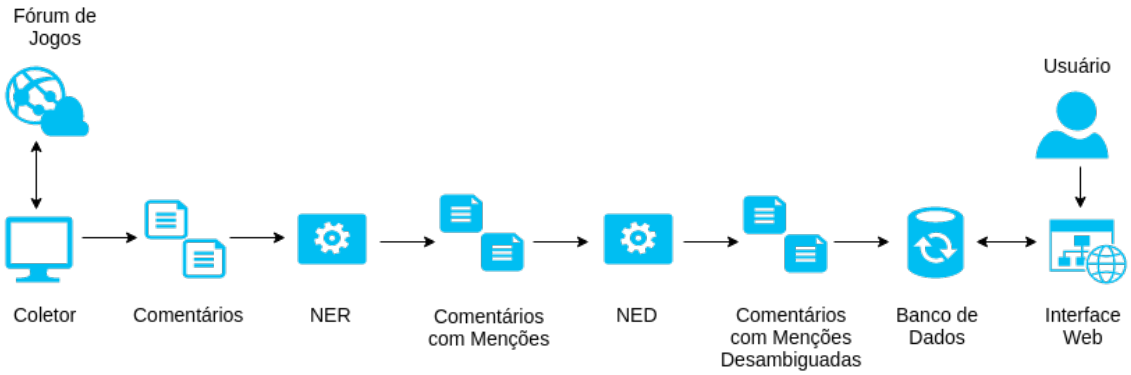

Figura 1.1: Resumo da arquitetura geral da Ferramenta GameSpotter.

1.2 A Ferramenta GameSpotter

A Figura 1.1 apresenta de forma resumida a arquitetura geral da ferramenta *GameSpotter*. Como ilustrado nessa figura, a ferramenta coleta diariamente um conjunto de comentários postados no fórum *Gamespot*. Esses comentários são passados para um método de reconhecimento de entidades nomeadas (NER) que identifica as menções a nomes de jogos mencionados por usuários nesses comentários. Após essa etapa, um método de desambiguação de entidades nomeadas (NED) é usado com o propósito de associar essas menções a nomes canônicos de jogos existentes no mundo real. Em seguida, esses comentários com menções desambiguadas pelo método de NED são inseridos em um banco de dados que é constantemente atualizado. Para que esses dados possam ser acessados, uma interface Web é utilizada pelos usuários a fim de que eles posam pesquisar por jogos de seu interesse.

Os principais componentes do GameSpotter são os métodos de NER e NED. Para a tarefa de NER, a ferramenta pode utilizar dois métodos alternativos. O primeiro, método chamado CRF-Games, é baseado no modelo (*Conditional Random Fields - CRF*), que é considerado o estado da arte em tarefas de NER [Lafferty et al., 2001, Sarawagi, 2008]. O segundo, é uma versão do ProdSpot [Vieira, 2016] que adaptamos especificamente para o domínio de jogos. Esta adaptação chamamos de ProdSpot-Games. A principal diferença entre o CRF-Games e o ProdSpot-Games é que enquanto o primeiro é supervisionado, necessitando de exemplos rotulados pelo usuário, o segundo é auto-supervisionado, o que

[2]http://finance.yahoo.com/blogs/daily-ticker/how-the-video-game-industry-became-bigger-than-movies-and-music-171225174.html - acesso em: 08.12.2016

[3]http://www.forbes.com/sites/johngaudiosi/2012/07/18/new-reports-forecasts-global-video-game-industry-will-reach-82-billion-by-2017/ - acesso em: 08.12.2016

[4]http://www.gamespot.com/

o torna mais prático para aplicação em nossa ferramenta.

No que diz respeito à desambiguação das menções a jogos identificadas pelos métodos de NER, adaptamos o método baseado em regras proposto por [Yao & Sun, 2015] para o domínio de jogos, uma vez que o seu domínio de interesse são as menções a smartphone feitas em fóruns. Ressaltamos que, apesar da ideia geral de nosso método de NED ser bastante similar a da proposta por [Yao & Sun, 2015], nosso conjunto de regras utilizadas para este propósito são bastante diferentes, visto que, as regras usadas no domínio de smartphones não se aplicam diretamente ao domínio de jogos. Em nosso método de desambiguação, fazemos uso de métricas de similaridade para casamento de nomes [Cohen et al., 2003, Navarro, 2001], para tentar associar de forma correta as menções que tenham erros ortográficos e outros tipos de ruídos encontrados em fóruns.

1.3 Desafios

A adaptação dos métodos existentes para lidar com menções a jogos em postagens de um fórum da Web é por si só uma tarefa desafiadora. Estas postagem são de natureza não-estruturada, caracterizada por textos com baixa qualidade de escrita, erros ortográficos, gramaticais, de pontuação e estilo de escrita telegráfico [Liu, 2012]. Além disso, muitas vezes os usuários citam várias entidades diferentes no mesmo comentário e interagem uns com os outros em discussões, o que torna mais trabalhosa a identificação da entidade alvo. Usuários de fóruns no domínio de jogos costumam referenciar um jogo nos comentários por meio de diversas formas de superfície (*surface forms*) ou expressões de entidade (*entity expressions*) que são formas alternativas utilizadas para fazer menções a uma mesma entidade [Feldman, 2013, Liu, 2007]. Uma dessas formas de superfície bastante comum são os acrônimos, que dependendo do contexto em que eles são empregados podem se tornar bastante ambíguos. Exemplos de várias formas de superfície que fazem referência a nomes de jogos são apresentadas na Tabela 1.1, onde são exemplificadas amostras reais, destacadas em negrito, usadas em um fórum de jogos.

Nas três primeiras sentenças da Tabela 1.1 são utilizadas como formas de superfície os acrônimos. No entanto, identificá-los como sendo uma menção válida a nomes de jogos não é uma tarefa trivial, visto que nem sempre os acrônimos fazem referência a jogos. Na quarta e quinta sentenças, são utilizados algarismos arábico e romano para referenciar um

Nome do Jogo	Forma de Superfície	Sentença
Battlefield 4	BF4	Ofcourse I just got done playing **BF4** and obviously there is no comparison.
Call of Duty	COD	But the coloring in **COD** is very bland and grainy.
Max Payne 3	MP3	I can understand how some couldn't get into **MP3** but it blew me away.
Diablo III	3	My top 3 are: Diablo 2, **3** - Mass Effect Trilogy - Skyrim
Metal Gear Solid 4: Guns of the Patriots	IV	The game was quite lengthy and had more game than MGS2 and **IV**, the graphics where quite good.
Call of Duty: Black Ops	Black Ops	At least **Black Ops** was a bit more fair, though some bits are downright impossible for most players.
The Legend of Zelda	Zelda	Like the first **Zelda**, the camera was fixed like this.

Tabela 1.1: Exemplos de formas de superfície usadas para referenciar jogos.

determinado jogo. Esses tipos de forma de superfície geralmente são empregados quando se quer referenciar um jogo que já foi mencionado antes no comentário ou fazer menção a um jogo da mesma franquia do jogo já citado. Nas duas últimas sentenças, por sua vez, são empregadas partes do nome canônico do jogo para referenciar um jogo específico.

Além do uso de formas de superfície distintas, como os fóruns têm a característica de serem informais, muitos jogos são escritos de forma incorreta, o que dificulta a tarefa de desambiguação, ou seja, associar a menção encontrada a um nome canônico de um jogo. Isso é exemplificado na Tabela 1.2, onde **call od duty 2** e **mass eff 3** são formas incorretas de mencionar os jogos "Call of Duty 2" e "Mass Effect 3", respectivamente.

You could also a bom same as in **call od duty 2** MP.
maybe **mass eff 3** ending sucked because it ran out of HL games to copy.

Tabela 1.2: Exemplos de menções a jogos escritas de forma incorreta.

Outra dificuldade encontrada no domínio de jogos é que um mesmo acrônimo pode fazer referência a mais de um jogo. Exemplos de sentenças reais extraídas do fórum *Gamespot* são mostradas na Tabela 1.3, que exemplifica acrônimos comumente usados para fazer referência a mais de um jogo.

COD - (Call of Duty, Chaos on Deponia, Castle of Deceit, Castle of Dragon)
SF - (Street Fighter, Sango Fighter, Shadow Fighter, Shaq Fu, Shining Force)

Tabela 1.3: Exemplos de acrônimos usados para fazer referência a mais de um jogo.

1.4 Contribuições

Nossas contribuições neste trabalho são: (*i*) um estudo de caso sobre os problemas de identificação e desambiguação de menções de produtos em textos gerados por usuários em um domínio diferente dos que são geralmente abordados na literatura atual; (*ii*) adaptação do modelo CRF por meio de um método supervisionado de NER para o domínio de jogos; (*iii*) adaptação de um método de identificação de menções a produto, o ProdSpot [Vieira, 2016], para o domínio de jogos; (*iv*) adaptação de um método de NED baseado em regras, proposto por [Yao & Sun, 2014], para o domínio de jogos; e (*v*) desenvolvimento da ferramenta GameSpotter como estratégia para auxiliar nesse estudo de caso.

1.5 Organização da Dissertação

Esta dissertação está estruturada como segue. No Capítulo 2, são apresentados conceitos importantes para o entendimento desse trabalho, bem como os trabalhos relacionados. No Capítulo 3 apresentamos o método supervisionado *CRF-Games* para identificação de menções a nomes de jogos. No Capítulo 4 descrevemos o método auto-supervisionado *ProdSpot-Games* para reconhecimento de menções a nomes de jogos. No Capítulo 5, descrevemos um método para desambiguação de entidades nomeadas baseado em regras adaptado ao cenário de jogos proposto em [Yao & Sun, 2014]. Os experimentos realizados e os resultados obtidos estão descritos no Capítulo 6. No Capítulo 7 descrevemos a ferramenta *GameSpotter* desenvolvida neste trabalho e, por fim, no Capítulo 8, discutimos as conclusões e o direcionamento para os trabalhos futuros.

Capítulo 2

Revisão de Literatura e Trabalhos Relacionados

Este capítulo introduz conceitos básicos necessários para melhor compreensão do trabalho proposto, bem como uma revisão da literatura relacionada a sua área de abrangência. Os conceitos apresentados incluem, reconhecimento de entidades nomeadas (NER), técnica utilizada para identificar entidades alvo em textos, desambiguação de entidades nomeadas (NED), técnica utilizada para mapear uma menção de entidade a uma entidade do mundo real presente em uma base de conhecimento e Métricas de Distância de Palavras utilizadas em nosso método de NED.

2.1 Identificação de Menções a Produtos

O problema de identificação de menções a produto consiste em identificar de forma automática trechos em um texto escrito por um usuário que se referem a um produto de determinada categoria, por exemplo, equipamentos eletrônicos, carros e, em nosso caso específico, jogos eletrônicos. Este problema é uma instância do problema mais geral de reconhecimento de entidades nomeadas (NER), onde, ao invés de localizar qualquer tipo de entidade (pessoas, lugares, organizações, etc.), busca-se especificamente por menções a jogos.

Reconhecimento de entidades nomeadas é uma subárea de extração de informação (EI) que se preocupa em identificar e classificar as entidades mencionadas em textos estruturados ou não. Dado um texto como entrada para uma tarefa de NER, ele então é

segmentado em sentenças e estas sentenças são divididas em várias palavras. Cada palavra que compõe esta sentença é classificada como sendo ou não uma entidade de interesse. Entende-se por entidade qualquer sujeito concreto ou abstrato que possui um nome próprio. Podem ser exemplos de entidades: nomes de pessoas, lugares, organizações, cidades, países, produtos, etc.

Para [Ling & Weld, 2012], NER é um tipo de tarefa relacionada à extração de informação que visa identificar regiões do texto (menções) correspondentes a entidades e categorizá-las numa lista pré-definida de tipos de entidades de interesse.

A ideia por trás de tarefas relacionadas a NER é que as entidades nomeadas, geralmente são fundamentais para a compreensão do que o texto se refere. Essas tarefas podem ser consideradas como uma etapa básica para mineração de dados [Jiang, 2012].

No contexto de mineração de opiniões, métodos de NER tentam extrair o alvo da opinião em textos como um passo inicial para outras tarefas que necessitam que a entidade alvo tenha sido identificada. Exemplos de tarefas que exigem o reconhecimento do alvo da opinião são: extração de aspectos, sumarização de opiniões, análise de sentimento, análise de polaridade, dentre outros.

Para contextualizar uma tarefa de NER em relação ao domínio de jogos, considere a sentença apresentada na Figura 2.1.

> There are games that build upon the formula and create something deeper and better to play (see **Silent Hill 2**, **Batman: Arkhan City**, **Assassin's Creed 2** and **Uncharted 2** as examples for games that made a series better).

Figura 2.1: Reconhecimento de menções a jogos em uma dada sentença.

Nossa entidade alvo a ser reconhecida são as menções a nomes de jogos que estão realçadas em negrito na Figura 2.1, para uma dada sentença de entrada.

Para identificar entidades nomeadas em textos, geralmente utiliza-se algum método de aprendizagem de máquina. Dentre os vários métodos de aprendizagem de máquina, destacam-se os métodos probabilísticos, como por exemplo o *Hidden Markov Models* (HMM) [Sarawagi, 2001, Sarawagi & Mansuri, 2006] e o *Conditional Random Fields* (CRF) [Lafferty et al., 2001, Sutton & McCallum, 2006, Sarawagi, 2008]. Os padrões assimilados a partir do *corpus* de entrada são utilizados para inferir sobre os novos dados que serão passados. Os algoritmos de aprendizagem de máquina são fundamentalmente dependentes de uma fase inicial de aprendizagem ou treinamento, o qual tentam produ-

zir um modelo matemático capaz de deduzir conhecimento com base nas amostras de dados [Witten et al., 2011]. Dentre as abordagens de aprendizagem de máquina comumente utilizadas para as tarefas de NER, pode-se citar: aprendizagem supervisionada e não supervisionada [Christopher, 2011, Mitchell, 1997, Witten et al., 2011].

Na literatura recente, métodos de NER têm sido utilizados para a tarefa de reconhecimento de menções a produtos. A seguir são apresentados alguns trabalhos relacionados à proposta deste trabalho.

Em [Vieira & da Silva, 2015], é apresentado o método chamado *ModSpot*, que adota uma abordagem de auto-treinamento utilizada para gerar treino um modelo baseado no CRF, a fim de reconhecer menções a números de modelos de produto em fóruns da Web. Neste trabalho, os autores utilizam 4 bases diferentes para 3 tipos de produtos eletroeletrônicos, sendo eles: blu-ray players, televisores e receptores de áudio e vídeo. Como o método é baseado em um arcabouço de auto-treinamento [Witten et al., 2011, Teixeira et al., 2011], seu algoritmo faz uso intensivo de fontes de dados não rotulados com o intuito de treinar um modelo CRF. Um treinamento inicial é realizado utilizando um conjunto de sementes (lista com uma quantidade suficiente de números de modelos de produtos) e uma fonte de dados não rotulada, em que cada semente é expandida em várias formas de superfície, que são formas alternativas de mencionar o mesmo produto. Para cada forma de superfície expandida, elas são anotadas automaticamente nas sentenças de entrada usadas para treinar um CRF inicial. O *ModSpot* utiliza a saída do treinamento inicial realizado para encontrar novos números de modelos de produtos em sentenças não rotuladas. Números de modelos de produto com alta confiança são adicionados ao conjunto inicial de sementes e são novamente expandidas em múltiplas formas de superfície, a qual outra vez são anotadas em sentenças de entrada não rotuladas utilizadas para treinar um novo modelo CRF. Este processo executa até que não sejam mais encontradas novas sementes para serem incluídas no conjunto inicial de sementes. Ao final são comparados os resultados de *Precisão*, *Revocação* e F_1 do método proposto com o modelo CRF supervisionado. Em seus experimentos realizados, o método desenvolvido supera o baseline em 19% na *Revocação* e 12% em F_1.

No trabalho proposto por [Vieira, 2016], atualmente em desenvolvimento em nosso grupo de pesquisa, é apresentado o método chamado ProdSpot para identificar menções a nomes de produtos em comentários de usuários postados em fóruns. Assim

como no *ModSpot* [Vieira & da Silva, 2015], este método utiliza uma abordagem de auto-treinamento, bootstrapping, para treinar um modelo baseado no CRF. Como fonte de dados para treinar esse modelo, são utilizados comentários coletados a partir de um fórum específico para produtos eletroeletrônicos e uma lista de produtos de interesse, como por exemplo: televisores, smartphones, câmeras fotográficas, etc. Cada item dessa lista de produtos, eles são anotados automaticamente nas sentenças dos comentários, que ao fim desse processo, essas sentenças anotadas são utilizadas como treino para geração do modelo baseado no CRF. Vale ressaltar que, ao contrário do *ModSpot*, o modelo é gerado apenas 1 vez e este é utilizado para identificar menções a produtos nos comentários. Outro ponto em destaque é que no ProdSpot a tarefa de NER é multi-palavras, onde se pretende extrair a menção completa aos nomes de produtos e não apenas uma palavra, como é o caso do *ModSpot*, no qual se deseja extrair menções a números de modelos de produtos.

O trabalho desenvolvido por [Wu et al., 2012], conseguiu alcançar o melhor desempenho na extração e ligação de entidades no concurso CPROD1[1] da ICDMW[2] de 2012. O concurso oferece um conjunto de dados que incluem milhares de comentários de usuários extraídos de mídia social, um catálogo com milhões de produtos e um conjunto de menções a nomes de produtos anotadas manualmente nos comentários. A proposta do concurso é determinar o melhor método para identificar menções a produtos nos comentários e ligá-las às suas entidades correspondentes no catálogo de produtos. Para isso, os autores propõem uma abordagem híbrida que combina os resultados obtidos por 3 métodos separadamente para a tarefa de NER. (i) - método para casamento de palavras padrão, o qual são usados menções a produtos utilizadas no treino para procurar essas menções diretamente na coleção de teste, (ii) - método baseado em regras, onde as menções a produtos são reconhecidas através de regras produzidas a partir da examinação dos comentários dos usuários e são específicos para produtos, e (iii) - método baseado em uma versão da implementação do modelo CRF [McCallum, 2002]. Os métodos propostos se concentram em identificar menções a produtos em diferentes aspectos do problema, com o intuito de se obter um grande conjunto de menções a produtos identificadas pelos três métodos. Ao final dessa tarefa, as menções identificadas nas mesmas sentenças dos comentários são eliminadas com o propósito de eliminar redundâncias, ficando apenas 1 das menções identificada na sentença. Para a ligação dessas menções a seus produtos refe-

[1]Consumer PRODucts contest #1

[2]International Conference on Data Mining Workshops

rentes no catálogo, é utilizado um método de ponderação de votos, o qual para todos os produtos contidos no catálogo de produtos, eles são candidatos se e somente se existe uma ocorrência da menção neste produto, e a seleção do melhor casamento menção/produto é definida pelo produto que obteve a maior quantidade de votos pelo método.

No trabalho apresentado em [Putthividhya & Hu, 2011] os autores apresentam um sistema de (NER) para extrair valores de atributos a partir de uma lista de ofertas de produtos. Eles utilizam uma base de dados do site eBay[3] para as categorias de roupas e sapatos masculinos e femininos. Os atributos por eles investigados são: marca, cor, tamanho e vestuário (tipo/estilo). Para essa tarefa, é adotada uma abordagem de aprendizagem de máquina supervisionada. Os classificadores utilizados foram: SVM *(Support Vector Machines)*, Modelos de Máxima Entropia *(Maximum Entropy Models)*, HMM (Hidden Markov Models) e o CRF *(Conditional Random Fields)*. Posteriormente, foram realizados experimentos utilizando a abordagem de bootstrapping para gerar treinamento automático e usar os métodos propostos a fim de identificar novas marcas nas listas de ofertas de produtos. Isso resultou em uma precisão de 90.33% com o modelo de Máxima Entropia.

2.2 Desambiguação de Menções a Produtos

O problema de desambiguação de menções a produtos consiste em associar de forma automática uma certa menção a um produto no texto escrito por um usuário ao seu respectivo nome canônico contido em uma base de conhecimento. Particularmente, em nosso caso, desejamos desambiguar menções a jogos eletrônicos em conteúdo gerado por usuários em mídia social. Este problema é uma instância do problema mais geral de Desambiguação de Entidades Nomeadas (NED), onde, ao contrário de desambiguar qualquer tipo de menção a entidades, como por exemplo pessoas, organizações, instituições, produtos diversos, etc, busca-se especificamente por menções a jogos.

Em um contexto mais amplo, desambiguação é a tarefa de eliminar ambiguidades em menções a entidades nomeadas em texto e ligá-las as suas entidades correspondentes do mundo real presentes em bases de conhecimento como por exemplo, DBpedia [Auer et al., 2007], Freebase [Bollacker et al., 2008], etc. Dado um texto como entrada em que se tenha a entidade alvo reconhecida, a ideia é mapear esta entidade a uma

[3]http://www.ebay.com

entidade do mundo real presente em uma base de conhecimento. Para realizar essa tarefa, geralmente são utilizados métodos baseados em aprendizagem de máquina, métodos baseados em regras, métricas de distância de edição, etc.

O problema fundamental da desambiguação de entidades nomeadas é medir a similaridade entre as ocorrências de nomes [Han & Zhao, 2009]. Para calcular a similaridade entre duas palavras, existem na literatura várias métricas. Dentre elas, uma bastante utilizada é a *Distância de Levenshtein* [Navarro et al., 2001, Navarro, 2001]. Essa medida baseia-se no número mínimo de transformações (inserção, exclusão e substituição) necessárias para transformar uma palavra "a" em uma palavra "b" [Levin, 2010]. Essa medida de similaridade entre palavras é bastante utilizada como um recurso para a tarefa de NED.

Técnicas de NED são bastante utilizadas em processamento de linguagem natural, com o intuito de eliminar ambiguidades a uma referência de entidade conhecida no texto. No entanto, a tarefa de NED é considerada difícil de ser resolvida quanto as mais difíceis tarefas em inteligência artificial [Navigli, 2009]. Nos últimos anos, várias técnicas de aprendizagem de máquina e baseadas em regras, têm sido propostas com o objetivo de tentar solucionar o problema. Em métodos supervisionados, as palavras distintas do *corpus* contendo o texto são treinadas a partir de um método a fim de gerar um modelo genérico que possa ser aplicado as novas instâncias de entrada. Métodos baseados em regras geralmente são específicos para cada domínio, onde é assumida a existência de uma lista de nomes canônicos e um *corpus* de entrada. As entidades alvo são identificadas através de algum método de NER, supervisionado ou não, e posteriormente elas são mapeadas para os seus respectivos nomes canônicos de entidade contido na lista.

O problema de NED em conteúdo gerado por usuários em fóruns para o domínio de jogos sofre algumas complicações adicionais devido os usuários citarem em seus comentários o mesmo jogo de diferentes formas. A Tabela2.1 mostra exemplos de variações de menções para o jogo "The Legend of Zelda: Ocarina of Time" encontradas em fóruns.

Forma de Superfície	Sentença
TLoZ OoT	Deus Ex Human Revolution: PS3 and Wii U, **TLoZ OoT** & MM: N64, GC and 3DS, TLoZ WW: GC and Wii U, TLoZ TP: GC and Wii, TLoZ: ALttP: physical and digital, TLoZ ALBW: physical and digital.
Ocarina	I think the best Zelda game was Twilight Princess followed by **Ocarina**.
Ocarina of Time	**Ocarina of Time** is one of these. Once you practically make it to Ganon's Castle, you really have no reason to continue.
The Legend of Zelda: Ocarina of Time	Back when I was a kid, **The Legend of Zelda: Ocarina of Time** was my fucking drug.

Tabela 2.1: Diferentes formas de superfície para referenciar o mesmo jogo.

Além do uso de formas de superfície distintas para referenciar o mesmo jogo, outro desafio encontrado no domínio de jogos é que uma mesma forma de superfície é utilizada para referenciar mais de um jogo. A Tabela 2.2 mostra exemplos do uso da mesma forma de superfície para referenciar os jogos: "Final Fight 3" e "Final Fantasy III", respectivamente.

Nome do Jogo	Sentenças
Final Fight 3	I would say that Final Fight 3 and Streets of Rage 3 are just ok... Edge to Final Fight 3, Just to break it down, this is my personal order for all 6 games... SoR2, FF2, FF, SoR, **FF3**, SoR3, Final Fight wins the series over all, but Streets of Rage 2 is the best game of all 6 in my opinion so either way it is almost too close to call...
Final Fantasy III	Sorry, to be more specific the SNES Versions called Final Fantasy 2 and Final Fantasy 3. I'm aware of the North American changes to the titles of some RPGs. Thanks for pointing that out. I enjoyed **FF3** (actually FF6) more than 2 (actually FF4). They are both awesome games, but FF6 had more customization, a better and more original story and a better cast of characters.

Tabela 2.2: Utilização da mesma forma de superfície para referenciar mais de um jogo.

Para tentar resolver esses problemas, adaptamos o método baseado em regras proposto por [Yao & Sun, 2015], para o domínio de jogos, visto que esse método obteve bons re-

sultados de precisão e revocação para o domínio de smartphones.

Para exemplificar a tarefa de desambiguação de entidades nomeadas no domínio de jogos, considere a sentença apresentada na Figura 2.2.

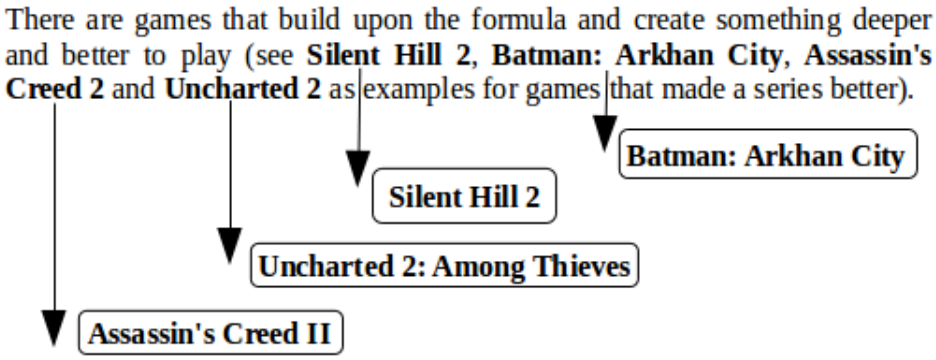

Figura 2.2: Desambiguação de menções a jogos em uma dada sentença.

Observamos nessa figura que as menções a nomes de jogos estão destacadas em negrito e o mapeamento correto dessas menções a seus respectivos nomes canônicos de jogos estão representados com uma seta para cada nome canônico do jogo em destaque.

Na literatura recente, métodos de NED têm sido aplicados para a tarefa de desambiguação de menções a produtos. A seguir são apresentados alguns trabalhos relacionados à proposta deste estudo.

No trabalho apresentado em [Yao & Sun, 2015], os autores propõem um método intitulado GREN, que identifica e desambigua menções a nomes de smartphones em fóruns. A diferença da sua proposta e outros trabalhos publicados na área de NED é que em vez de reconhecer diretamente menções a smartphones, é realizada uma etapa anterior chamada de "geração de nomes candidatos" que podem ou não ser utilizados como menções válidas a serem desambiguadas. O método GREN recebe como entrada dois conjuntos distintos: (1) a coleção formal/canônica de nomes de smartphones e (2) a coleção de comentários coletados a partir de um fórum destinado a discussão sobre smartphones. A partir deste momento são executados os três componentes principais do método GREN, sendo eles: (i) gerador de nomes candidatos, no qual é realizado o processo de captura nos comentários de possíveis menções a nomes de smartphones; (ii) reconhecedor de nomes baseados no modelo CRF, que nesta etapa são passadas sentenças com menções a nomes de smartphones para serem identificadas por um classificador previamente treinamento baseado no CRF e (iii) normalização de nomes de smartphones baseados em regras, que consiste em desambiguar as menções identificadas na etapa anterior e associá-las aos seus

respectivos nomes canônicos contidos numa lista. Para isso, é construída uma lista com variações de nomes *(surface forms)* para cada nome formal f da lista de nomes canônicos, onde são consideradas duas etapas. Primeiro, se todos os caracteres de um nome candidato c estão contidos na marca e/ou no modelo do nome formal do smartphone e estão dispostos na mesma sequência, então c é adicionado a lista de variações de nomes para f denotado por L^f, exemplos: Samsung Galaxy SIII, sgs 3, etc. Segundo, se o nome candidato c contem o número do modelo do nome formal do smartphone f então o nome candidato também é adicionado a L^f, exemplos: i9300, samsung i9300 galaxy s iii, etc. A partir dessas duas etapas, cada nome formal de smartphone contém uma lista de formas de superfície utilizadas para mencionar este nome canônico de smartphone. No entanto, nem todas as formas de superfície mencionadas no fórum atendem a essas duas etapas, dessa maneira, a fim de aumentar a lista de formas de superfície distintas usadas para mencionar um mesmo smartphone, é realizada a tokenização de todas as formas de superfície com mais de uma palavra para cada nome formal f. Dessa forma, se um nome candidato c casa com alguma palavra que foi tokenizada, então c é adicionado a lista do nome formal L^f do smartphone em que ele casou, exemplos: s3 lte, s3 pebble blue, etc. No entanto, pode ocorrer que duas formas de superfície podem estar em mais de uma lista de nomes canônicos de smartphones L^f. Para tentar solucionar esse problema, é utilizada a co-ocorrência de maior frequência dessas formas de superfície com que elas ocorrem no fórum, e a forma de superfície que foi citada com maior frequência para um determinado nome formal f, é a que continua na lista formal para L^f e as demais são removidas das outras listas L^f de nomes formais de smartphones, com isso se elimina ambiguidades.

No trabalho proposto por [Hoffart et al., 2011], os autores apresentam um novo método de desambiguação de entidades nomeadas chamada AIDA, um sistema NED robusto que faz uso de um grafo ponderado entre a menção e as entidades candidatas para encontrar o melhor mapeamento entre a menção a ser desambiguada e a entidade presente em uma base de conhecimento. Este método utiliza abordagens já usadas em trabalhos anteriores, e combina três medidas amplamente empregadas em problemas de NER, sendo elas: probabilidade a priori de uma menção a uma entidade sendo citada no texto, a similaridade entre os contextos de uma menção e uma entidade candidata, assim como a coerência entre as entidades candidatas para todas as menções juntas. Posteriormente para cada ligação menção-entidade candidata é calculado um subgrafo denso com todas

as possibilidades possíveis entre a menção citada no texto e a entidade do mundo real. O algoritmo realiza algumas iterações até que a melhor configuração menção-entidade seja encontrada. Como base de conhecimento de relações de entidades, este trabalho utiliza a base YAGO [Suchanek et al., 2007].

Em [Li et al., 2013], é apresentado um método para desambiguação de entidades nomeadas em texto de linguagem natural. Dado um texto e a menção que se deseja desambiguar, o método extrai o contexto e as evidências internas em que esta menção ocorre no documento como sendo características úteis para mapear uma menção a uma entidade do mundo real presente em uma base de conhecimento. Assim como em outros trabalhos propostos na literatura, em relação a tarefa de NED, sua base de conhecimento foi a *Wikipedia*. Entretanto existem dificuldades com tal abordagem, pois geralmente as bases de conhecimento são incompletas levando assim a um fraco desempenho na desambiguação quando o contexto das menções não são bem conhecidas. Um outro fator relevante, é que nem sempre as evidências internas, como por exemplo o texto de âncora na menção, são suficientes para desambiguar de forma correta essas menções a entidades do mundo real. Para tentar resolver este problema os autores desenvolveram um algoritmo incremental que extrai novas evidências em documentos externos, como por exemplo a homepage da entidade a ser desambiguada, páginas da *Wikipedia*, que têm hiperlinks para sua página da *Wikipedia* e a DBLP para pesquisadores, a fim de aumentar o conhecimento sobre uma determinada menção que se quer desambiguar. Os Experimentos mostraram que seu método aumenta a precisão de 43% para 86% em comparação ao baseline deste trabalho.

2.3 Métricas de Similaridade para Casamento de Nomes

Como será detalhado no Capítulo 5, nosso método adaptado de NED é composto por um conjunto de regras que são executas sequencialmente a fim de associar de forma automática um nome canônico de jogo a uma menção. Em algumas destas regras utilizamos métricas de similaridade de casamento de nomes para fazer esta associação. Nesta seção explicaremos algumas métricas de casamento de nomes utilizadas em nosso método de NED, dentre elas, Distância de Levenstein, Jaro, Jaro-Winkler, TFIDF e SoftTFIDF.

2.3.1 Funções de Similaridade Baseadas em Edição

Métricas baseadas em distância de edição podem ser definidas como uma função $d(s,t)$ que mapeia um par de palavras s e t para um valor real r, onde quanto menor o valor de r indica maior similaridade entre s e t [Cohen et al., 2003].

Uma métrica que utiliza operações de edição de caracteres é a Distância de Levenstein [Navarro et al., 2001, Navarro, 2001]. Assim, essa distância entre um par de palavras s e t é computada pelo custo unitário para todas as operações a serem realizadas a fim de transformar uma palavra s em t. Basicamente, as operações de edição são inserção, remoção e substituição de caracteres, onde cada operação tem um custo atribuído a ela [Cohen et al., 2003, Navarro, 2001].

Uma outra métrica semelhante a Distância de Levenstein, mas que não se baseia em distância de edição é a Jaro [Jaro, 1989, Winkler, 1999]. Esta métrica é baseada no número de caracteres comuns entre duas palavras, e também na ordem com que elas ocorrem. Assim, seja as palavras $s = a_1 \dots a_k$ e $t = b_1 \dots b_L$, um caractere a_i de s é considerado comum com t se existir um caractere $b_j = a_i$ em t tal que $i - H \leq j \leq i + H$, ou seja, o valor de i não pode ser diferente do valor de j mais do que um valor H, onde $H = \frac{min(|s|,|t|)}{2}$, e $|s|$ e $|t|$ são os tamanhos das palavras s e t, respectivamente.

Seja $s' = a'_1 \dots a'_k$ os caracteres em s que são comuns com os de t (na mesma ordem que aparecem em s) e seja $t' = b'_1 \dots b'_L$ análogo a s'. Uma transposição ocorrerá quando um caractere na posição i em s' for diferente de um caractere na mesma posição i em t', ou seja, $a'_i \neq b'_i$. Seja também $T_{s',t'}$ a metade do número de transposições para s' e t' [Cohen et al., 2003, Bilenko et al., 2003]. Dessa maneira, a métrica de similaridade Jaro para as palavras s e t é computada como:

$$Jaro(s,t) = \frac{1}{3}.(\frac{|s'|}{|s|} + \frac{|t'|}{|t|} + \frac{|s'| - T_{s',t'}}{|s'|}) \tag{2.1}$$

Uma variante da métrica Jaro é a Jaro-Winkler [Winkler, 1999]. Esta métrica utiliza um tamanho P do prefixo comum de caracteres mais longo entre as palavras s em t. Assim, $P' = max(P,4)$, ou seja, é o prefixo mais longo de caracteres comuns entre s e t não superior a 4 caracteres. Dessa maneira, a métrica de Jaro-Winkler é computada de

acordo com a Equação 2.2.

$$Jaro-Winkler(s,t) = Jaro(s,t) + \frac{P'}{10}.(1 - Jaro(s,t)) \tag{2.2}$$

As métricas de Jaro e Jaro-Winkler são destinadas principalmente a casamentos de palavras curtas como por exemplo: nome e sobrenome de pessoas [Cohen et al., 2003].

2.3.2 Funções de Similaridade Baseadas em Palavras

Em muitos casos a ordem das palavras não é importante, como por exemplo as strings s = "Ray Mooney" e t = "Mooney, Ray" são duas formas distintas de mencionar uma mesma entidade presente no mundo real. Nesses casos, pode-se transformar as strings s e t em multiconjuntos de tokens (onde cada token é uma palavra) e considerar métricas de similaridade sobre estes multiconjuntos[Cohen et al., 2003, Bilenko et al., 2003].

Uma métrica baseada em palavras é a TFIDF[4], o qual é bastante utilizada em recuperação de informação. Ela é definida de acordo com a Equação 2.3

$$TFIDF(S,T) = \sum_{w \in S \cap T} V(w,S).V(w,T) \tag{2.3}$$

Onde $TF_{w,S}$ é a frequência da palavra w em S, e IDF_w é o inverso da fração de nomes no corpus que contêm w.

$$V'(w,S) = log(TF_{w,S} + 1).log(IDF_w) \tag{2.4}$$

e $V(w,S)$ é definido como sendo:

$$V(w,S) = \frac{V'(w,S)}{\sqrt{\sum_{w'} V'(w,S)^2}} \tag{2.5}$$

2.3.3 Funções de Similaridade Híbridas

As métricas descritas acima são baseadas em distância de edição, número de caracteres comuns entre duas palavras s e t e multiconjuntos de tokens. A métrica SoftTFIDF é função de similaridade Híbrida, que combina métricas de casamento de tokens, como é o caso do TFIDF, com métricas baseadas em palavras, como é o caso do Jaro-Winkler.

[4]Em nosso trabalho, adotamos a mesma terminologia empregada em [Cohen et al., 2003], que considera TFIDF como sendo uma função de similaridade.

Neste sentido, o SoftTFIDF é descrito como sendo uma versão "soft" do TFIDF em que tokens similares são considerados em $S \cap T$. Assim, seja sim' uma função de similaridade secundária e $CLOSE(\theta, S, T)$, um conjunto de palavras $w \in S$ tal que existe algum $v \in T$ onde $dist'(w,v) > \theta$, e para $w \in CLOSE(\theta, S, T)$, seja $D(w,T) = max_{v \in T} dist(w,v)$ [Cohen et al., 2003]. Portanto, a função de similaridade SoftTFIDF é definida como:

$$SoftTFIDF(S,T) = \sum_{w \in CLOSE(\theta,S,T)} V(w,S).V(w,T).D(w,T) \tag{2.6}$$

Em nossos experimentos, assim como em [Cohen et al., 2003], utilizamos a função Jaro-Winkler como sendo a função de similaridade secundária sim' e $\theta = 0.9$.

Capítulo 3

Identificação de Menções a Jogos – Um Método Supervisionado

Neste capítulo descrevemos um método supervisionado que desenvolvemos para o nosso estudo de caso. Este método é baseado no conhecido modelo *Conditional Random Fields - CRF*, que adaptamos para reconhecer menções a nomes de jogos em comentários de usuários de um fórum Web. Iniciamos o capítulo apresentando conceitos básicos sobre o CRF e em seguida descrevemos como este modelo foi utilizado em nosso método. Este é um dos métodos de identificação de menções a nomes de jogos utilizado na ferramenta GameSpotter. O outro método, neste caso auto-supervisionado, será descrito no Capítulo 4.

3.1 Conditional Random Fields

Muitas das tarefas relacionadas à predição e classificação de textos, como por exemplo, análise sintática de texto em processamento de linguagem natural, envolvem um grande número de variáveis que dependem umas das outras, bem como de outras variáveis observadas. Métodos de predição estruturados são essencialmente uma combinação de classificação e modelagem gráfica. Eles combinam a capacidade de modelos gráficos em modelar de forma compacta dados multivariados com a capacidade dos métodos de classificação em realizar a predição usando grandes conjuntos de características de entrada. Como solução proposta na literatura para esses tipos de tarefas, são utilizadas abordagens discriminativas como por exemplo, o CRF [Sutton & McCallum, 2011].

O CRF é um arcabouço para construção de modelos probabilísticos utilizado para segmentar e rotular sequências de dados. Ele é um modelo matemático probabilístico baseado em uma abordagem condicional e pode ser modelado na forma de um grafo não dirigido que define uma única distribuição logarítmica linear sobre uma sequência de rótulos, dada uma sequência de observação. Assim, as influências das diferentes características em estados distintos podem ser tratadas independentemente umas das outras [Lafferty et al., 2001]. Um CRF é uma distribuição condicional $P(Y/X)$ com um modelo gráfico associado. A variável X é um vetor de variáveis aleatórias de entrada e Y é um vetor de variáveis aleatórias de saída. Portanto $P(Y|X)$ é a probabilidade de obter Y dado como entrada o vetor X [Sutton & McCallum, 2011].

$$P(Y/X) = \frac{P(X/Y)P(Y)}{P(X)} \tag{3.1}$$

Devido ao CRF ser originalmente um modelo discriminativo, ele modela a distribuição de probabilidade condicional *P(Y|X)*. Modelos baseados no CRF evitam o conhecido *label bias problem*, que normalmente é observado em Modelos Markovianos Condicionais, tal como o Modelo de Markov de Entropia Máxima (Maximum entropy Markov model) [Sutton & McCallum, 2006].

Para modelarmos a probabilidade condicional $P(Y/X)$, utilizada para rotular Y dado como entrada um vetor X, fazemos uso de um caso especial do CRF que é o Campos Aleatórios Condicionais de Cadeias Lineares (*Linear-Chain Conditional Random Field - LCCRF*). De maneira mais formal, essa distribuição condicional segue a modelagem apresentada por [Sutton & McCallum, 2011].

Definição: Sejam Y e X vetores aleatórios de tamanho T, seja $\lambda = \{\lambda_k\}$ ε $\Re^k$, um parâmetro do vetor, e $F = \{f_k(y, y', x_t)\}_{k=1}^{K}$, um conjunto de funções características e t uma posição do vetor em X. Portanto um *linear-chain Conditional Random Field* é uma distribuição $P(Y/X)$ que possui a seguinte fórmula:

$$P(Y/X) = \frac{1}{Z(x)} exp\{\sum_{t=1}^{T}\sum_{k=1}^{K} \lambda_k f_k(y_t, y_{t-1}, x_t)\}, \tag{3.2}$$

Note que podemos mover o somatório em T para fora da função exponencial na equa-

ção 3.2 e transformá-la em:

$$P(Y/X) = \frac{1}{Z(x)} \prod_{t=1}^{T} exp\{\sum_{k=1}^{K} \lambda_k f_k(y_t, y_{t-1}, x_t)\}, \tag{3.3}$$

onde $Z(x)$ é uma função de normalização definida por:

$$Z(x) = \sum_{y} \prod_{t=1}^{T} exp\{\sum_{k=1}^{K} \lambda_k f_k(y_t, y_{t-1}, x_t)\} \tag{3.4}$$

A definição exemplificada acima é derivada do *Hidden Markov Models (HMM)* [Bikel et al., 1997], no qual *P(Y|X)* é calculada a partir da distribuição *P(X,Y)* dada pelo *HMM*. Uma característica importante deste modelo, tal como todos os derivados do *Conditional Random Field*, é que não há necessidade de se conhecer a distribuição *P(X)*, que pode ser uma distribuição demasiadamente complexa [Santos, 2012].

Dessa forma, pela definição dada pela equação 3.3 é necessário definir o vetor de funções características $f = (f_1, f_2, ..., f_k)$ e o vetor de pesos $\wedge = (\lambda_1, \lambda_2, \ldots, \lambda_k)$. Como o vetor $\wedge$ aplica pesos sobre o vetor de funções características, então algumas funções possuem maior relevância do que outras na contabilização final da probabilidade estimada [Santos, 2012].

As funções características possuem o formato $f_k(y_t, y_{t-1}, x_t)$, onde y_t é a rotulação da palavra na posição t do vetor de entrada X, y_{t-1} é a rotulação da palavra na posição $t-1$ do vetor de entrada X e x_t é o vetor formado pelas palavras de X que são relevantes para o cálculo do rótulo y_t [Sutton & McCallum, 2006, Sutton & McCallum, 2011] .

Métodos baseados no modelo CRF têm sido aplicados com sucesso em problemas de predição de saída estruturada, assim como são bastante utilizados para tarefas de Reconhecimento de Entidades Nomeadas (NER) em Processamento de Linguagem Natural (NLP), que consiste na identificação de menções a entidades presentes em texto na forma livre de dados textuais [Sarawagi, 2008].

3.2 Adaptação do Modelo CRF para o Domínio de Jogos

A geração de um modelo para reconhecimento de entidades nomeadas baseada no CRF envolve uma série de etapas, entre elas: definição das características (*features*) utilizadas no modelo, definição da notação usada na marcação dos exemplos de treino e teste, de-

finição do tipo da entrada que será recebida pelo modelo e, finalmente, como será feita a validação do modelo gerado. Nesta seção descrevemos cada uma dessas etapas para o caso do modelo de reconhecimento de menções a jogos CRF-Games.

3.2.1 Conjunto de Features Utilizadas

Um aspecto muito importante na tarefa de reconhecimento de entidades a partir do CRF é a definição das características (*features*) utilizadas no modelo, o que corresponde as funções do tipo f_k na Equação 3.3. Essas características são representadas como vetores, que correspondem aos dados de entrada que são aplicados no treinamento do CRF. As características são extraídas para todas as palavras do texto de entrada e são examinadas pelo modelo gerado no momento de predizer os rótulos de uma sequência de palavras Y_t em uma dada sentença X_t de entrada, como sendo de uma classe de interesse ou não.

As funções de características produzem valores binários (1 ou 0) dependendo das características que são analisadas pelo modelo, dado como entrada o vetor X. Por exemplo, podemos definir uma simples função de característica f_1 que fornece o valor 1 se a palavra atual é KILLZONE e seu rótulo de marcação nessa posição é JOGO.

$$f_1(y_t, y_{t-1}, x_t) = \begin{cases} 1 & \text{se } x_t = KILLZONE \text{ e } y_t = JOGO \\ 0 & \text{outros casos} \end{cases}$$

De maneira semelhante, uma outra função de característica f_2 poderia levar em consideração as transições de estados dos rótulos definidos, como por exemplo os rótulos (OUTRO, JOGO). Neste caso, a função retornaria 1 olhando o estado do rótulo anterior da palavra na posição y_{t-1} (OUTROS) para o rótulo da palavra atual y_t (JOGO), e 0 para outros casos.

$$f_2(y_t, y_{t-1}, x_t) = \begin{cases} 1 & \text{se } y_{t-1} = OUTRO \text{ e } y_t = JOGO \\ 0 & \text{outros casos} \end{cases}$$

Assim como essas funções de características que retornam valores binários, outras funções mais complexas também podem ser introduzidas a partir do conjunto de características analisadas nas funções de características $f_k = f_1, f_2, \ldots, f_t$. Por exemplo, poderia ser analisada a combinação das palavras vizinhas em uma janela de contexto de tamanho 3, a partir da palavra observada com os rótulos *part-of-speech* (POS) da palavra atual e

suas vizinhas. Nesse caso, a função de características poderia retornar o valor 1 ou 0 dependendo da melhor configuração de escolha das características [Zhu, 2010].

Desta forma, de acordo com a Equação 3.3, o cálculo das probabilidades dos rótulos Y_t para uma dada sentença de observação X_t é feito com base no produto entre o valor da função de característica f_t por um fator de peso λ_t , que é o peso ou importância dessa característica na posição X_t do vetor de observações.

Característica de Agrupamento

As características acima mencionadas são comumente empregadas em diversas tarefas de NER. Além destas, outra característica muito importante empregada em métodos do estado-da-arte como [Koo et al., 2008] e [Ratinov & Roth, 2009], é obtida a partir de algoritmos de agrupamento de palavras. Mas especificamente, em nosso trabalho utilizamos o algoritmo de agrupamento Brown [Brown et al., 1992].

Este algoritmo recebe como entrada um vocabulário de palavras a serem agrupadas e um *corpus* contendo essas palavras. Inicialmente, todas as palavras do vocabulário são consideradas como sendo de grupos distintos. Em seguida, várias operações de uniões entre pares de grupos de palavras são realizadas a fim de agrupar as palavras que possuem um alto grau de semelhança, considerando sua ocorrência com outras palavras no *corpus*. Ao executar as operações de união de pares de grupos de palavras, obtém-se um agrupamento hierárquico, que pode ser representado como uma árvore, assim como exemplificado na Figura 3.1.

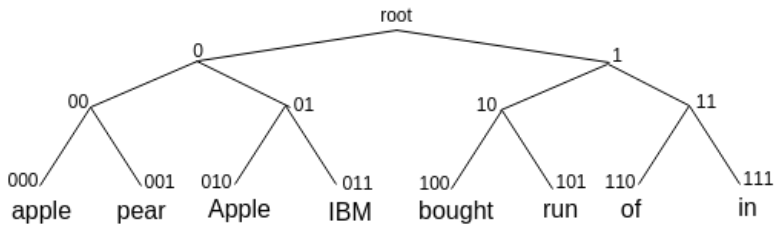

Figura 3.1: Exemplo de agrupamento hierárquico de palavras pelo algoritmo Brown [Ratinov & Roth, 2009].

Nessa figura, observamos que o agrupamento de palavras é feito de forma hierárquica em uma árvore binária, na qual cada palavra pode ser identificada por seu caminho a partir do nó raiz. Por exemplo, os 4 grupos formados a partir da seleção dos nós de profundidade 2, constituído pelas palavras, {apple, pear}, {Apple, IBM}, {bought, run} e {of, in} na

Figura 3.1, também poderiam ser representados de forma compacta como uma cadeia de bits, sendo {00}, {01}, {10} e {11}, respectivamente [Koo et al., 2008].

Caminhos de diferentes profundidades a partir do nó raiz fornecem níveis distintos de abstração das palavras a serem analisadas pelo modelo. Por exemplo, prefixos binários formados por 4 *bits* podem ser utilizados para representar o grupo das palavras a serem analisadas e os prefixos mais longos, como por exemplo 20 *bits*, podem ser usados para representar propriamente as palavras [Koo et al., 2008]. Assim como em [Ratinov & Roth, 2009], em nosso trabalho utilizamos prefixos de diferentes granularidades (4, 6, 10 e 20), como características adicionais para o modelo CRF a fim de representar o caminho completo das palavras a partir do nó raiz até o seu grupo.

Tal como realizado por [Yao & Sun, 2015], em nosso trabalho utilizamos como características para o modelo CRF treinado, os prefixos de cadeia de bits de tamanhos 4, 6, 10 e 20 das palavras em seus grupos de palavras, apenas as que ocorrem em pelo menos 10 vezes no *corpus*. Consideramos este um valor razoável, já que nosso *corpus* é o conjunto de comentários coletados do fórum Gamespot, que possui aproximadamente 200 mil comentários. Dessa maneira elimina-se palavras pouco frequentes que podem atrapalhar a classificação pelo modelo.

Através da observação e análise dos comentários dos usuários do fórum-alvo, definimos um conjunto de características que foram utilizadas pelo modelo CRF treinado tanto para o treino, quanto para o teste. Estas características são descritas a seguir.

1. A palavra atual e seus vizinhos em uma janela de contexto de tamanho 3, x_{i-3}, x_{i-2}, x_{i-1}, x_i, x_{i+1}, x_{i+2}, x_{i+3}.

2. Características da forma da palavra atual e seus vizinhos, em uma janela de contexto de tamanho 3: palavra começando com letra maiúscula; todos os caracteres da palavra são maiúsculos; se possui caractere maiúsculo; se a palavra é numeral; se palavra é uma combinação de caracteres alfanuméricos; e se a palavra tem pontuação.

3. Rótulos *part-of-speech* (POS) da palavra atual e seus vizinhos em uma janela de contexto de tamanho 3.

4. Prefixos de caminho de comprimentos 4, 6, 10, 20 (ou seja, comprimento máximo) da palavra atual pelo agrupamento Brown.

3.2.2 Rotulação de Exemplos de Treinamento e Teste

Para rotular as menções a nomes de jogos nas sentenças dos comentários para treinamento e teste, duas notações alternativas são geralmente usadas na literatura: BILOU e BIO [Ratinov & Roth, 2009]. A primeira notação utiliza os seguintes rótulos: **B***(Begin)* para a primeiro palavra da menção, **I***(Inside)* para um ou mais palavras que fazem parte da menção, exceto a primeira e a última palavra, **L***(Last)* para a última palavra da menção encontrada, **O***(Other)* para qualquer palavra que não faz parte da menção e **U***(Unit)*, para a a menção que possui apenas um palavra. Já a segunda notação utiliza apenas os rótulos **B** e **I**, nesse caso, **(I)**, também é utilizado para a última palavra da menção e **O** para qualquer outra palavra que não faz parte da menção.

Na literatura recente, a notação BILOU tem alcançado melhores resultados experimentais para reconhecimento de entidades nomeadas, tais como pessoas, organizações, localizações, etc, relatados em [Ratinov & Roth, 2009]. No entanto, em experimentos de validação que realizamos no domínio de jogos, após experimentar estas duas notações, optamos por utilizar uma terceira alternativa, na qual dois rótulos são usados: (**P**) para menções a menções a jogos e (**O**) para outros. Esta marcação simples levou aos melhores resultados dentre as três notações consideradas em nosso trabalho.

3.2.3 Preparação das Entradas

Preparação das Sentenças de Treino

Com o propósito de preparar as sentenças usadas para o treinamento do CRF-Games, inicialmente selecionamos aleatoriamente 250 comentários de um total de aproximadamente 200 mil comentários de usuários coletados do fórum Gamespot. Em seguida, executamos o algoritmo de agrupamento Brown sobre a coleção completa de 200 mil comentários, a fim de gerar diferentes grupos de palavras como exemplificado na Figura 3.1. O processo de geração das sentenças de treino em si então pode ser iniciado. Este processo é descrito pelo Algoritmo 1.

Nosso algoritmo recebe como entrada um conjunto de comentários U e produz como saída um conjunto A, que contém as sentenças extraídas dos comentários em U, anotadas com as características que serão utilizadas pelo modelo. Para simplificar o algoritmo, apresentamos apenas as características de POS e agrupamento.

Algoritmo 1 Algoritmo para converter comentários em sentenças anotadas

Input: Um conjunto de comentários U
Output: Um conjunto de sentenças anotadas A

```
1:  A ← ∅                                                    ▷ Conjunto vazio
2:  for each c em U do
3:      S ← segmentar(c)                          ▷ Segmenta os comentários em sentenças
4:      for each s em S do
5:          Seja n o número de palavras em s
6:          ⟨p_1, p_2, ..., p_n⟩ ← segmentar(s)    ▷ Segmenta as sentenças em palavras
7:          for each p em ⟨p_1, p_2, ..., p_n⟩ do
8:              p.pos ← POSTagger(p)
9:              p.prefixo_4 ← brown(p,4)
10:             p.prefixo_6 ← brown(p,6)
11:             p.prefixo_10 ← brown(p,10)
12:             p.prefixo_20 ← brown(p,20)
13:         end for
14:         s ← reconstruirSentença(⟨p_1, p_2, ..., p_n⟩)        ▷ Reconstroi a sentença
15:         A ← A ∪ {s}
16:     end for
17: end for
18: return A
```

No Laço 2–17, cada comentário é processado. Na Linha 3, o comentário atual é segmentado em sentenças, sendo cada sentença processada no Laço 4–16. Na Linha 6, a sentença é divida em palavras. Para cada palavra $p_1, p_2, \ldots, p_n$ pertencente a sentença s, é feita uma anotação através de um algoritmo de *Part-of-Speech* (POS) *tagging* [Gimpel et al., 2011], que determina a classe gramatical da palavra em análise, como por exemplo substantivo, adjetivo, verbo, pronome, etc. Isto é implementado no método *POSTagger* na Linha 8. Na Linha 9, a palavra atual é anotada com o caminho que identifica o grupo de nível 4 no qual a palavra foi colocada pelo algoritmo de Brown. Esse caminho é retornado pelo método *brown*. O mesmo ocorre para os grupos de nível 6, 10 e 20, nas Linhas de 10 a 12, respectivamente. Na Linha 14, a sentença é reconstruída a partir das palavras anotadas e, em seguida, na Linha 15, essa sentença reconstruída é adicionada ao conjunto A de sentenças anotadas. Esse processo é repetido até que todos os comentários tenham sido processados e o conjunto A é retornado pelo algoritmo.

Uma vez que temos todos os comentários segmentados em sentenças no formato de entrada para o CRF, rotulamos manualmente cada palavra das sentenças com as classes de interesse definidas nesse trabalho, sendo "(P)" palavras que fazem menções a nomes de jogos e "(O)" para outros.

Como um exemplo da preparação de sentenças rotuladas utilizadas para treinar um modelo baseado no CRF, considere a sentença extraída de um comentário mostrada na Figura 3.2.

Fallout New Vegas I must have beaten 15 times and I'm currently going through again.

Figura 3.2: Sentença com uma menção a nome de jogo.

Ao executarmos o Algoritmo 1 sobre essa sentença, teremos uma sentença anotada com as *tags* POS e a representação do seu grupo pelos prefixos de tamanhos (4, 6, 10 e 20). Ao final da execução desse algoritmo, rotulamos cada palavra com sua determinada classe. As palavras de interesse nessa sentença são "Fallout New Vegas" as quais são marcadas com o rótulo "(P)", ou seja, são palavras que fazem referência a um nome de jogo específico, e as demais recebem a marcação "(O)". Finalizada essa marcação manual, temos a sentença de saída exemplificada na Figura 3.3

palavra	pos	prefixo_4	prefixo_6	prefixo_{10}	prefixo_{20}	rótulo
Fallout	IN	1011	101101	1011010100	101101010010	P
New	NNP	1011	101111	1011111000	1011111000	P
Vegas	NNP	1010	101010	1010100111	101010011101010	P
I	NNP	1111	111110	1111100	1111100	O
must	MD	1110	111010	1110100001	111010000110	O
have	VB	1110	111000	11100010	11100010	O
beaten	VBN	1110	111011	1110110001	111011000111111	O
15	CD	1011	101100	1011001010	10110010100	O
times	NNS	1011	101100	1011000101	1011000101	O
and	CC	1101	110101	11010100	11010100	O
I	PRP	1111	111110	1111100	1111100	O
'm	VBP	1110	111011	1110111111	11101111111	O
currently	RB	1111	111101	1111011011	1111011011111111	O
going	VBG	1000	100000	10000000	10000000	O
through	IN	1100	110010	1100101110	11001011101	O
again	RB	1101	110111	1101111010	11011110100	O
.	.	1101	110110	1101101	1101101	O

Figura 3.3: Exemplo de preparação das sentenças de treino.

Podemos observar nesta figura cada palavra separadamente, assim como, o conjunto de características em cada palavra, como por exemplo as *tags* POS, e os prefixos de diferentes granularidades utilizados para representar a palavra em seu grupo.

Preparação das Sentenças de Teste

A preparação das sentenças de teste segue o mesmo processo da preparação das sentenças de treino explicado acima. Assim, depois de preparadas de acordo com o procedimento descrito no Algoritmo 1, estas sentenças podem ser fornecidas para o modelo CRF a partir do método CRF-Games, previamente treinado.

Para exemplificar a aplicação do modelo CRF treinado, considere a mesma sentença apresentada na Figura 3.3. Ao ser aplicado o modelo CRF sobre essa sentença, tem-se a sentença classificada como ilustrado na Figura 3.4. Nesta figura apresentamos para cada palavra o rótulo predito pelo modelo, sendo P (Jogo) ou O (outros), e um score de confiança na predição. O valor 0.923930 apresentado no topo da figura corresponde a um *score* de confiança do modelo CRF na rotulação da sentença inteira. Em nosso caso, estes valores de *score* de confiança são gerados pelo Wapiti [Lavergne et al., 2010], que é uma implementação bastante utilizada do modelo CRF.

palavra	pos	prefixo$_4$	prefixo$_6$	prefixo$_{10}$	prefixo$_{20}$	rótulo	rótulo crf	confiança
# 0.923930								
Fallout	IN	1011	101101	1011010100	101101010010	P	P	P/0.999980
New	NNP	1011	101111	1011111000	1011111000	P	P	P/0.973521
Vegas	NNP	1010	101010	1010100111	101010011101010	P	P	P/0.950011
I	NNP	1111	111110	1111100	1111100	O	O	O/0.999980
must	MD	1110	111010	1110100001	111010000110	O	O	O/1.000000
have	VB	1110	111000	11100010	11100010	O	O	O/1.000000
beaten	VBN	1110	111011	1110110001	111011000111111	O	O	O/0.999878
15	CD	1011	101100	1011001010	10110010100	O	O	O/0.999256
times	NNS	1011	101100	1011000101	1011000101	O	O	O/0.999910
and	CC	1101	110101	11010100	11010100	O	O	O/1.000000
I	PRP	1111	111110	1111100	1111100	O	O	O/1.000000
'm	VBP	1110	111011	1110111111	11101111111	O	O	O/1.000000
currently	RB	1111	111101	1111011011	1111011011111111	O	O	O/1.000000
going	VBG	1000	100000	10000000	10000000	O	O	O/0.999996
through	IN	1100	110010	1100101110	11001011101	O	O	O/1.000000
again	RB	1101	110111	1101111010	11011110100	O	O	O/1.000000
.	.	1101	110110	1101101	1101101	O	O	O/1.000000

Figura 3.4: Saída da classificação pelo modelo CRF.

Como pode ser visto na Figura 3.4, o modelo CRF classificou a menção "Fallout New Vegas" como sendo da classe (P), em nosso caso específico sendo um jogo, e todas as outras palavras da sentença como sendo da classe (O) de outros. Nesse exemplo, o modelo CRF acertou em todos os rótulos da sentença em análise.

3.2.4 Validação do Modelo

Para avaliar o modelo CRF gerado a partir dos comentários de treino, utilizamos a técnica de validação cruzada de 10-*folds*, onde separamos a coleção de teste em teste e treino. Em cada *k-fold*, tomamos 10% da base para teste e 90% da base para treinamento. A técnica de validação cruzada de *k-folds* é uma forma amplamente utilizada para dividir uma única amostra em k conjuntos de testes estatisticamente independentes, e assim poder avaliar cada modelo gerado separadamente para obtenção de estimativas mais confiáveis [Jain et al., 2000].

Como métricas de avaliação utilizadas para medir o desempenho de cada modelo CRF gerado a partir da divisão da coleção de teste em *10-folds*, adotamos as métricas de Precisão (*Pr*), revocação (*Rc*) e Medida-F1 (F_1), que são amplamente empregadas na literatura para avaliar o desempenho de classificadores [Vieira & da Silva, 2015, Jakob & Gurevych, 2010, Yao & Sun, 2015]. A estimação final das métricas (*Pr*, *Rc*, F_1) é computada pelo somatório de seus valores individuais em cada *fold* de teste, divido pela quantidade de *folds*, em nosso caso 10. Essas métricas foram computadas conforme os somatórios de *True Positives - (TP)*, *False Positives - (FP)* e *False Negatives - (FN)* conforme as equações abaixo:

$$Pr = \frac{\sum TP}{\sum TP + \sum FP} \tag{3.5}$$

$$Rc = \frac{\sum TP}{\sum TP + \sum FN} \tag{3.6}$$

$$F1 = \frac{2 * \sum TP}{2 * \sum TP + \sum FP + \sum FN} \tag{3.7}$$

3.2.5 Distribuição do CRF Utilizada

Atualmente, existem várias distribuições que implementam o CRF e podem ser utilizadas para as tarefas de NER. Dentre elas podemos citar o Wapiti − distribuição desenvolvida

em C, CRF++ – distribuição desenvolvida em C++, DGM – distribuição desenvolvida em C++, MALLET – distribuição desenvolvida em Java, GRMM – distribuição desenvolvida em Java, etc. Dentre as várias distribuições citadas acima, escolhemos utilizar a distribuição Wapiti[1], devido a sua simplicidade e utilização em outros projetos de pesquisa.

[1] https://wapiti.limsi.fr/

Capítulo 4

Identificação de Menções a Jogos – Um Método Auto-Supervisionado

Neste capítulo, descrevemos um método alternativo ao método CRF-Games apresentado no Capítulo 3 para reconhecimento de menções a nomes de jogos. Este método, chamado *ProdSpot-Games* é uma versão do ProdSpot[1] que está em desenvolvimento por [Vieira, 2016], o qual adaptamos para nosso estudo de caso no domínio de jogos eletrônicos. Ao contrário do CRF-Games, o ProdSpot-Games é auto-supervisionado, que o torna mais prático para aplicação em nossa ferramenta GameSpotter.

4.1 Visão Geral

Como descrito no Capítulo 3, o treinamento de um modelo de reconhecimento de entidades nomeadas baseado no modelo CRF exige um número significativo de instâncias de treinamento rotuladas manualmente. No entanto, obter e rotular tais instâncias pode consumir bastante tempo e tornar-se muito custoso. Em particular, para o propósito deste trabalho, o fato de que novos jogos são lançados com frequência e novas formas de superfície são utilizadas para mencionar tanto jogos novos como os já existentes, pode requerer que o modelo de reconhecimento de entidades tenha que ser atualizado periodicamente.

Para contornar essa dificuldade encontrada em métodos como o CRF-Games, adaptamos o método ProdSpot, para reconhecimento de menções a produtos feitas em comentários em fóruns. Chamamos esta adaptação de ProdSpot-Games. Da mesma forma que

[1] **Prod**uct **Spot**ter - reconhecimento de menções a nomes de produtos eletrônicos.

o ProdSpot, o ProdSpot-Games é um método auto-supervisionado, ou seja, ele não necessita de instancias de treinamento rotuladas manualmente para treinar um modelo CRF, para isso, ele utiliza grandes volumes de comentários não rotulados extraídos de um fórum voltado ao domínio de jogos, assim como um conjunto de sementes, que são nomes canônicos de jogos obtidos de uma base de conhecimento, com o objetivo de gerar instâncias de treino automaticamente rotuladas utilizadas para treinar um modelo baseado no CRF. Para esse fim, nosso método ProdSpot-Games utiliza uma abordagem bootstrapping [Teixeira et al., 2011].

4.2 Geração do Modelo pelo ProdSpot-Games

Para gerar instâncias rotuladas automaticamente, o bootstrapping utiliza comentários de usuários coletados do fórum Gamespot e um conjunto de sementes obtidas de uma lista de jogos disponível na DBpedia [2].

O processo de preparação das instâncias de treino usadas no ProdSpot-Games é feito da mesma maneira como descrito no Algoritmo 1 do CRF-Games. No entanto, ao invés de um humano ter que rotular manualmente as instâncias de treinamento usadas no CRF-Games, o ProdSpot-Games utiliza o bootstrapping para rotular automaticamente essas instâncias. O processo de rotulação automática das instâncias de treino usadas no ProdSpot-Games é descrito no Algoritmo 2.

No Algoritmo 2, são passados como entrada um conjunto de sementes S, um conjunto de sentenças anotadas A, processadas conforme o Algoritmo 1 descrito na Seção 3.2.3, um valor de limiar para sentenças LS e um valor de limiar para a quantidade de palavras que formam uma dada sentença LP. Inicialmente, na Linha 1 é criado um conjunto vazio SR de sentenças rotuladas. Posteriormente, no Laço 2–13 se inicia o processo de buscas e rotulação das sementes no conjunto de sentenças anotadas A. Na Linha 3 são retornadas todas as sentenças anotadas em A em que a semente s ocorre. Na linha 4 é verificado se a quantidade de sentenças retornadas em $|ST|$ é igual ou superior a um limiar LS. A ideia é que só serão consideradas sementes com algum suporte, ou seja, que ocorrem com uma certa frequência, nas sentenças dos comentários dos usuários. Depois de alguns experimentos de prévios de validação foi estabelecido o valor de LS igual a 10. Na Li-

[2] http://web.informatik.uni-mannheim.de/DBpediaAsTables/DBpediaClasses.htm

Algoritmo 2 Algoritmo para treinar um modelo CRF pelo método ProdSpot-Games

Input: Sementes S, sentenças anotadas A, limiar de sentenças LS, limiar de palavras LP
Output: Um modelo CRF treinado Θ

```
1:  SR ← ∅                                                    ▷ Conjunto vazio
2:  for each s ∈ S do
3:      ST ← setenças em A que contém s
4:      if |ST| ≥ LS then
5:          for each st ∈ ST, tal que |st| > LP do
6:              Atribuir o rótulo "O" para todas as palavras de st
7:              for each substring su ∈ st, tal que su = s do
8:                  Atribuir o rótulo "P" para todas as palavras de su
9:              end for
10:             SR ← SR ∪ {st}
11:         end for
12:     end if
13: end for
14: Θ ← treino(SR)
15: return Θ
```

nha 6 do Laço 5–11, cada palavra das sentenças pertencentes a ST é rotulada previamente com o rótulo "O" (outros). Somente são consideradas sentenças que possuem mais que LP palavras. Em nosso algoritmo, adotamos LP igual 5, pois desejamos obter sentenças com uma quantidade razoável de palavras, que possam transmitir alguma informação útil em relação aos jogos citados nessa sentença. Ressaltamos que este limiar pode ser facilmente modificado dependendo do domínio em que nosso algoritmo for aplicado. Caso essa condição seja satisfeita, a sentença em análise é processada no Laço 7–9. Na Linha 8, os rótulos das palavras da sentença anotada que correspondem à semente que está sendo processada são trocadas por "P" (produto). Na Linha 10, as sentenças rotuladas são atribuídas ao conjunto SR de sentenças rotuladas. Esse processo de rotulação automática das palavras nas sentenças é repetido até que todas as sementes tenham sido processadas. Finalmente, na Linha 14 o conjunto de sentenças rotuladas SR são utilizadas para treinar um modelo baseado no CRF e, por fim, na Linha 15 este modelo é retornado.

Este processo de rotulação automática das palavras é realizado com o objetivo de evitar que um humano precise rotular cada palavra manualmente, o que causaria um esforço muito grande para tal tarefa. Além disso, instâncias de treino rotuladas erroneamente poderiam ser inseridas, o que causaria um prejuízo ao modelo treinado.

Uma vez que o modelo CRF foi treinado, ele então pode ser aplicado sobre os outros comentários extraídos do fórum. A Figura 4.1 exemplifica de forma resumida a geração

de um modelo baseado no CRF.

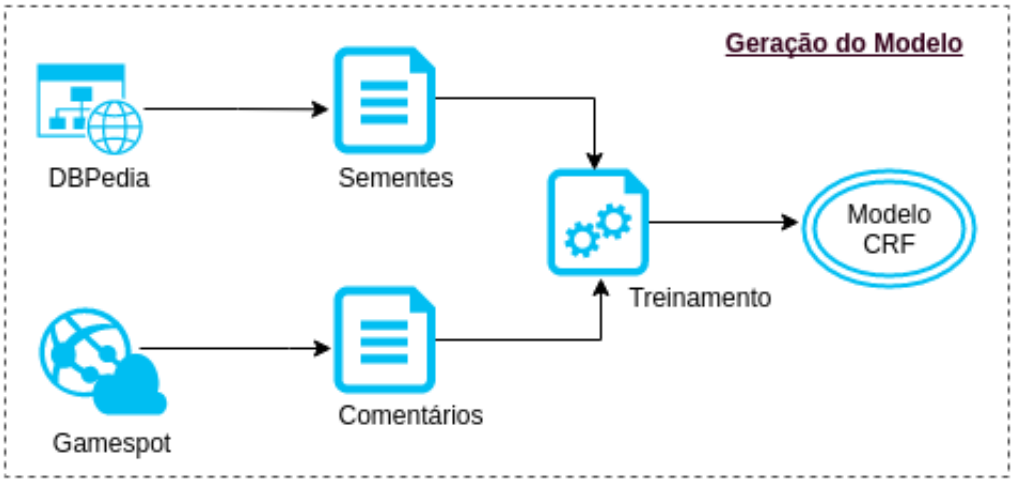

Figura 4.1: Processo automático de geração do modelo CRF.

Na Figura 4.1, a lista de sementes é obtida do site da DBPedia e o conjunto de comentários é obtido do fórum Gamespot. Em seguida, essas sementes (nomes de jogos) são anotadas e rotuladas automaticamente nas sentenças dos comentários extraídos do fórum. Por fim, essas sentenças rotuladas são utilizadas para treinar e gerar um modelo baseado no CRF.

Índice Invertido

Para agilizar o processo de busca dos jogos nas sentenças dos comentários, utilizamos um índice invertido para armazenar os documentos (sentenças) e a posição das palavras contidas nesse documento, Linha 3 do Algoritmo 2 [Baeza-Yates & Ribeiro-Neto, 2013]. Nosso vocabulário é composto pelas palavras distintas da coleção de dados coletada, com cerca de 200 mil comentários. Para armazenar esse índice invertido evitando colocá-lo em memória principal, que neste caso poderia ser limitado pela configuração da máquina utilizada, utilizamos uma biblioteca de código aberto que implementa as funções de um SGBD relacional, chamada *SQLite* [3]. Como um exemplo da criação desse índice invertido, considere a Figura 4.2 a seguir.

Na Figura 4.2, consideramos que possuímos apenas um documento em nossa coleção de documentos, ou seja, um comentário a ser indexado. Nessa figura, a primeira divisão da chave representa um comentário de um usuário que possui 5 palavras distintas. Na

[3]https://www.sqlite.org/

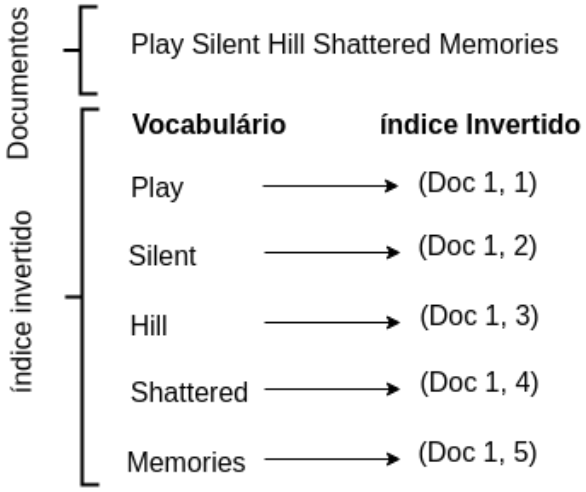

Figura 4.2: Criação de um índice invertido.

segunda chave da mesma figura apresentamos o vocabulário composto pelas palavras diferentes presentes na coleção de documentos e logo em seguida o índice invertido. Por exemplo, a palavra “Silent” ocorre no documento 1 na posição 2, a palavra “Hill” ocorre no documento 1 na posição 3, e assim sucessivamente.

Capítulo 5

Método para Desambiguação de Menções a Jogos

Neste capítulo, apresentamos um método de desambiguação de menções a jogos baseado em regras com o intuito de associar de forma automática uma menção a um nome canônico de jogo contido em uma base de conhecimento. Descrevemos as estratégias utilizadas, bem como todas as etapas envolvidas para essa finalidade. Ressaltamos que esse método de desambiguação de menções a jogos é uma de nossas contribuições deste trabalho.

5.1 Desambiguação de Menções a Jogos

Em um contexto mais amplo, a tarefa de desambiguação de entidades nomeadas em texto de linguagem natural consiste em associar de forma automática menções ambíguas a entidades canônicas como pessoas, lugares, organizações, etc, representadas em uma base de conhecimento como por exemplo DBpedia ou Yago [Hoffart et al., 2011].

No domínio de jogos, seja um conjunto de menções $M = \{m_1, m_2, \ldots, m_n\}$, todas distintas, ou seja $m_i \neq m_j$, para $i, j = 1, \ldots, n$, encontradas em um *corpus* de comentários de um fórum. Considere que existem subconjuntos $M_1, \ldots, M_K$ de M, tais que todas as menções em M_i, para $i = 1, \ldots, K$, se referem ao mesmo jogo j_i. O problema de desambiguação consiste em associar todas as menções em M_i a um mesmo nome canônico c_i que representa o jogo j_i em uma lista de nomes canônicos de jogos previamente existente.

Nossa abordagem para resolver este problema, foi baseada no método descrito

por [Yao & Sun, 2015], onde os autores propõem um método baseado em regras para desambiguação de menções a smartphones feitas em comentários de um fórum. Apesar da ideia geral de nosso método ser bastante similar a da proposta nesse trabalho, o conjunto de regras utilizado é bastante distinto.

De forma geral, nosso método consiste em avaliar cada uma das menções encontradas por um método de NER, tais como os apresentados nos Capítulos 3 e 4, e tentar associar esta menção com um nome canônico de jogo em uma lista previamente obtida. No nosso caso, esta lista corresponde à lista de jogos disponíveis na DBpedia [1] e cada menção é avaliada por um conjunto de regras avaliadas em sequência. Se a menção é associada a uma forma canônica por uma regra, a menção é considerada desambiguada. Caso contrário, a menção é avaliada pela próxima regra. Ao final do processo, podem haver menções que não foram desambiguadas.

Antes do processo de desambiguação iniciar, realizamos um pré-processamento em todas as menções, assim como na lista de nomes canônicos de jogos. Este pré-processamento consiste na remoção de todas as pontuações e na transformação de todas as palavras para letras minúsculas. Além disso, no domínio de jogos, as versões dos diversos jogos são identificadas por algarismos arábicos ou romanos de forma indistinta. Assim, em nosso pré-processamento, normalizamos todos os algarismos romanos encontrados para algarismos arábicos.

A seguir apresentamos as regras de desambiguação que utilizamos e em seguida descrevemos como elas são aplicadas. Posteriormente, no Capítulo 6, apresentamos resultados experimentais obtidos com o nosso método.

5.2 Regras de Desambiguação

Apresentamos a seguir as regras que utilizamos para desambiguar as menções a jogos e alguns exemplos de sua utilização.

- **Regra 1**: Casamento Exato de Menções. De acordo com esta regra, uma menção m_i é associada a um nome canônico c_j, se $m_i = c_j$. Senão, m_i não é desambiguada. Ao aplicar esta regra, tentamos casar diretamente a menção a um nome canônico

[1] http://web.informatik.uni-mannheim.de/DBpediaAsTables/DBpediaClasses.htm

de jogo, ou seja, a menção é o próprio nome canônico do jogo, por exemplo, “Killzone”, “Call of Duty”, “Far Cry”, etc.

- **Regra 2**: Casamento Exato de Acrônimos. Segundo esta regra, uma menção m_i é associada a um nome canônico c_j, se a concatenação das palavras que formam m_i casa com a forma siglada de um nome canônico c_j. Senão, m_i não é desambiguada nesta regra. Como exemplo, considere a menção “GTA 5”, cuja sua concatenação gera a palavra “GTA5”. Esta menção casa de maneira exata com a forma siglada do nome canônico “Grand Theft Auto 5”. Note que se a menção contiver apenas uma palavra, a concatenação resulta na própria palavra. Por exemplo, a menção “TLOU” casa com a forma siglada do nome canônico “The Last of Us”. Como outro exemplo, considere a menção “SSFIV”. Ela deveria casar com o nome canônico “Super Street Fighter 4”. Para tratar casos como este, esta regra também considera formas sigladas de nomes canônicos onde números arábicos são convertidos para romanos. É importante notar que existem casos em que as formas sigladas de vários nomes canônicos casam de maneira exata com a menção. Nestes casos, baseado em observações feitas na base de testes, consideramos o nome canônico mais curto. Por exemplo, considere a menção “COD” e os nomes canônicos de jogos “Call of Duty” e “Castle of Dragon” que possuem a mesma forma siglada da menção. Neste exemplo, o nome canônico de jogo “Call of Duty” será o jogo a ser desambiguado para a menção “COD”, visto que “Call of Duty” é o nome mais curto em relação ao “Castle of Dragon”.

- **Regra 3**: Casamento Aproximado de Menções baseado em Palavras. De acordo com esta regra, uma menção m_i é associada a um nome canônico c_j, se c_j é o nome canônico mais similar a m_i de acordo com a função de similaridade aproximada de strings SoftTFIDF [Cohen et al., 2003, Bilenko et al., 2003]. Se não for encontrado nenhum nome canônico para o qual a função SoftTFIDF retorna score de similaridade maior que 0, m_i não é desambiguada nesta regra. O detalhamento desta função é descrita na Seção 2.3. Como exemplo, considere a menção “Black Ops 2”. A função SoftTFIDF retorna scores de similaridade diferentes de 0 para vários nomes canônicos, tais como “Call of Duty: Black Ops 2”, “Call of Duty: Black Ops”, “Delta Force: Black Hawk Down”, etc. Dentre estes, o maior score é obtido para “Call of Duty: Black Ops 2”, sendo este o nome canônico escolhido. Nesta

regra também é importante notar que existem casos em que vários nomes canônicos apresentam o mesmo valor como sendo o maior escore computado pela função SoftTFIDF. Nestes casos, a menção é considerada não desambiguada, e é aplicada a Regra 3.1 de Casamento Aproximado de Menções baseado em Edição, descrita na Seção 2.3.

- **Regra 3.1**: Casamento Aproximado de Menções baseado em Edição. De acordo com esta regra, uma menção m_i é associada a um nome canônico c_j, se c_j é o nome canônico mais similar a m_i de acordo com a função de Levenstein para distância de edição entre strings [Cohen et al., 2003, Navarro, 2001]. Ao invés de considerar todas as menções, esta regra considera somente um conjunto de menções candidatas fornecidas pela Regra 3, que é aplicada antes desta regra. O detalhamento desta função é descrita na Seção 2.3. Como um exemplo, considere a menção "Zelda" e os nomes canônicos "The Legend of Zelda", "The Legend of Zelda: Ocarina of Time", "Zelda II: The Adventure of Link", etc., os quais possuem scores iguais computados pela função de similaridade SoftTFIDF. Neste exemplo, o jogo "The Legend of Zelda" possui menor Distância de Levenstein em relação aos outros nomes e por isso é o nome canônico associado à menção "Zelda".

5.2.1 Aplicação das Regras de Desambiguação

O processo de desambiguação das menções é feito através de uma iteração sobre o conjunto M de menções encontradas no *corpus* de comentários. Na Figura 5.1, apresentamos um fluxograma de aplicação das regras para cada menção encontrada.

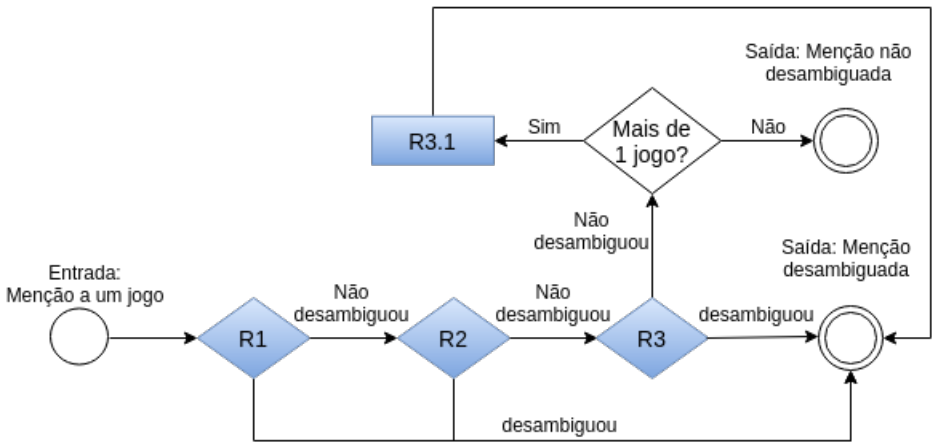

Figura 5.1: Fluxograma das regras utilizadas pelo método de desambiguação.

De início, cada menção a ser desambiguada é processada na Regra 1. Caso a menção seja desambiguada nesta regra, a próxima menção é processada, caso contrário, são executadas sequencialmente as Regras 2, 3 e 3.1. Observando o fluxograma da Figura 5.1, podemos perceber que o único caso em que nosso método de NED deixa de desambiguar uma menção é quando a menção não é desambiguada pelas Regras 1, 2 e 3 executadas em sequência, e na Regra 3 não houve nenhum nome canônico de jogo para o qual a função de similaridade SoftTFIDF retornou score maior que 0. Assim, a quantidade de jogos casados nesta regra é igual a zero e, portanto, a menção processada é considerada não desambiguada.

Capítulo 6

Experimentos

Neste capítulo, avaliamos os dois métodos de NER desenvolvidos neste trabalho, explicados nos Capítulos 3 e 4, bem como nosso método de desambiguação de menções a nomes de jogos explicado no Capítulo 5. Iniciamos apresentando a configuração da coleção de teste, na qual realizamos nossos experimentos e apresentamos os critérios e as métricas de avaliação utilizados.

6.1 Coleção de Teste

Para avaliar os dois métodos de NER apresentados neste trabalho (CRF-Games e ProdSpot-Games), assim como o método de desambiguação de menções a nomes de jogos, utilizamos uma coleção de dados composta por aproximadamente 200 mil comentários de usuários coletados do fórum *Gamespot*[1] no período de seis meses, compreendido entre Março a Setembro de 2015.

A partir desse conjunto de comentários, selecionamos aleatoriamente um subconjunto contendo 250 comentários para serem rotulados manualmente. Nestes comentários, foram marcadas 252 menções a nomes de jogos, o que resulta em uma menção por comentário, em média. A Tabela 6.1 mostra a configuração da coleção de teste.

A partir dessa tabela podemos dizer que em aproximadamente metade dos comentários existe pelo menos uma menção a nome de jogo. Uma outra característica que pode ser observada na Tabela 6.1 é que temos uma quantidade pequena de variações de menções a nomes de jogos para cada nome canônico de jogo. Isso pode ser observado comparando a

[1] http://www.gamespot.com/

Coleção de Teste	Total
Quantidade de comentários	250
Quantidade de menções	252
Quantidade de comentários que possuem menções	118
Quantidade de sentenças que possuem menções	189
Quantidade de jogos diferentes mencionados	139
Quantidade de menções diferentes	168

Tabela 6.1: Configuração da coleção de teste.

quantidade de jogos diferentes mencionados, 139, pela quantidade de menções diferentes na coleção de teste, 168. Um aspecto relevante que vale ser ressaltado, é que existem jogos que possuem diferentes formas de superfície utilizadas pelos usuários para fazer referência a eles e outros jogos possuem poucas formas distintas em serem mencionadas em nossa coleção teste. Além disso, há formas de superfície que são utilizadas para mencionar mais de um jogo, dessa forma, não sendo contabilizadas para a quantidade de menções diferentes em nossa coleção de teste.

6.2 Metodologia dos Experimentos

Nossos experimentos foram realizados da seguinte maneira: a coleção de dados rotulada com menções a nomes de jogos foi processada de acordo com as estratégias de preparação das entradas descrita na Seção 3.2.3, observe o exemplo da Figura 3.3; após executarmos os modelos treinados baseados no CRF gerados a partir dos dois métodos de NER (CRF-Games e ProdSpot-Games) sobre a coleção de teste, foram geradas sentenças rotuladas como as do exemplo da Figura 3.4; a partir dessas sentenças, construímos um gabarito em que associamos manualmente cada menção encontrada na coleção de teste ao seu nome canônico de jogo; após isso, medimos a qualidade dos métodos de reconhecimento e desambiguação de menções a nomes de jogos utilizando as métricas de Precisão (Pr), Revocação (Rc) e Medida-F1 (F_1), que são amplamente utilizadas para medir o desempenho de métodos em tarefas relacionadas ao reconhecimento e desambiguação de entidades nomeadas [Vieira & da Silva, 2015, Putthividhya & Hu, 2011, Yao & Sun, 2015, Wu et al., 2012]. Para contabilizar estas métricas em relação ao método supervisionado CRF-Games, utilizamos validação cruzada de 10-*folds*.

Para avaliar o desempenho de nosso método de desambiguação de menções a nomes

de jogos, realizamos três tipos de experimentos, considerando: (1) o caso ideal, supondo que um método de NER conseguiu identificar como menções a nomes de jogos todas as menções rotuladas da coleção de teste. Neste caso queremos avaliar nosso método de desambiguação independentemente dos métodos de NER; (2) considerando a resposta do método CRF-Games. Para este caso, pegamos o melhor modelo CRF treinado na validação cruzada de 10-*folds*; e (3) considerando a resposta do método ProdSpot-Games. Nestes dois últimos casos, queremos avaliar nosso método de desambiguação em relação às menções a jogos que os métodos de NER nos fornecem.

6.3 Experimentos com os Métodos de NER

Nesta seção avaliamos os dois métodos de NER (CRF-Games e ProdSpot-games) implementados neste trabalho em relação à identificação correta de menções a nomes de jogos nos comentário da coleção de teste. Os resultados experimentais são mostrados na Tabela 6.2

Métodos	*Pr*	*Rc*	F_1
CRF-Games	0,87	0,76	0,81
ProdSpot-Games	0,99	0,31	0,47

Tabela 6.2: Resultados dos experimentos de reconhecimento de menções a jogos pelo CRF-Games e ProdSpot-Games.

Nos resultados reportados na Tabela 6.2 para os dois métodos de NER, observamos claramente que o CRF-Games teve um desempenho geral superior comparado ao ProdSpot-games. Contudo, em relação à *Pr*, os dois métodos tiveram um bom desempenho, sendo que o ProdSpot-Games é mais preciso que o CRF-Games, alcançando próximo de $1,0$ nesta métrica, enquanto o CRF-Games alcançou 0,87. Já em relação à *Rc*, o CRF-Games é bem superior ao ProdSpot-Games, visto que o CRF-Games consegue identificar muito mais menções a jogos do que o ProdSpot-Games. Nesta métrica, o CRF-Games atingiu 0,76, enquanto o ProdSpot-Games obteve 0,31, ou seja, o CRF-Games obteve mais que o dobro se comparado à mesma métrica em relação ao ProdSpot-Games.

O bom desempenho do CRF-Games nesta coleção de teste pode ser explicado devido este método ser supervisionado, onde no treinamento dos modelos gerados, cada modelo foi treinamento com diferentes formas de superfície marcadas como menções a jogos na

coleção de teste, o que contribuiu para o seu bom desempenho nesta base.

O desempenho mais baixo do ProdSpot-Games em relação à *Rc* pode ser explicado devido o mesmo não conseguir generalizar tanto para outros nomes canônicos de jogos senão àqueles que foram passados como sementes para treinar um modelo baseado no CRF, isto é, o modelo treinado ficou muito específico para os nomes canônicos de jogos que foram submetidos como sementes. Essa evidência fica ainda mais clara nos resultados experimentais reportados na Seção 6.4.

6.4 Experimentos com o Método de NED

Nesta seção, avaliamos nosso método de desambiguação de menções a nomes de jogos considerando três cenários distintos. No primeiro, consideramos o caso ideal, em que todas as menções a jogos foram reconhecidas por um método de NER. Neste cenário, avaliamos o método de desambiguação utilizando todas as menções existentes na coleção de teste. No segundo cenário, a avaliação é feita usando somente as menções encontradas pelo CRF-Games e, no terceiro cenário, são consideradas somente as menções encontradas pelo ProdSpot-Games.

Na Tabela 6.3, mostramos os resultados experimentais em relação as métricas de *Pr*, *Rc* e F_1 nos três cenários.

Métodos	*Pr*	*Rc*	F_1
Ideal	0,83	0,82	0,82
CRF-Games	0,84	0,83	0,83
ProdSpot-Games	0,97	0,97	0,97

Tabela 6.3: Resultados dos experimentos em relação à desambiguação considerando o caso ideal e as respostas dos métodos CRF-Games e ProdSpot-Games.

Observando os resultados dos experimentos mostrados na Tabela 6.3, percebemos que, nos três cenários, as métricas de *Pr*, *Rc* e F_1 tiveram valores expressivos, estando acima de 0,8 nos três cenários avaliados. Os valores destas métricas para o cenário ideal e para o cenário onde é o usado o CRF-Games estão muito próximas, evidenciando que o método de NED teve comportamento semelhante em ambos os cenários. Já em relação ao resultado do ProdSpot-Games, os valores das três métricas analisadas ficaram bem acima dos demais cenários, o que, em princípio, nos permitiria dizer que este é o melhor cenário

de desambiguação. Todavia, deve ser considerado que, como discutido na Seção 6.3, o ProdSpot-Games não conseguiu identificar a maioria das menções ambíguas. Assim, o método de NED utilizou, na maioria das menções neste cenário, a Regra 1.

Para detalhar melhor os resultados da desambiguação, na Figura 6.1 comparamos a eficácia de cada regra de desambiguação nos três cenários experimentais definidos neste trabalho.

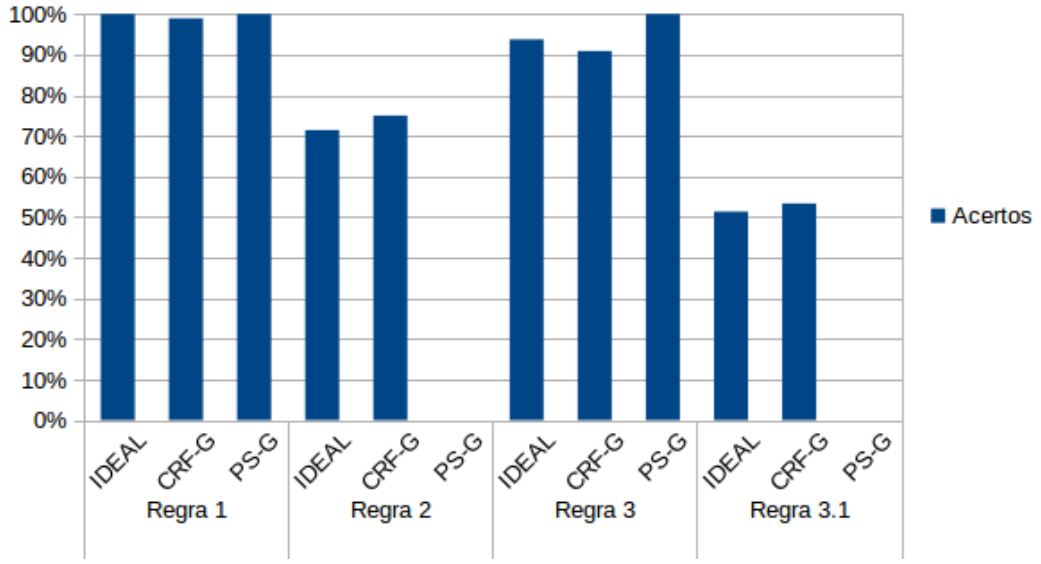

Figura 6.1: Eficácia de cada regra de desambiguação nos três cenários considerados.

Neste gráfico, percebemos que nos três cenários, a Regra 1 de Casamento Exato de Menções foi a mais eficiente, desambiguando corretamente quase 100% das menções a jogos nela casadas. A segunda regra mais eficaz foi a Regra 3, de Casamento Aproximado de Menções baseado em Palavras, a qual desambiguou em média 90% das menções a jogos nela casadas. Nesta regra, o método de NED obteve seu melhor resultado considerando a resposta do ProdSpot-Games (PS-G), tendo desambiguado 100% das menções, enquanto que para os outros dois cenários, Ideal e CRF-Games, o método de NED desambiguou um pouco mais de 90% das menções. A Regra 2 de Casamento Exato de Acrônimos foi a terceira mais eficaz, desambiguando a maioria das menções a jogos nela casada, considerando o caso Ideal, como também no cenário do CRF-Games.

Observamos também neste gráfico que, em relação à resposta do ProdSpot-Games, não houve menção a ser desambiguada. A não desambiguação das menções na Regra 2 pelo método de NED é devido a nenhuma menção ter sido casada nesta regra, o que reforça o resultado da baixa *Rc* do ProdSpot-Games mostrada na Tabela 6.2, em que ele não conseguiu generalizar tanto para outras menções utilizadas para referenciar jo-

gos senão àquelas das sementes. Por fim, a Regra 3.1 foi a que, dentre as quatro, menos desambiguou corretamente as menções a nomes de jogos nela casadas. Nesta regra, em média, cerca de 50% das menções foram desambiguadas, com exceção do resultado do ProdSpot-Games. Isso ocorreu devido ao fato de o método de NED não conseguir desambiguar corretamente as menções a jogos duvidosas, no qual as menções obtiveram o mesmo valor de score na Regra 3, de acordo com o fluxograma de aplicação das regras de desambiguação mostrado na Figura 5.1. O gráfico da Figura 6.1 mostra uma evidência de que a maioria das menções a jogos reconhecidas pelo ProdSpot-Games são os próprios nomes canônicos de jogos, e por esse motivo foram desambiguadas na Regra 1. Essa evidência fica ainda mais clara analisando a configuração da quantidade de menções reconhecidas e desambiguadas em cada regra para os três cenários de experimentos mostrados na Tabela 6.4.

Na Tabela 6.4, apresentamos o número de menções desambiguadas por cada regra. Considere *R*, *M*, *A*, *E*, *ND* e *T* como sendo Regras, Menções, Acertos, Erros, Não Desambiguou e Total, respectivamente.

R	*M*	*A*	*E*	*ND*
1	109	109	0	0
2	70	50	20	0
3	36	30	2	4
3.1	37	19	18	0
T	252	208	40	4

(a) Ideal.

R	*M*	*A*	*E*	*ND*
1	95	94	1	0
2	60	45	15	0
3	25	20	2	3
3.1	30	16	14	0
T	210	175	32	3

(b) CRF-Games.

R	*M*	*A*	*E*	*ND*
1	63	63	0	0
2	0	0	0	0
3	3	3	0	0
3.1	2	0	2	0
T	68	66	2	0

(c) ProdSpot-Games.

Tabela 6.4: Experimentos para desambiguação de menções a jogos pelo método de NED.

Na Tabela 6.4a, para o cenário Ideal percebe-se que a Regra 1 foi a que mais desambiguou menções a jogos corretamente, sendo casadas nesta regra 109 menções de um total de 252. Para a mesma regra, na Tabela 6.4b, agora em relação ao cenário do CRF-Games, 94 menções a jogos foram desambiguadas corretamente, ou seja, quase a metade de todas as menções a jogos reconhecidas pelo CRF-Games foram desambiguadas nesta regra. Vale ressaltar que o método de desambiguação errou em desambiguar uma menção a jogo nesta Regra. Isso ocorreu devido o modelo treinado do CRF-Games identificar apenas uma palavra da menção completa rotulada na coleção de teste como sendo uma menção a nome de jogo e essa palavra ser um nome canônico de jogo presente na lista de nomes canônicos de jogos disponível na DBPedia. Esta menção a nome de jogo foi “Perfect

Dark" rotulada em nossa coleção de teste, no entanto, o modelo do CRF-Games identificou apenas a palavra "Dark" como sendo uma menção a jogo e esta palavra "Dark" é o nome canônico de um jogo. Por fim, na Tabela 6.4c, é mostrado o resultado para o ProdSpot-Games, em que este desambiguou 63 menções a jogos de um total de 68. Neste caso, quase a totalidade de todas as menções a jogos foram desambiguadas nesta regra. Na Regra 2, os resultados foram muito semelhante para os cenários Ideal e CRF-Games, onde a proporção de *A* e *E* foi basicamente a mesma. No entanto, considerando cenário do ProdSpot-Games, na Tabela 6.4c, percebe-se que o método de NED não desambiguou nenhuma menção a jogo nesta regra. Isso explica o motivo de não haver acertos na Regra 2 mostrada no gráfico da Figura 6.1. Para a Regra 3, em todos os cenários, o método de NED conseguiu desambiguar corretamente as menções casadas nesta regra. A proporção de *A* e *E* basicamente se mantém em relação ao caso Ideal e o CRF-Games. Já considerando no ProdSpot-Games, foi possível desambiguar as 3 menções a jogos nela casada, obtendo-se assim *Pr* de 100%. Por fim, na Regra 3.1, praticamente a metade das menções a jogos foram desambiguadas corretamente considerando os cenários Ideal e CRF-Games. Contudo, em relação ao ProdSpot-Games o método de NED desambiguou incorretamente as menções a jogos que não foram desambiguadas pala Regra 3.1. Nela, 2 menções a jogos foram desambiguadas. Isso é mostrado na Tabela 6.4c. Esse é o motivo de não existirem acertos nesta regra no gráfico da Figura 6.1, pois essa figura representa apenas a porcentagem de acertos para cada um dos três cenários em relação a cada uma das quatro regras de desambiguação.

Capítulo 7

Ferramenta GameSpotter

Neste capítulo, apresentamos a ferramenta *GameSpotter* desenvolvida neste trabalho, como um estudo de caso voltado ao domínio de jogos. Descrevemos a sua arquitetura geral, onde detalhamos cada etapa do processo, desde a coleta dos dados até o desenvolvimento da interface Web da aplicação. Por fim, são apresentadas algumas funcionalidades da ferramenta, bem como algumas estatísticas em relação à coleta de comentários no fórum GameSpot.

7.1 Arquitetura Geral da Ferramenta GameSpotter

A Ferramenta GameSpotter é composta basicamente por dois módulos principais, sendo eles: *1. Geração do Modelo* e *2. Aplicação Gamespotter*. Na Figura 7.1, é mostrada a arquitetura geral de nossa ferramenta.

O primeiro módulo da ferramenta, *1. Geração do Modelo* descrito no Capítulo 4 é responsável por gerar um modelo baseado no CRF para identificação de menções a nomes de jogos nos comentários extraídos do fórum GameSpot. O segundo módulo, *2. Aplicação Gamespotter* é responsável pela coleta periódica de comentários no fórum para que o modelo CRF treinado seja aplicado sobre eles. Após a aplicação do modelo sobre os comentários, tem-se sentenças rotuladas contendo ou não menções a nomes de jogos. Para as sentenças que contenham pelo menos uma menção a nome de jogo, utilizamos um limiar para filtrar essas sentenças. Caso a confiança de classificação pelo modelo CRF nessas sentenças seja abaixo desse limiar, essas sentenças são desprezadas, caso contrário, as sentenças filtradas contendo uma ou mais menções a nomes de jogos são passadas como

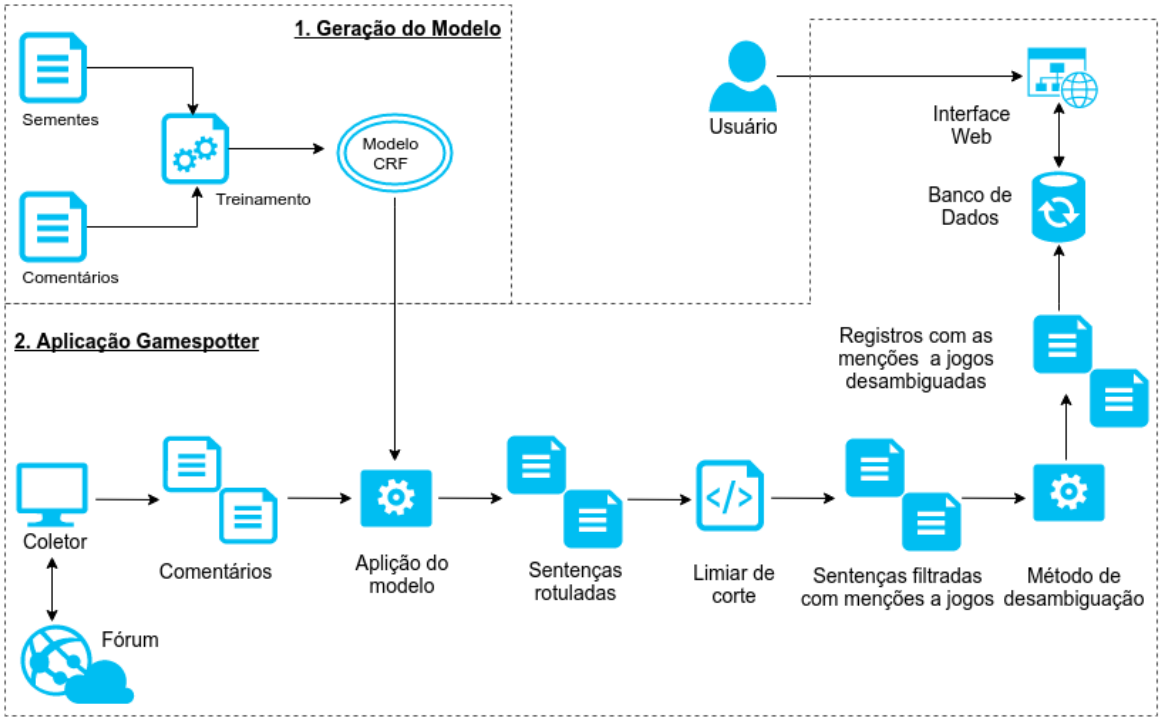

Figura 7.1: Arquitetura geral da ferramenta GameSpotter.

entrada para o método de NED baseado em regras, o qual tenta associar automaticamente essas menções a nomes canônicos de jogos contidos em uma base de conhecimento. A saída do processo de desambiguação pelo método de NED é um registro composto por cinco campos para cada menção, sendo eles: menção a um nome de jogo, a url da página onde essa menção foi identificada, o nome canônico do jogo para a menção, o comentário e a sentença onde essa menção ocorre. Os dados sobre os comentários e as urls para cada sentença com menção a nome de jogo são previamente processados, assim quando uma menção é desambiguada, pegamos a url e o comentário previamente processados. Esses registros gerados após a tarefa de NED são inseridos em um banco de dados que constantemente é atualizado. Por fim, uma interface Web foi desenvolvida com o propósito de fornecer aos usuários uma interface de consulta para que eles possam pesquisar por suas menções a jogos de seu interesse. Nas subseções a seguir, explanaremos com mais riqueza de detalhes cada um dos componentes do módulo *2. Aplicação Gamespotter*.

7.1.1 Coletor

Como parte da coleta de comentários, foi implementado um coletor que é executado periodicamente para pegar os comentários postados por usuários do fórum alvo. Este coletor possui três módulos principais: (i) módulo responsável por acessar o fórum do site e pegar

os comentários dos usuários, (ii) módulo responsável por guardar as páginas coletadas em um cache. Esta etapa é bastante útil pois a cada vez que o coletor é executado, primeiramente ele verifica se a página a ser coletada está contida no cache. Se verdadeiro, extrai os comentários, caso contrário, baixa a página do site, extrai os comentários e insere o conteúdo html da página no cache, evitando uma sobrecarga de requisições a serem enviadas ao site e (iii) módulo responsável pelo pré-processamento dos comentários, onde é transformado o conteúdo HTML dos comentários em codificação UTF-8.

7.1.2 Aplicação do Modelo

Com o intuito de se obter sentenças de comentários rotuladas, o modelo CRF treinado é aplicado sobre os novos comentários coletados do fórum. Neste caso, consideramos os modelos gerados pelos métodos CRF-Games e ProdSpot-Games descritos nos capítulos 3 e 4, sendo aplicado ou um ou o outro de cada vez. A saída deste processo é sentenças rotuladas em que se possui ou não menções a nomes de jogos. Observe o exemplo da Figura 3.4 no Capítulo 3.

7.1.3 Limiar de Corte

A fim de se obter sentenças rotuladas com uma confiança razoável de certeza na classificação por um de nossos métodos de NER (CRF-Games, ProdSpot-Games), utilizamos um limiar de corte. Após o processo da aplicação do modelo nos comentários em que se tem sentenças rotuladas, utilizamos a confiança do modelo CRF treinado em classificar tal sentença. Se a confiança do modelo para essa sentença for abaixo deste limiar, esta sentença é desprezada. O valor de corte escolhido através de experimentos realizados foi de 0.75.

7.1.4 Método de Desambiguação de Menções a Jogos

Nesta etapa dispomos sentenças que contêm menções a nomes de jogos identificadas por um dos métodos de NER utilizados neste trabalho. O método de NED recebe como entrada essas menções a nomes de jogos e tenta mapeá-las a seus respectivos nomes canônicos contidos em uma base de conhecimento. Após o processamento dessas menções a jogos, obtém-se registros com as menções desambiguadas, caso o método de NED con-

siga desambiguar tais menções. Caso contrário, são gerados registros com uma *flag* para o campo do nome canônico para cada menção que representa a não desambiguação da menção ou das menções analisadas.

7.1.5 Banco de Dados

Para armazenar os registros gerados após o processamento das menções pelo método de desambiguação de menções a nomes de jogos, foi criado um banco de dados para guardar esses registros. O SGBD utilizado para o gerenciamento do banco de dados foi o Mysql [1]. Para a tarefa de atualização desta base de dados foi implementado um *script* em linguagem PHP que verifica se tais registros já estão armazenados no banco de dados, caso contrário, esses registros são inseridos no banco de dados que constantemente é atualizado.

7.1.6 Interface Web

Esta é a interface principal da Ferramenta *GameSpotter* desenvolvida neste trabalho, uma forma visual de apresentarmos todo o processo desenvolvido, que vai desde a coleta de dados, passando pelo método de NER, filtragem de sentenças a partir do limiar de corte, aplicação do método de NED até a atualização do banco de dados. Essa é a interface que o usuário utiliza para realizar suas pesquisas por menções a nomes de jogos. A Figura 7.2 apresenta a interface de consulta.

Na Figura 7.2 observa-mos alguns campos da interface realçados em vermelho. Em **1**, na barra de menu, são mostradas as opções *Home*, que ao ser clicada a página é redirecionada para a página principal; *estatísticas*, responsável por mostrar as estatísticas da ferramenta GameSpotter, como por exemplo a quantidade de menções, sentenças e comentários armazenados no banco de dados, e o campo *sobre*, o qual descreve o que é a ferramenta GameSpotter, bem como os autores responsáveis por sua idealização. Em **2**, na área de pesquisa, é mostrado o campo de consulta, em que os usuários podem pesquisar por suas menções a jogos de seu interesse e em **3**, no campo menções ordenadas, é mostrada uma tabela com as menções a nomes de jogos ordenadas pela frequência com que elas ocorrem nos comentários. Essa tabela também pode ser utilizada como um campo de pesquisa rápida.

[1] https://www.mysql.com/

Figura 7.2: Interface de consulta.

7.2 Funcionalidades da Ferramenta

Como uma funcionalidade geral da Ferramenta GameSpotter, apresentamos uma consulta por uma menção a nome de jogo. A Figura 7.3 mostra a consulta pela menção "gtav". Na tabela à esquerda é mostrada a menção realçada em amarelo juntamente com a frequência em que esta menção ocorre nos comentários. Na tabela a direita é mostrada todas as sentenças em que esta menção ocorre bem como o campo "ver", utilizado para visualizar mais detalhes de uma dada sentença. Pode-se observar ainda que a menção consultada é realçada em amarelo nessas sentenças.

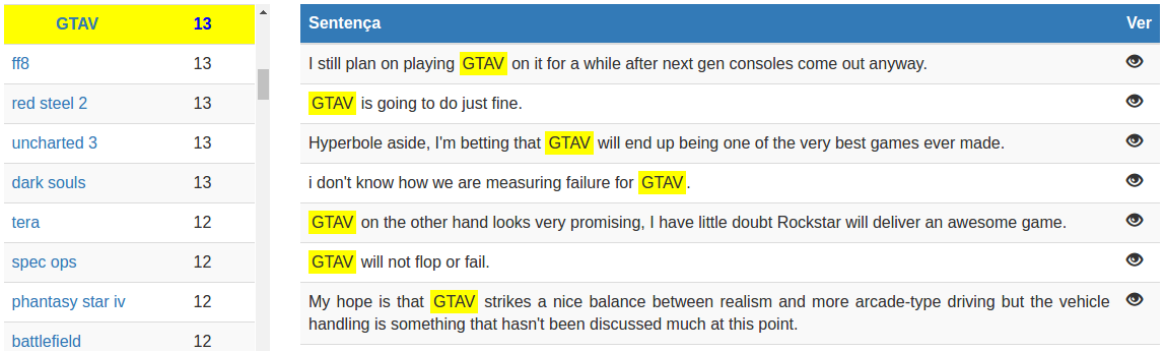

Figura 7.3: Consulta geral por uma menção a nome de jogo.

Para se obter mais detalhes acerca de uma determinada menção em uma sentença,

Figura 7.3, consultamos, por exemplo, a última sentença da lista de resposta retornada e clicamos no campo “ver”. O resultado mostra informações adicionais, tais como o nome canônico para a menção pesquisada, a menção realçada em amarelo, todas as outras menções destacadas em azul que foram identificadas juntamente com a menção alvo e também fornece um link para o site onde esta sentença e o comentário foram coletados.

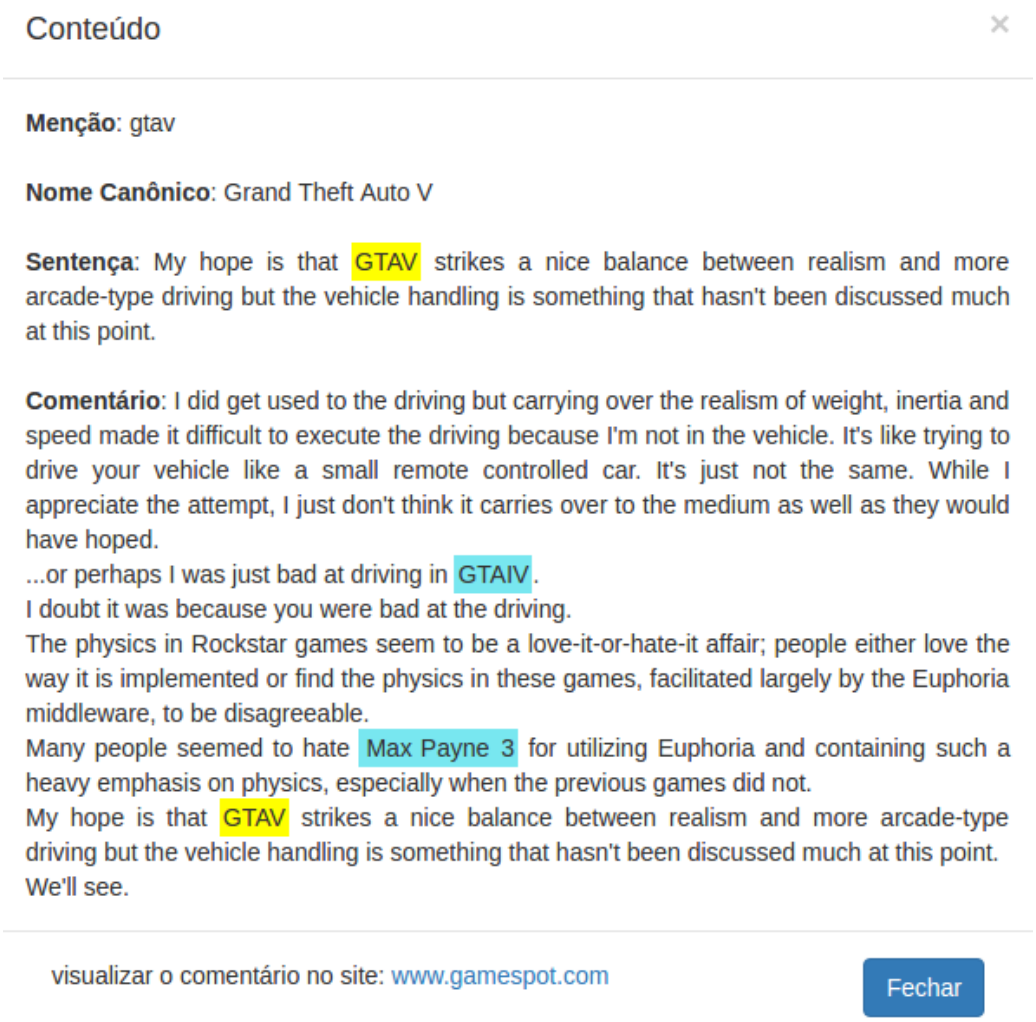

Figura 7.4: Detalhes adicionais em uma determinada sentença.

Na Figura 7.4, podemos observar que o nome canônico do jogo: “Grand Theft Auto V” foi associado à menção “gtav” por nosso método de NED. Existem ainda as menções “GTAIV” e “Max Payne 3” realçadas na cor azul, as quais fazem referência a outros dois jogos.

7.3 Estatísticas para a Ferramenta GameSpotter

Nesta seção apresentamos algumas estatísticas para a ferramenta *GameSpotter*. Durante um período de 30 dias, Março de 2016, coletamos diariamente 19 páginas do fórum gamespot[2], que ao final deste período resultou em um total 570 páginas coletadas com 10982 comentários distintos. Após esta coleta, executamos os dois métodos de NER (CRF-Games e ProdSpot-Games), assim como o nosso método de NED nestes comentários. Em relação aos métodos de NER, consideramos apenas as sentenças com menções a jogos em que o modelo CRF rotulou estas sentenças com confiança acima de nosso limiar preestabelecido, em que adotamos através de experimentos de validação prévio o valor 0.75. Lembramos que este limiar é usado apenas para a nossa ferramenta GameSpotter. Posteriormente unimos as menções distintas identificadas por ambos os métodos de NER com o intuito de demonstrar que as menções a nomes de jogos geralmente se concentram em comentários que possuem uma quantidade relevante de palavras. Ressaltamos que para o método CRF-Games, utilizamos o melhor modelo CRF treinado a partir da técnica de validação cruzada de 10-*folds*.

Vale ressaltar que não sabemos previamente se todas as menções a nomes de jogos identificadas pelos modelos CRF treinados, a partir dos dois métodos de NER implementados neste trabalho, realmente são menções reais a jogos, visto que não possuímos um gabarito prévio dessas menções identificadas nesses comentários. No entanto, para o levantamento de nossas estatísticas em relação a nossa ferramenta GameSpotter, adotamos como válidas todas as menções a jogos identificadas.

7.3.1 Método CRF-Games

Ao executarmos o modelo gerado pelo CRF-Games nos comentários coletados, obtivemos um total de 2368 menções a nomes de jogos identificadas. Após o processamento dessas menções pelo método de desambiguação de menções a jogos, foram gerados registros conforme descrito na Seção 7.1.4. A Tabela 7.1 mostra a configuração da base para o método de extração CRF-Games.

Observando a Tabela 7.1 percebemos que a porcentagem de comentários que possuem pelo menos uma menção a nome de jogo identificada pelo modelo gerado a partir do CRF-

[2]http://www.gamespot.com/

CRF-Games	Total
Quantidade de total de comentários coletados	10982
Quantidade de comentários com menções	1458
Quantidade de menções a jogos identificadas	2368
Quantidade de menções a jogos desambiguadas	2163
Quantidade de menções a jogos não desambiguadas	205

Tabela 7.1: Estatísticas para o método de extração CRF-Games.

Games (1458) em relação à quantidade total de comentários coletados (10982) é próxima de 13,28%.

7.3.2 Método ProdSpot-Games

Ao executarmos o modelo CRF gerado pelo método ProdSpot-Games nos comentários coletados, obtivemos um total de 1307 menções a nomes de jogos identificadas e foram gerados a mesma quantidade de registros da mesma maneira como citado na Seção 7.1.4. A Tabela 7.2 mostra a configuração da base para o método de extração ProdSpot-Games.

ProdSpot-Games	Total
Quantidade de total de comentários coletados	10982
Quantidade de comentários com menções	853
Quantidade de menções a jogos identificadas	1307
Quantidade de menções a jogos desambiguadas	1307
Quantidade de menções a jogos não desambiguadas	0

Tabela 7.2: Estatísticas para o método de extração ProdSpot-Games.

Analisando a Tabela 7.2 percebemos que a porcentagem de comentários que possuem pelo menos uma menção a nome de jogo identificada pelo modelo gerado pelo ProdSpot-Games (853) em relação à quantidade total de comentários distintos coletados (10982) é próximo de 7,77%. Esta porcentagem é considerada pequena quando comparado ao total de comentários coletados. Além disso, Esta porcentagem de comentários distintos que possuem pelo menos uma menção a nome de jogo identificada é quase a metade em relação ao método CRF-Games, em que o CRF-Games encontra bem mais menções a jogos nos comentários do que o ProdSpot-Games.

7.3.3 União de menções entre o CRF-Games e o ProdSpot-Games

A fim de obtermos uma estatística de quantos comentários e menções a jogos conseguimos obter com a união de menções identificadas pelos dois métodos, realizamos a união das menções que o modelo gerado pelo CRF-Games identifica como sendo uma menção a jogo e o modelo gerado pelo ProdSpot-Games não identifica. Na Tabela 7.3 apresentamos estas estatísticas.

União entre CRF-Games e ProdSpot-Games	Total
Quantidade de total de comentários coletados	10982
Quantidade de comentários com menções	1635
Quantidade de menções a jogos identificadas	2718
Quantidade de menções a jogos desambiguadas	2513
Quantidade de menções a jogos não desambiguadas	205

Tabela 7.3: Estatísticas para a união entre os método de extração CRF-Games e ProdSpot-Games.

Com a nova configuração da Tabela 7.3, temos 1635 comentários distintos que possuem pelo menos uma menção a nome de jogo. Isso equivale à aproximadamente 14,89% do total de comentários distintos coletados. Na subseção a seguir mostramos as estatísticas em relação aos comentários com menções a nomes de jogos com essa nova configuração.

Gráfico da união entre o CRF-Games e o ProdSpot-Games

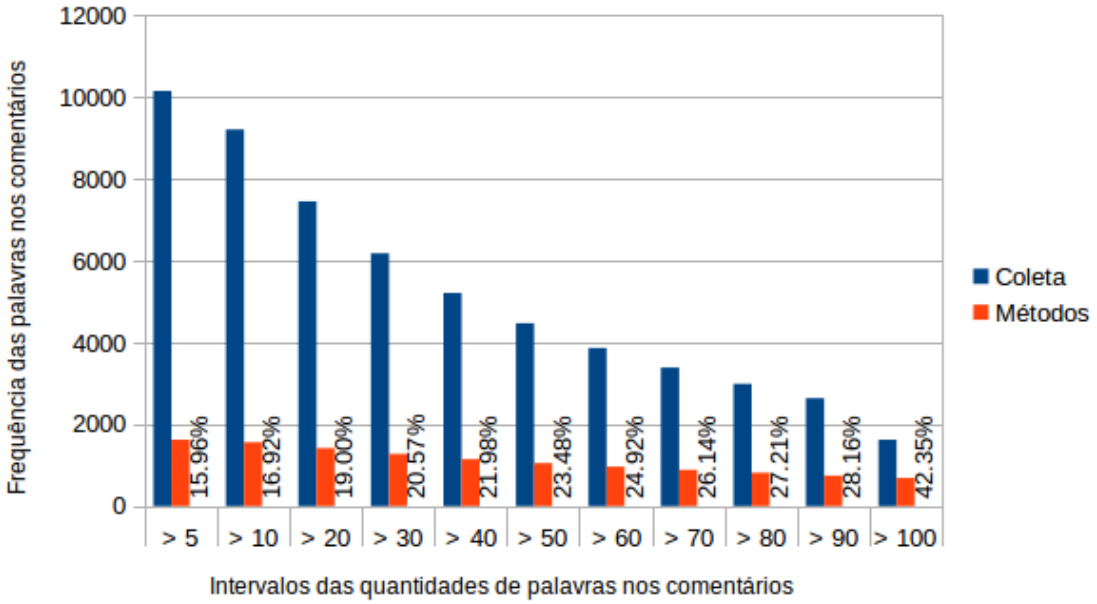

Figura 7.5: União de menções entre o ProdSpot-Games e o CRF-Games.

No gráfico da Figura 7.5 temos a relação entre a quantidade total de comentários coletados (10982) representada pela barra azul e a quantidade de comentários distintos que possuem pelo menos uma menção a nome de jogo (1635) representada pela barra vermelha, sobre o intervalo da quantidade de palavras que cada comentário possui. Definimos os intervalos entre os comentários que possuem mais de 5 palavras até os comentários que possuem mais de 100 palavras. Neste gráfico, percebemos que as menções a nomes de jogos se concentram principalmente em comentários que possuem uma quantidade relevante de palavras, onde os comentários com mais de 100 palavras que possuem menções a jogos representam quase 50% dos comentários em relação à coleta com a mesma quantidade de palavras. Isso nos dá um indício de onde é mais provável encontrar menções a jogos em comentários de usuários neste fórum.

Capítulo 8

Conclusão

Neste capítulo, resumimos as questões de pesquisa abordadas neste trabalho e apresentamos nossas conclusões finais. Por fim, sugerimos novas ideias de pesquisa direcionadas à trabalhos futuros.

8.1 Resultados Obtidos

Neste trabalho, estudamos questões de pesquisa relacionadas ao reconhecimento e desambiguação de menções de produtos em conteúdo gerado por usuários, utilizando como estudo de caso o domínio de jogos. Neste nosso estudo, procuramos abordar um domínio diferente do de produtos eletrônicos, domínio este que tem sido abordado por vários trabalhos na literatura recente [Vieira, 2016, Vieira & da Silva, 2015, Yao & Sun, 2014, Wu et al., 2012].

Esperamos com este nosso estudo de caso ter contribuído com a extensão da aplicabilidade destes métodos e, em particular, com o método proposto por [Vieira, 2016], que está em desenvolvimento em nosso grupo de pesquisa. Com o intuito de ajudar em nosso estudo de caso, implementamos uma ferramenta chamada GameSpotter que utiliza dois método de reconhecimento de entidades nomeadas (NER), para identificar nos comentários dos usuários as menções a nomes de jogos, e um método de desambiguação de entidades nomeadas (NED), para associar estas menções a seus respectivos nomes canônicos de jogos, estando estes nomes presentes em uma lista pré-definida. O desenvolvimento desta ferramenta teve um importante papel em nosso estudo de caso, pois, através dela, pudemos demonstrar o potencial de aplicação das técnicas de NER e NED

para lidar com um grande volume de conteúdo gerado por usuários. Por exemplo, dentre outras facilidades, através de nossa ferramenta o usuário pode pesquisar por menções a jogos de seu interesse, e obter os comentários relacionados a sua busca, sem ter que verificar manualmente os comentários postados no fórum. Tal tarefa seria custosa e sujeita a erros em um contexto de um fórum com milhões de comentários.

Para a tarefa de NER, desenvolvemos dois métodos alternativos, que foram utilizados na ferramenta e comparados experimentalmente. O primeiro, que chamamos de CRF-Games, é um método supervisionado que aplica de forma direta o conhecido modelo CRF [Sutton & McCallum, 2011]. O segundo, que chamamos de ProdSpot-Games, é uma adaptação do método ProdSpot [Vieira, 2016], que atualmente está em desenvolvimento no nosso grupo de pesquisa e que foi desenvolvido originalmente para identificação de menções a produtos eletrônicos.

Para a tarefa de NED, desenvolvemos um método de desambiguação baseado em regras a partir do método descrito em [Yao & Sun, 2015], que foi proposto para desambiguação de menções a smartphones. Apesar de similar de forma geral, este método utiliza regras específicas para o seu domínio-alvo, enquanto que para o nosso método foram desenvolvidas regras adequadas para o domínio de jogos.

Nossos resultados experimentais em relação aos métodos de NER desenvolvidos neste trabalho mostraram que tanto o CRF-Games quanto o ProdSpot-Games obtiveram uma boa precisão, tendo valores acima de 0,80. Porém, em relação à revocação, o ProdSpot-Games ficou muito abaixo em relação ao CRF-Games. O ProdSpot-Games obteve uma revocação de 0,31 e o CRF-Games 0,76. Esta baixa revocação pode ser explicada devido o ProdSpot-Games não conseguir generalizar tanto para outros nomes canônicos de jogos e diferentes formas de menções em relação as menções a jogos feitas pelos usuários em um fórum. Diferentemente do ProdSpot-Games, o CRF-Games é um método supervisionado treinado com instâncias marcadas manualmente e, com isso, pode encontrar muito mais menções a jogos.

Já em relação ao método de NED desenvolvido também neste trabalho, este mostrou-se eficiente em desambiguar corretamente as menções a nomes de jogos, obtendo médias de precisão, revocação e medida-F1, respectivamente, $0,88$, $0,87$ e $0,87$, em três cenários testados.

8.2 Trabalhos Futuros

Uma das possíveis direções futuras do presente trabalho é a sua expansão para outros domínios populares, como por exemplo: filmes, músicas, hotéis, livros, etc., ampliando assim, os cenários de aplicação dos métodos. Em particular, antevemos que outros domínios relacionados a produtos culturais, tais como livros e filmes, seriam potencialmente mais adequados para a expansão dos métodos aqui apresentados.

Além disso, é muito importante que os problemas aqui detectados com relação ao método ProdSpot-Games sejam investigados, pois, apesar de o CRF-Games ter tido um bom desempenho, o fato de que ele necessita de treinamento manual é um grande inconveniente. Assim, é necessário investir na melhoria do ProdSpot-Games para que atinja níveis de qualidade similares aos do CRF-Games, sem no entanto necessitar de treinamento manual. Uma possível abordagem para isto é utilizar auto-treinamento como em [Vieira & da Silva, 2015]. Nesta abordagem, novas formas de superfície são identificadas no corpus e adicionadas ao conjunto de sementes, a fim de treinar um novo modelo CRF com um conjunto maior de sementes distintas.

Com relação ao nosso método de NED, neste trabalho nos limitados a explorar as características locais sintáticas das menções e dos nomes canônicos. No entanto, existe ainda a possibilidade de explorar características de contexto dos comentário onde a menção ocorre no fórum. Por exemplo, determinas menções podem ser mais utilizadas em uma *thread* específica de comentários ou ser preferida por usuários específicos. Assim, características como a thread em que este comentário está inserido e autor do comentário no fórum podem ser consideradas.

Uma outra possível extensão interessante para este trabalho é a extração de aspectos [Hu & Liu, 2004, Rana & Cheah, 2016] relacionados aos jogos. Muitas vezes é importante para um usuário saber os aspectos referentes a um determinado jogo, como por exemplo: a sua qualidade gráfica, sua jogabilidade, qualidade de seu cenário, etc. O propósito por trás disso é enriquecer essas entidades, os jogos, com conteúdo extraído dos comentários em mídia social.

Referências Bibliográficas

[Auer et al., 2007] Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., & Ives, Z. (2007). *The Semantic Web: 6th International Semantic Web Conference, 2nd Asian Semantic Web Conference, ISWC 2007 + ASWC 2007, Busan, Korea, November 11-15, 2007. Proceedings*, chapter DBpedia: A Nucleus for a Web of Open Data, (pp. 722–735). Springer Berlin Heidelberg: Berlin, Heidelberg.

[Baeza-Yates & Ribeiro-Neto, 2013] Baeza-Yates, R. & Ribeiro-Neto, B. (2013). chapter Recuperação de Informação, Conceitos e Tecnologia das Máquinas de Busca, (pp. 614). Bookman.

[Bikel et al., 1997] Bikel, D. M., Miller, S., Schwartz, R., & Weischedel, R. (1997). Nymble: a high-performance learning name-finder. In *Proceedings of the fifth conference on Applied natural language processing* (pp. 194–201).: Association for Computational Linguistics.

[Bilenko et al., 2003] Bilenko, M., Mooney, R., Cohen, W., Ravikumar, P., & Fienberg, S. (2003). Adaptive name matching in information integration. *IEEE Intelligent Systems*, (5), 16–23.

[Bollacker et al., 2008] Bollacker, K., Evans, C., Paritosh, P., Sturge, T., & Taylor, J. (2008). Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data* (pp. 1247–1250).: ACM.

[Brown et al., 1992] Brown, P. F., deSouza, P. V., Mercer, R. L., Pietra, V. J. D., & Lai, J. C. (1992). Class-Based n-gram Models of Natural Language. *Computational Linguistics*, 18, 467–479.

[Christopher, 2011] Christopher, M. B. (2011). chapter Pattern Recognition & Machine Learning, (pp. 738). Springer.

[Cohen et al., 2003] Cohen, W. W., Ravikumar, P. D., Fienberg, S. E., et al. (2003). A comparison of string distance metrics for name-matching tasks. In *IIWeb*, volume 2003 (pp. 73–78).

[Dang et al., 2010] Dang, Y., Zhang, Y., & Chen, H. (2010). A lexicon-enhanced method for sentiment classification: An experiment on online product reviews. *Intelligent Systems, IEEE*, 25(4), 46–53.

[Feldman, 2013] Feldman, R. (2013). Techniques and applications for sentiment analysis. In *Communications of the ACM, Volume 56* (pp. 82–89).: ACM.

[Gimpel et al., 2011] Gimpel, K., Schneider, N., O'Connor, B., Das, D., Mills, D., Eisenstein, J., Heilman, M., Yogatama, D., Flanigan, J., & Smith, N. A. (2011). Part-of-speech tagging for twitter: Annotation, features, and experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2* (pp. 42–47).: Association for Computational Linguistics.

[Han & Zhao, 2009] Han, X. & Zhao, J. (2009). Named entity disambiguation by leveraging wikipedia semantic knowledge. In *Proceedings of the 18th ACM conference on Information and knowledge management* (pp. 215–224).: ACM.

[Hoffart et al., 2011] Hoffart, J., Yosef, M. A., Bordino, I., Fürstenau, H., Pinkal, M., Spaniol, M., Taneva, B., Thater, S., & Weikum, G. (2011). Robust disambiguation of named entities in text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (pp. 782–792).: Association for Computational Linguistics.

[Hu & Liu, 2004] Hu, M. & Liu, B. (2004). Mining opinion features in customer reviews. In *AAAI*, volume 4 (pp. 755–760).

[Jain et al., 2000] Jain, A. K., Duin, R. P., & Mao, J. (2000). Statistical pattern recognition: A review. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(1), 4–37.

[Jakob & Gurevych, 2010] Jakob, N. & Gurevych, I. (2010). Extracting opinion targets in a single-and cross-domain setting with conditional random fields. In *Proceedings of the 2010 conference on empirical methods in natural language processing* (pp. 1035–1045).: Association for Computational Linguistics.

[Jaro, 1989] Jaro, M. A. (1989). Advances in record-linkage methodology as applied to matching the 1985 census of tampa, florida. *Journal of the American Statistical Association*, 84(406), 414–420.

[Jiang, 2012] Jiang, J. (2012). Information extraction from text. In *Mining text data* (pp. 11–41). Springer.

[Kaplan & Haenlein, 2010] Kaplan, A. & Haenlein, M. (2010). Users of the world, unite! The challenges and opportunities of Social Media. In *ESCP, Business School* (pp. 59–68).: Business Horizons.

[Kietzmann et al., 2011] Kietzmann, J., Hermkens, K., McCarthy, I., & Silvestre, B. (2011). Social media? get serious! understanding the functional building blocks of social media. In *Kelley School of Business, Indiana Universit* (pp. 241–251).: Business Horizons.

[Koo et al., 2008] Koo, T., Carreras, X., & Michael, C. (2008). : (pp. 595–603).: ACM.

[Lafferty et al., 2001] Lafferty, J., McCallum, A., & Pereira, F. (2001). Conditional random fields, probabilistic models for segmenting and labeling sequence data. In *ICML - International Conference on Machine Learning*.

[Lavergne et al., 2010] Lavergne, T., Cappé, O., & Yvon, F. (2010). Practical very large scale CRFs. In *Proceedings the 48th Annual Meeting of the Association for Computational Linguistics (ACL)* (pp. 504–513).: Association for Computational Linguistics.

[Lee & Pang, 2008] Lee, L. & Pang, B. (2008). Opinion mining and sentiment analysis. In *Foundations and trends in information retrieval, Vol 2, no 1-2,pp. 1-35*.

[Levin, 2010] Levin, F. H. (2010). Desambiguaçã de autores em bibliotecas digitais utilizando redes sociais e programaçã genética. Master's thesis, Universidade Federal do Rio Grande do Sul - Programa de Pós Graduação., RS - Brasil.

[Li et al., 2013] Li, Y., Wang, C., Han, F., Han, J., Roth, D., & Yan, X. (2013). Mining evidences for named entity disambiguation. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 1070–1078).: ACM.

[Ling & Weld, 2012] Ling, X. & Weld, D. S. (2012). Fine-grained entity recognition. In *AAAI*.

[Liu, 2007] Liu, B. (2007). *Web data mining: exploring hyperlinks, contents, and usage data*. Springer Science & Business Media.

[Liu, 2012] Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1), 1–167.

[McCallum, 2002] McCallum, A. K. (2002). Mallet: A machine learning for language toolkit. Acessado em: 18-03-2016.

[Mitchell, 1997] Mitchell, T. M. (1997). chapter Machine Learning, (pp. 432). McGraw-Hill.

[Navarro, 2001] Navarro, G. (2001). A guided tour to approximate string matching. *ACM computing surveys (CSUR)*, 33(1), 31–88.

[Navarro et al., 2001] Navarro, G., Baeza-Yates, R., Sutinen, E., & Tarhio, J. (2001). Indexing methods for approximate string matching. *IEEE, Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*.

[Navigli, 2009] Navigli, R. (2009). Word sense disambiguation: A survey. *ACM Computing Surveys (CSUR)*, 41(2), 10.

[Putthividhya & Hu, 2011] Putthividhya, D. P. & Hu, J. (2011). Bootstrapped named entity recognition for product attribute extraction. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, CEMNLP* (pp. 1557–1567).: ACM.

[Rana & Cheah, 2016] Rana, T. A. & Cheah, Y.-N. (2016). Aspect extraction in sentiment analysis: comparative analysis and survey. *Artificial Intelligence Review*, (pp. 1–25).

[Ratinov & Roth, 2009] Ratinov, L. & Roth, D. (2009). Design challenges and misconceptions in named entity recognition. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL)* (pp. 147–155).

[Santos, 2012] Santos, T. (2012). Reconhecimento de Entidades Nomeadas em Notícias de Governo. Master's thesis, Universidade Federal do Rio de Janeiro, http://www.cos.ufrj.br/uploadfile/1337948172.pdf. NER.

[Sarawagi, 2001] Sarawagi, S. (2001). Automatic segmentation of text into structured records. In *SIGMOD 2001 Proceedings of the 2001 ACM SIGMOD International Conference on Management of Data* (pp. 175–186).: ACM.

[Sarawagi, 2008] Sarawagi, S. (2008). Information extraction. In *Foundations and Trends in Databases*: Now Publishers Inc.

[Sarawagi & Mansuri, 2006] Sarawagi, S. & Mansuri, I. R. (2006). Integrating unstructured data into relational databases. *Data Engineering, 2006. ICDE 2006. Proceedings of the 22nd International Conference on*, (pp. 1–23).

[Suchanek et al., 2007] Suchanek, F. M., Kasneci, G., & Weikum, G. (2007). Yago: a core of semantic knowledge. In *Proceedings of the 16th international conference on World Wide Web* (pp. 697–706).: ACM.

[Sutton & McCallum, 2006] Sutton, C. & McCallum, A. (2006). An introduction to conditional random fields for relational learning. *Introduction to statistical relational learning*, (pp. 93–128).

[Sutton & McCallum, 2011] Sutton, C. & McCallum, A. (2011). An introduction to conditional random fields. *Machine Learning*, 4(4), 267–373.

[Teixeira et al., 2011] Teixeira, J., Sarmento, L., & Oliveira, E. (2011). *Progress in Artificial Intelligence: 15th Portuguese Conference on Artificial Intelligence, EPIA 2011, Lisbon, Portugal, October 10-13, 2011. Proceedings*, chapter A Bootstrapping Approach for Training a NER with Conditional Random Fields, (pp. 664–678). Springer Berlin Heidelberg: Berlin, Heidelberg.

[Vieira & da Silva, 2015] Vieira, H. d. S. & da Silva, A. S. (2015). A Self-training CRF Method for Recognizing Product Model Mentions in Web Forums. In *Advances in Information Retrieval, 37th European Conference on IR Research - ECIR 2015, Vienna, Austria, March 29 - April 2, Proceedings* (pp. 257–264).: ACM.

[Vieira, 2016] Vieira, H. S. (2016). *Automatically Enriching Product Catalogs with Related Social Media Content*. PhD thesis, Federal University of Amazonas.

[Winkler, 1999] Winkler, W. E. (1999). The state of record linkage and current research problems. In *Statistical Research Division, US Census Bureau*: Citeseer.

[Witten et al., 2011] Witten, I., Hall, M., & Frank, E. (2011). chapter Practical Machine Learning Tools and Techniques, (pp. 629). Morgan Kaufmann.

[Wu et al., 2012] Wu, S., Fang, Z., & Tang, J. (2012). Accurate product name recognition from user generated content. In *Data Mining Workshops (ICDMW), 2012 IEEE 12th International Conference on* (pp. 874–877).: IEEE.

[Yao & Sun, 2014] Yao, Y. & Sun, A. (2014). Product Name Recognition and Normalization in Internet Forums. In *SIGIR Symposium on IR in Practice (SIRIP'14), Gold Coast, Austrália*: ACM.

[Yao & Sun, 2015] Yao, Y. & Sun, A. (2015). Mobile phone name extraction from internet forums: a semi-supervised approach. *Journal of Computational Science*, (pp. 1–23).

[Zhang & Liu, 2011] Zhang, L. & Liu, B. (2011). Entity set expansion in opinion documents. In *Proceedings of the 22nd ACM conference on Hypertext and hypermedia* (pp. 281–290).: ACM.

[Zhu, 2010] Zhu, X. (2010). Conditional random fields. *CS769 Advanced Natural Language Processing*.