Universidade Federal do Amazonas
Instituto de Computação
Programa de Pós-Graduação em Informática

ONILTON DE OLIVEIRA MACIEL JUNIOR

Detecção e classificação de revisões de produtos em ambientes ruidosos

Manaus
2012

Onilton de Oliveira Maciel Junior

Detecção e classificação de revisões de produtos em ambientes ruidosos

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

Orientador: Prof. Dr. Edleno Silva de Moura

Manaus

2011

Onilton de Oliveira Maciel Junior

Detecção e classificação de revisões de produtos em ambientes ruidosos

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

Banca Examinadora

Prof. Dr. Edleno Silva de Moura (Orientador)
Universidade Federal do Amazonas

Prof. Prof. Dr. Altigran Soares da Silva
Universidade Federal do Amazonas

Profa. Dra. Carina Friedrich Dorneles
Universidade Federal de Santa Catarina

Manaus – 2011

Ficha Catalográfica

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

Maciel Junior, Onilton de Oliveira
M152d Detecção e Classificação de Revisões de Produtos em Ambientes Ruidosos / Onilton de Oliveira Maciel Junior. 2012
54 f.: il. color; 31 cm.

Orientador: Prof. Dr. Edleno Silva de Moura
Dissertação (Mestrado em Informática) - Universidade Federal do Amazonas.

1. detecção de produtos. 2. classificação. 3. fóruns. 4. aprendizado de máquina. 5. revisões de produto. I. Moura, Prof. Dr. Edleno Silva de II. Universidade Federal do Amazonas III. Título

Aos meus pais

Agradecimentos

A Deus, pelas pessoas e todas oportunidades até agora.

Aos meus amados pais, Iza e Onilton, exemplos de honestidade, trabalho, fé e sabedoria, agradeço por toda provisão, amor, pelo cuidado na minha educação e todos os valores que me foram passados. Não seria quem sou sem eles.

A minha amada esposa, Cristiane Maciel, que sempre esteve ao meu lado em todos os momentos. Agradeço também a minha sogra, Idalécia, que apesar de não estar mais entre nós, sempre torceu pelo nosso sucesso e acompanhou boa parte dessa caminhada.

A toda minha família, minhas duas irmãs, Isabele e Thamires, tias e tios, e especialmente tia Célia, pelo apoio e suporte.

Sou grato ao meu orientador Prof. Edleno Moura, pelos conhecimentos passados e por esta oportunidade de crescimento acadêmico e profissional. Agradeço pela ajuda, paciência e compreensão ao longo deste trabalho. A minha gratidão também ao Prof. Altigran da Silva, que teve papel importante para conclusão deste trabalho.

Agradeço a todos os amigos e colegas de faculdade e especialmente do laboratório BDRI, por toda ajuda, além da troca de conhecimento e experiência. Também agradeço pelas horas de descontração, tão importantes nessa caminhada.

Minha gratidão a todos que tiveram alguma participação nessa jornada.

Sumário

Lista de Figuras

Lista de Tabelas

Resumo

O comércio eletrônico exercido por varejistas dos mais diversos tamanhos através de suas lojas virtuais é um dos mais lucrativos segmentos da web. Diariamente, um grande número de usuários realiza a compra de produtos através desses sites, em um processo que, não raramente, inicia-se por uma pesquisa sobre diversas informações do produto. As revisões de produto existentes nas lojas virtuais são uma ferramenta que visa auxiliar esse processo ao prover opiniões de outros usuários a respeito de um produto, sem demandar esforço do usuário para obter tais informações. Essa ferramenta, no entanto, possui algumas limitações, entre elas, a quantidade de revisões disponíveis e a dificuldade em obtê-las. Por esse motivo, neste trabalho, apresentamos um método que, utilizando uma base externa de documentos candidatos, pode ser utilizado para encontrar e exibir revisões nas páginas dos produtos correspondentes. Nossos experimentos demonstram que nossa proposta é um alternativa viável para melhorar a experiência dos usuários em sites de comércio eletrônico.

Palavras-Chave: detecção de produtos, classificação, fóruns, aprendizado de máquina, revisões de produto.

Abstract

Eletronic commerce, which has retailers of the most variable sizes, is one of the most profitable segments of the web. Every day, a large number of users buys products through those websites, in a process that, not rarely, begins with a search for product information. The product reviews available at online stores are tools that help this process by providing other users impressions about a product, without demanding any effort from the user for him to get more information. This tool, however, have some limitations, among them, the amount of reviews available and the difficulty to obtain them. Therefore, in this work, we present a method that, using a external database of documents, can be utilized to find and show reviews in the pages of the corresponding product. Our experiments show that our proposal is a practicable alternative to improve the user experience in eletronic commerce websites.

Keywords: product detection, classification, forums, machine learning, product review.

Capítulo 1

Introdução

1.1 Problema e Motivação

Sites de comércio eletrônico têm sido cada vez mais usados, à medida que um número maior de consumidores vem trocando as compras tradicionais por transações eletrônicas. Esse aumento no número de usuários é acompanhado pelo aumento na concorrência entre empresas do segmento e consequentemente na busca por diferenciação. Todos esses fatores culminam na necessidade da criação de ferramentas que auxiliem e facilitem o processo de compra, em todas suas etapas.

As revisões (reviews, em inglês) de usuários são uma das ferramentas disponibilizadas pelos vendedores de comércio eletrônico nas páginas dos produtos. Eles permitem que os próprios usuários deixem opiniões sobre o produto em questão. Em muitos casos também é possível atribuir uma nota ao produto, juntamente com o texto publicado. Essas publicações tem o objetivo de prover mais informações para os consumidores principalmente para que possam ter mais confiança ao adquirir um produto baseados nas experiências relatadas por outros usuários. Essas revisões também podem ser utilizadas pelos fabricantes para identificar problemas com seus produtos ou mesmo para adquirir informações sobre seus competidores [7, 13].

Geralmente submetidos através de um formulário web, em alguns serviços as revisões também costumam apresentar algum meio para que os próprios usuários avaliem a qualidade de uma revisão. Através de uma interface como um botão que permita ao usuário votar se aquela revisão foi útil ou não, a utilidade das revisões pode assim ser estimada pelos próprios usuários [10]. Um exemplo de revisão de usuário pode ser visto na figura 1.1 .

Figura 1.1: Exemplo de revisão de usuário

Apesar de ser considerada uma fonte formidável de opiniões de usuários acerca de produtos, as revisões presentes em sites de comércio eletrônico possuem alguns problemas. Em produtos inseridos recentemente no acervo do site é impossível existir alguma revisão de imediato pois não houve tempo suficiente para algum usuário comprar e analisar o produto. Vale observar que essa situação é ainda mais grave quando o site é recém-criado e, portanto, todos os produtos são novos.

Já no caso de sites amplamente estabelecidos, detentores de um acervo maior de produtos, ainda nota-se o fenômeno da má distribuição das revisões: uma grande quantidade de produtos recebe pouquíssimas revisões enquanto uma minoria dos produtos recebe a maioria das revisões [9].

Há ainda uma hipótese que talvez as páginas dos produtos desses sites não sejam o canal preferido pelos usuários para comentar sobre determinado produto ou tipo de produto. Usuários especialistas sobre determinado assunto costumam postar suas avaliações em blogs ou fóruns de discussão dedicados ao assunto. É possível que as avaliações de melhor qualidade sobre determinado produto não estejam nas páginas

dos vendedores.

Da mesma forma, não é incomum que usuários postem suas reclamações, problemas ou dúvidas sobre um produto em redes sociais, blogs ou fóruns de discussão.

Nesta dissertação propomos um método que, utilizando uma base externa de documentos candidatos, pode ser utilizado para encontrar e exibir revisões na página de um produto. Dado um produto, o método retornará publicações ou revisões relacionadas ao mesmo e que possam auxiliar no processo da compra. Neste trabalho focamos especificamente em reclamações a respeito de possíveis defeitos apresentados pelos produtos.

Os resultados experimentais mostram que o método proposto é capaz de melhorar a precisão dos resultados em relação ao baseline com um ganho de 15%.

1.2 Trabalhos relacionados

Revisões de produtos têm sido utilizados e estudados em vários trabalhos anteriores, principalmente devido ao seu valor comercial.

Jindal et al [9] apresentam um estudo sobre a presença de spam em revisões utilizando uma base de produtos e revisões coletadas da Amazon [1]. Segundo o estudo, uma revisão de produto é classificado como spam quando não é confiável, quando é apenas um revisão da marca do produto (e não do próprio) ou quando o texto não representa uma revisão de fato, chamados de tipos 1, 2 e 3 respectivamente. Uma revisão é denominada não confiável quando o texto não é espontâneo, geralmente construído e submetido propositalmente para denegrir ou inflar as qualidades de um determinado produto. Como estratégia inicial para tratar spam, os autores detectam as réplicas da base, que podem ser consideradas como uma forma de spam. Para detectar o tipo 2 e 3, é construído um modelo utilizando regressão logística e treinando com uma base rotulada manualmente. Dada a dificuldade de determinar

manualmente a confiabilidade de uma revisão, é difícil rotular exemplos de treino para o tipo 1. A estratégia proposta então é utilizar as revisões duplicadas como instâncias positivas de treino, e todas as outras revisões como instâncias negativas.

Kim et al [10] propõem um método para determinar automaticamente a utilidade de uma revisão. Apesar de tal informação poder ser reportada manualmente através do voto dos usuários no site, sabe-se que muitas revisões apresentam poucos ou nenhum voto em relação à sua utilidade. Determinar essa informação automaticamente e utilizá-la para ordenar as revisões pode melhorar significativamente a experiência do usuário ao utilizar o site. Os autores coletaram duas bases da Amazon, uma de tocadores de MP3 e outra de câmeras digitais, incluindo as revisões e os votos de utilidade associados aos mesmos. Para a tarefa de treino, utilizaram regressão SVM. O conjunto de características utilizado contém características estruturais, léxicas, sintáticas e semânticas. O tamanho do texto e o número de frases são exemplos de características estruturais enquanto tf-idf de cada unigrama é um exemplo de característica léxica. Entre as características sintáticas temos como representantes a porcentagem de *tokens* que são verbos e porcentagem que são adjetivos. Por fim, das características semânticas temos a presença de termos que indicam sentimento negativo ou positivo.

Hu et al [7] propõem a geração de resumos baseados em características dos produtos a partir de revisões de um determinado produto. A abordagem desse trabalho distancia-se da sumarização convencional porque os resumos almejados não são especificamente focados no texto das revisões. Ao contrário do que geralmente acontece com esse tipo de problema, os autores não estão procurando os trechos mais representativos do texto. O objetivo é obter uma lista de sentenças que descrevem características do produto de forma positiva ou negativa, por exemplo, “a qualidade da imagem da câmera é incrível”ou “o design desse celular é horrível”, respectivamente. O método apresentando consiste em três passos. O primeiro passo consiste

em obter características que foram comentadas pelos clientes utilizando mineração de dados e processamento de linguagem natural. Após isso, é feita a detecção de sentenças de opinião em cada revisão e é decidido se a sentença é negativa ou positiva. Esse processo é feito detectando adjetivos nas sentenças e posteriormente utilizando a uma lista semente de palavras com suas polaridades e os sinônimos e antônimos contidos na base em inglês WordNet [4] para inferir a polaridade da sentença baseado na polaridade dos adjetivos presentes. No fim, os resultados são agrupados, ordenados e exibidos conforme a frequência da característica, exibindo as sentenças positivas e negativas em duas listas separadas para cada característica.

Tchalakova et al [17] apresentam uma estratégia para determinação de polaridade direcionada a revisões de produtos. Os autores buscam encontrar ocorrências de frases distintas de tamanho máximo em textos previamente classificados como sentimento positivo ou negativo. Ao contrário da maioria dos métodos para determinação de polaridade, o modelo proposto por esses autores não utiliza termos de sentimento pré-definidos. A ideia é extrair estatisticamente as frases de tamanho máximo distintas mais comuns e representativas para cada conjunto, tanto para o conjunto de texto com opinião positiva quanto para o de opinião negativa. As frases obtidas, as informações de frequência em cada conjunto, assim como os algumas informações derivadas são utilizadas como características das instâncias de treino para o aprendizado usando SVM. Os experimentos realizados mostram ganhos em uma base de câmera e outra de livros quando comparados a utilização de características como unigramas ou bigramas, características mais comumente utilizadas nesse tipo de problema.

Ding et al [5] apresentam um sistema cujo objetivo é identificar a que entidades (produtos) cada sentença de um texto se refere. Se a sentença possui nomes de produtos, o sistema tenta identificá-las, essa tarefa é denominada de entity discovery. Caso o nome do produto não seja mencionado na sentença, o sistema tenta inferir o

produto através de pronomes e convenções linguísticas, sendo essa tarefa denominada entity assignment. A primeira tarefa é completada utilizando mineração de padrões sequenciais acompanhado de algumas heurísticas para remover casos errados e refinar os resultados. A segunda tarefa, por sua vez, faz uso de processamento de linguagem natural e de gramáticas construídas pelos autores para processar os textos buscando, entre outras coisas, determinar a polaridade da sentença em relação ao produto. Os autores afirmam que o maior problema dessa etapa são a sentenças comparativas entre dois produtos diferentes.

1.3 Contribuições

Este trabalho tem as seguintes contribuições:

- Um estudo sobre métodos que utilizem a informação contida nas revisões de produtos ;
- Proposta de um método que utiliza publicações de usuários em fontes externas para fornecer mais informações que auxiliem o possível comprador de um produto na sua decisão ;
- Experimentos com o método proposto em conjunto com algumas alternativas possíveis .

1.4 Organização do Trabalho

Esta dissertação está estruturada da seguinte maneira. No capítulo 2 apresentamos alguns conceitos básicos necessários para o entendimento do trabalho. No capítulo 3 descrevemos o método proposto para obtenção de informações sobre produtos a partir de uma fonte externa. No capítulo 4 detalhamos os experimentos

realizados e analisamos os resultados. Por fim, no capítulo 5 apresentamos nossas conclusões e possíveis trabalhos futuros.

Capítulo 2

Conceitos Básicos

Neste capítulo, são explicados alguns conceitos necessários para o entendimento do modelo de proposto para detecção e exibição de revisões. Primeiramente, falaremos da fonte a partir da qual construímos a base de dados necessária para desenvolvimento do método e experimentação, o site Reclame Aqui [3]. Depois falaremos sobre métodos pré-existentes que foram utilizados ao longo deste trabalho.

2.1 Reclame Aqui

O Reclame Aqui [3] é um serviço web colaborativo destinado a receber reclamações de usuários contra empresas em diferentes quesitos como atendimento, compra, venda, produtos e serviços. Ao realizar o cadastro dos dados pessoais no site, o usuário pode enviar uma reclamação, no entanto, os dados do usuário não estão acessíveis publicamente, ficando o seu uso restrito à empresa envolvida para contato e uma possível solução do problema apontado. No entanto, as demais informações da reclamação ficam disponíveis publicamente e podem, inclusive, ser indexadas por máquinas de busca.

Através do site é possível também acessar a lista de reclamações organizadas

por empresa. Na figura 2.1, temos um exemplo, em português, de uma página de reclamação de um usuário sobre um produto no site.

Figura 2.1: Exemplo de reclamação de um usuário

As reclamações geralmente possuem um título, uma descrição e também uma empresa associada. Caso a reclamação tenha sido respondida também aparece abaixo da descrição a resposta da empresa ou fabricante e uma opinião final sobre o atendimento e resolução do problema vindo do usuário que criou a reclamação.

2.2 Conditional Random Fields

Tendo X como uma variável aleatória sobre sequências de dados a serem rotuladas e Y como uma variável aleatória sobre sequências de rótulos correspondentes, Lafferty et al [12] definem um CRF da seguinte maneira:

Seja $G = (V, E)$ um grafo tal que $Y = (Y_v)_{v \in V}$, de modo que Y é indexado pelos vértices de G. Então (X, Y) é um *conditional random field* quando as variáveis aleatórias Y_v, condicionadas em X, obedecem a propriedade de Markov em relação ao grafo: $P(Y_v|X, Y_w, w \neq v) = P(Y_v|X, Y_w, w \sim v)$, onde $w \sim v$ significa que w e v são vizinhos em G.

A figura 2.2 mostra um exemplo de CRF dado que $P(Y_v|X, \text{todos outros Y}) = P(Y_v|X, \text{vizinhos}(Y_v))$. Assim, através da figura, poderíamos dizer que $P(Y_2|X, \text{todos outros Y}) = P(Y_2|X, Y_1, Y_3)$.

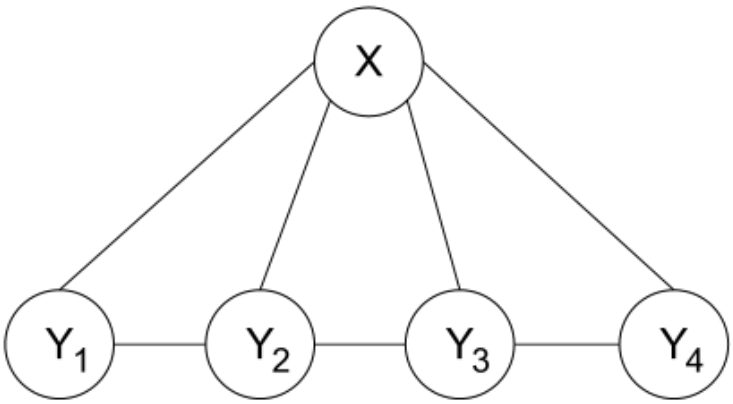

Figura 2.2: Exemplo de CRF

Conditional Random Fields (CRF) são, portanto, modelos gráficos não-dirigidos cujo objetivo é computar a probabilidade $p(Y|X)$ de uma possível saída $Y = (Y_1, ..., Y_n) \in Y_n$ dada a entrada $X = (X_1, ..., X_n) \in X_n$ que também pode ser chamada de observação.

Trata-se de um modelo probabilístico que vem sendo utilizado em diversas áreas,

incluindo processamento de linguagem natural, visão computacional e bioinformática [16]. No contexto de rotulamento de sequências de texto, uma forma especial de um CRF, que é estruturado como um cadeia linear, é especialmente importante e denominado *Linear-chain CRF*. Nesse caso, a probabilidade é dada por:

$$\begin{aligned} P(y|x) &= \frac{1}{Z(x)}\ exp(\sum_{i=1}^{n}\sum_{k}\lambda_k f_k(y_i, y_{i-1}, x) + \sum_{l}\mu_l g_l(y_i, x)) \\ &= \frac{1}{Z(x)} exp(\sum_{i=1}^{n}(\lambda^T f(y_i, y_{i-1}, x) + \mu^T g(y_i, x)) \end{aligned} \tag{2.1}$$

$Z(x)$ é uma constante de normalização. f e g são funções que retornam valores booleanos relacionados a diversas características, sendo que f corresponde às características de transição observadas enquanto g corresponde às características de estado. Por sua vez, $\lambda_1, \lambda_2, \lambda_3, ...$ e $\mu_1, \mu_2, \mu_3, ...$ são parâmetros que determinam a importância de cada característica.

O processo de treino consiste em encontrar os valores para $\lambda_1, \lambda_2, \lambda_3, ...$ e μ_1,μ_2,$\mu_3, ...$ que maximizem a probabilidade $P(y|x)$ em um conjunto de sequências previamente rotuladas.

Após esse processo, para uma nova sequência de texto não rotulada X, o problema de inferência é encontrar a sequência Y mais provável para a observação X [11].

2.3 Modelo Vetorial

O modelo vetorial é um modelo algébrico para representação de documentos de texto proposto por Salton et al [15]. Amplamente utilizado na área de Recuperação da Informação, o escore de similaridade obtido através deste modelo é utilizado como evidência em diversas aplicações entre elas as máquinas de busca na web. O modelo define um espaço vetorial onde cada dimensão corresponde a um termo no

documento (w_i).

Por sua vez, um termo pode corresponder a uma palavra, palavra-chave ou frase dependendo da aplicação. As consultas e documentos então são representadas como vetores:

$$\overrightarrow{V}(d_j) = (w_{1j}, w_{2j}, ..., w_{tj}) \tag{2.2}$$

$$\overrightarrow{V}(q) = (w_{1q}, w_{2q}, ..., w_{tq}) \tag{2.3}$$

Caso um termo w_i ocorra em um documento (d_j), o valor do termo no vetor é diferente de zero ($w_{ij} \neq 0$). Esses valores dos termos podem ser calculados de várias formas utilizando diferentes esquemas de pesos do termo. Entre os mais conhecidos e utilizados está o esquema de pesos TF-IDF.

TF-IDF é uma informação estatística que representa o quão importante um termo é para um documento em uma determinada coleção. É levada em consideração tanto a frequência do termo no documento (TF), quanto uma métrica de importância na coleção (IDF). O peso é dado pela combinação desses dois valores:

$$w_{ij} = TF_{ij} \times IDF_i \tag{2.4}$$

Onde o TF corresponde ao número de ocorrências do termo no documento:

$$w_{ij} = fij \tag{2.5}$$

A intuição por trás do TF é que quanto mais vezes um termo aparece em um documento, mais importante o termo é para esse documento.

Por outro lado, o IDF é dado por:

$$IDF_i = \log\left(\frac{N}{n_i}\right) \tag{2.6}$$

Onde o N refere-se ao total de documentos na coleção e o denominador n_i diz respeito ao número de documentos dessa coleção que possuem esse termo. A razão por trás do IDF é mensurar o quão comum ou o quão raro é um termo entre todos os documentos de uma determinada coleção.

Dado o espaço vetorial, sendo os documentos e as consultas vetores nesse espaço, podemos utilizar operações comuns sobre vetores para compará-los entre si. Assim, podemos calcular a similaridade entre um documento d_j e uma consulta q utilizando o cosseno do ângulo entre os vetores:

$$sim(d_j, q) = \frac{\overrightarrow{V}(d_j) \cdot \overrightarrow{V}(q)}{\left\|\overrightarrow{V}(d_j)\right\| \cdot \left\|\overrightarrow{V}(q)\right\|} = \frac{\sum_{i=1}^{t} w_{ij} w_{iq}}{\sqrt{\sum_{i=1}^{t} w_{ij}^2}\sqrt{\sum_{i=1}^{t} w_{iq}^2}} \tag{2.7}$$

Capítulo 3

Métodos propostos

Nosso método visa aproveitar as informações fornecidas por usuários em outros meios, como fóruns de discussão ou blogs, conforme explicado anteriormente. O método consiste em três etapas, após realizarmos a coleta em um site externo que possa conter as informações desejadas, temos uma etapa de classificação, uma etapa de detecção (ou extração) e finalmente uma etapa de casamento (*match*) conforme pode ser visto no esquema da figura 3.1.

3.1 Classificação

Para descobrirmos se uma determinada publicação é uma reclamação sobre um defeito de um produto, utilizamos aprendizagem de máquina. Modelamos o problema como uma tarefa de classificação onde o retorno do modelo aprendido é 1 se aquela publicação corresponde a uma reclamação de defeito de um produto e 0 caso contrário.

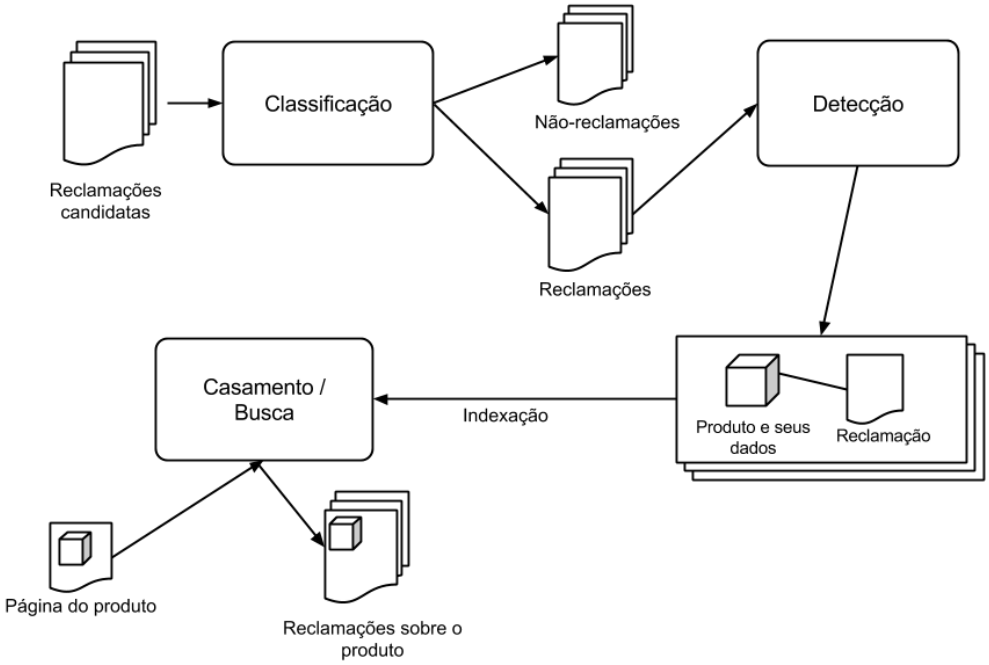

Figura 3.1: Visão geral do modelo

Figura 3.2: Exemplo de página que relata defeito de produto

A figura 3.2 mostra um exemplo de publicação que relata um defeito de um produto. Neste exemplo, vemos uma publicação dirigida à fabricante do produto, a empresa "LG Electronics", onde o usuário relata defeitos no celular *LG Cookie Plus GS900*, entre eles "*descalibração do touchscreen*"e "*travamento ao acessar algumas funções*". Convencionamos que qualquer publicação que relate um defeito em um produto, ainda que não descrevendo explicitamente os defeitos encontrados é classificada como 1 pelo nosso modelo. Isso significa que nosso exemplo ainda seria aceito mesmo que a única frase da publicação fosse "Comprei um celular LG Cookie Plus GS900 e o mesmo apresentou defeito".

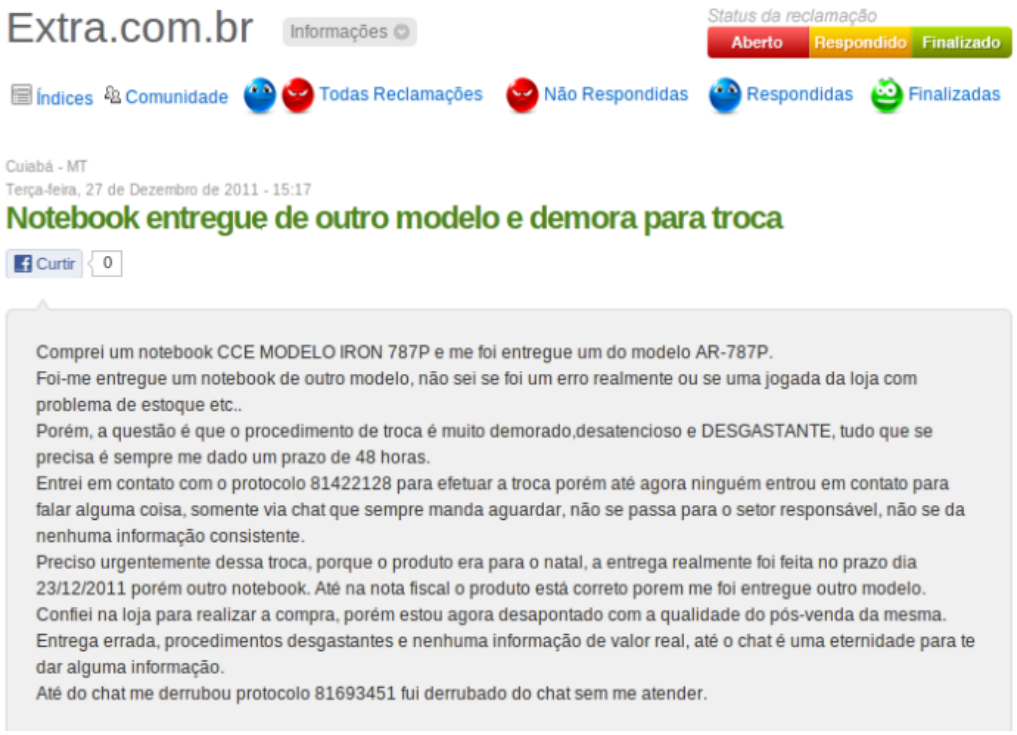

Figura 3.3: Exemplo de página que não relata um defeito de produto

Do lado oposto, a figura 3.3 mostra um exemplo de publicação que não trata de um defeito de um produto e que seria classificada como 0. Agora vemos uma publicação dirigida a um varejista, na qual o usuário reclama de uma suposta troca entre modelos de produtos, onde o modelo entregue é diferente do modelo comprado.

Em nenhum momento um defeito de qualquer um dos dois produtos mencionados é relatado. Outros exemplos semelhantes a esse são atraso na entrega e falhas na compra, entre outros, sendo frequentemente direcionados a varejistas e empresas de comércio eletrônico. É importante notar que mesmo que a publicação esteja direcionada a um varejista, é possível que o cliente relate algum defeito de um produto. Analisando nossa base de dados, percebemos que essa situação é comum e isso pode ser explicado pelo fato dos usuários recorrerem primeiro a empresa da qual adquiriram seu produto.

Com a intenção de manter a solução geral e facilmente adaptável, optamos por apenas remover as *tags* html e utilizar o texto resultante. Outra possibilidade seria utilizar separadamente partes diferentes da página como título da página e conteúdo da página. Sendo mais específico no domínio, poderíamos também utilizar o nome da empresa sobre a qual a publicação foi registrada, título e conteúdo da publicação. No entanto, visando a simplicidade conforme foi explicado, usamos praticamente todos esses campos mas apenas reunidos em um único texto que representa toda a página.

Em relação às características para treino, utilizamos *bag of words* (BOW) a partir das palavras do texto das publicações. Através de experimentos preliminares, percebemos que a retirada de stopwords contribuía para a melhoria do resultado. Então, por esse motivo, esses termos foram removidos. A tabela 3.1 apresenta alguns membros do vetor de característica acompanhados do seus respectivos valores para a publicação da figura 3.2.

...	
celular	2
comprei	1
entrega	0
funcionar	1
problema	1
televisão	0
troca	0
...	

Tabela 3.1: Algumas características e seus valores para a figura 3.2

Para fornecer mais evidências para o modelo desejado, também utilizamos a frequência desses termos em uma base de produtos. Utilizando uma base de produtos fornecida pelo site de comparação de preços Nhemu [2], calculamos a frequência com a qual esses termos ocorrem em diferentes produtos (DF_NH). Lembrando que caso não ocorresse na base, o valor para o termo seria 0. A última característica utilizada é a ocorrência de algum termo na publicação que pertença a uma lista de marcas (MARCA).

3.2 Detecção

Com o objetivo de extrair o produto ao qual a publicação se referia, utilizamos o CRF (*Conditional Random Field*) [12]. Inicialmente as páginas para treino foram usadas apenas removendo as *tags* html. Alguns experimentos e observações nos levaram a concluir que o texto presente nos blocos de template estaria atrapalhando o método. Desse modo, visando melhorar os resultados, removemos manualmente o

template das páginas o que trouxe uma melhoria nos resultados. Para incrementar os resultados do CRF, aplicamos características internas de estado que dizem respeito a ocorrência de termos comumente associados à marca, tipo de produto e atributos. A tabela 3.2 lista essa características e alguns exemplos de termos.

Característica	Termos
marca	huawei|pioneer|leadership|philco|vivitar|tectoy|soyo|casio| zetel|gopro|daten|ever pictures|gradiente|leadership|nokia ...
tipo de produto	camera digital|tv|filmadora|notebook|netbook|celular| blu-ray|monitor ...
atributo	mini-usb|ntsc|gps|infravermelho|estabilizador de imagem| closed caption|tft|lcd|4 kg|2.66 ghz|dual core| 2.27 ghz|polegadas|298 mm ...

Tabela 3.2: Características de estado com exemplos dos termos utilizados

Utilizando o exemplo da figura 3.2 temos o rotulamento como "*... Comprei/OTHER um/OTHER celular/PRODUCT LG/BRAND Cookie/MODEL Plus/MODEL GS290/MODEL na/OTHER loja/OTHER virtual/OTHER submarino/OTHER ...* ".

Ao rotular as instâncias também experimentamos utilizar o rotulamento BILOU conforme utilizado por Ratinov et al[14]. No esquema de rotulamento BILOU, B marca o começo (begin) de um bloco, L marca o último (last) token de um bloco, I marca um token dentro (inside) de um bloco, U indica um único token no segmento e O indica qualquer segmento que não seja um dos anteriores.

Então, para o exemplo da Figura 3.2 temos o rotulamento como "*... Comprei/OTHER um/OTHER celular/PRODUCT-U LG/BRAND-U Cookie/MODEL-B Plus/MODEL-I GS290/MODEL-L na/OTHER loja/OTHER virtual/OTHER sub-*

marino/OTHER ... ”

3.3 Match

Nesta etapa procuramos casar uma publicação a um produto. Para isso, utilizamos o modelo vetorial. Indexamos a base de publicações como um conjunto de documentos cujo conteúdo textual foi extraído na etapa anterior e não o conteúdo original da publicação. Esse passo é dado pela levantamento de todos os termos reconhecidos e rotulados seguido pela concatenação dos mesmos em um documento que será utilizado para indexação e busca. Além disso, utilizamos um campo de valor binário que indica a classificação do documento original obtido na primeira etapa. Podemos ver um exemplo desse processo na figura 3.4. Nesse exemplo, vemos o conjunto de campos e valores extraídos para uma reclamação para uma televisão da marca LG. De posse desses dados, utilizamos todo os valores extraídos no mesmo campo para a concatenação e, no caso da figura, o campo Texto exemplifica o resultado da concatenação. Uma vez construído esse documento transformado, o passo final é simplesmente indexá-lo como um documento de texto comum.

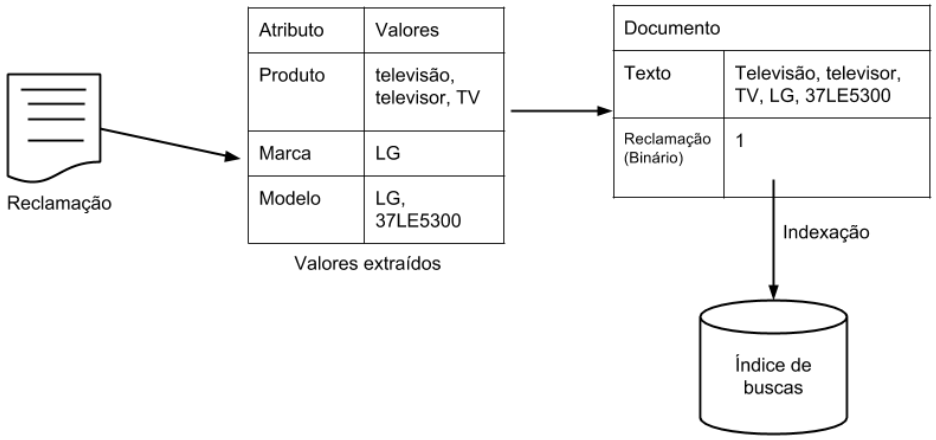

Figura 3.4: Exemplo de reclamação de um usuário

Dado um produto, utilizamos alguns campos específicos do produto como marca, tipo e modelo para construir uma consulta que será utilizada para recuperar as publicações. Os resultados da consulta por um produto então são as publicações relacionadas ao produto. Esse é resultado final do método. A figura 3.5 exemplifica esse processo. Utilizando os dados do produto *Televisão LG 37LE5300*, construímos a consulta "Televisão AND LG AND 37LE5300"que é submetida diretamente ao índice de documentos construído anteriormente.

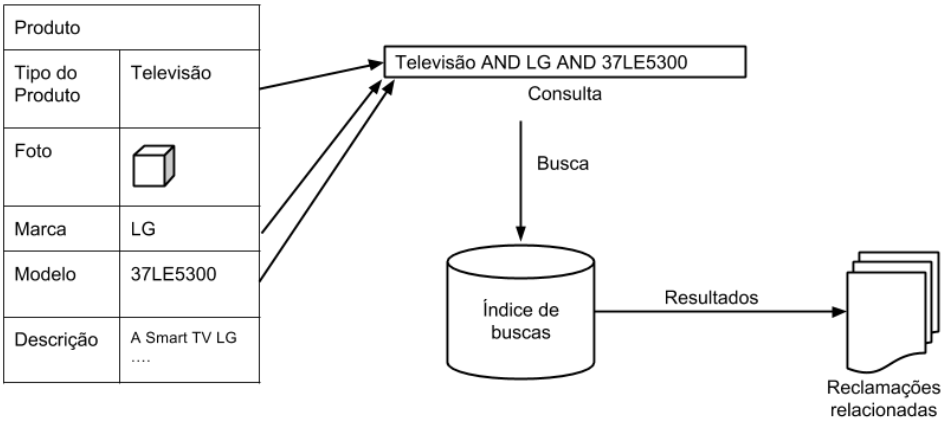

Figura 3.5: Exemplo de reclamação de um usuário

Assumindo que o produto em questão agora fosse o *celular LG GS290* teríamos, por sua vez, a consulta "celular lg gs290"que retornaria entre os resultados o documento da figura 3.2. Apesar de não ser o foco do nosso trabalho, a exibição desses resultados poderia se dar de várias maneiras diferentes. Uma possibilidade seria, para cada resultado, mostrar o título da reclamação acompanhado de um breve trecho do texto onde o produto é mencionado.

Capítulo 4

Experimentos

4.1 Ambiente de Experimentação

4.1.1 Coleção

Nos experimentos utilizamos um conjunto de 628.358 publicações extraídas do site Reclameaqui[3], que contém reclamações relacionadas a empresas que atuam em diversos segmentos como operadoras de telefonia, redes de supermercados, fabricantes de eletrônicos, entre outros.

A coleta foi realizada de forma simples através do uso da página principal como semente, o que implica que todas as páginas da nossa coleção são alcançáveis a partir da raiz do site.

Como o alvo de estudo do nosso trabalho são as publicações dos usuários no site, é necessário remover páginas indesejadas como índices de reclamações de uma empresa, notícias, páginas de navegação entre os menus do site, páginas de dúvidas, entre outras. Essa tarefa foi realizada com uma heurística simples que considera o formato das urls, dado que as urls das publicações dos usuários seguem um padrão bem formado. Alguns exemplos de urls e como a heurística é aplicada podem ser

vistos na Tabela 4.1.

Url	Página de reclamação
http://www.reclameaqui.com.br/noticias/noticias/empresas-estrangeiras-que-atuam-no-brasil-e-nao-atendem-o-br_81/	não
http://www.reclameaqui.com.br/7542011/claro/comprei-celular-e-nao-recebi/	sim
http://www.reclameaqui.com.br/ranking/	não
http://www.reclameaqui.com.br/acompanheaqui/saiba-mais/	não
http://www.reclameaqui.com.br/7539528/gtech-imports/produto-muito-fragil/	sim

Tabela 4.1: Urls filtradas

Embora o número de páginas em nossa base de dados possa ser considerado um número expressivo, durante o desenvolvimento do trabalho observamos que algumas páginas do site não são alcançáveis pelo nosso coletor, uma vez que nem todas as empresas são listadas ou referenciadas diretamente nos menus ou em alguma parte do site. A única forma de coletar essas páginas seria utilizar um formulário do site, porém, optamos por não perseguir essa tarefa e deixá-lo aqui apenas como um problema conhecido, pois, baseados em outros trabalhos anteriores similares, acreditamos que a base construída é suficiente para nossos experimentos.

4.1.1.1 Coleção de Treino

Escolhemos aleatoriamente 100 páginas de reclamação da base de dados inicial, e classificamos manualmente essas instâncias. Alguns ajustes foram necessários para termos uma distribuição justa, onde metade das instâncias de treino são positivas e

a outra metade contém instâncias negativas.

As instâncias positivas são aquelas cuja reclamação apresenta algum tipo de relato sobre um defeito de um produto eletrônico, enquanto as negativas são todos os outros casos.

4.1.1.2 Coleção de Treino CRF

Para construir uma base de treino para o método CRF, da coleção de páginas reclamação, selecionamos 120 instâncias que contêm menções a produtos e rotulamos manualmente. Através de experimentos preliminares, observamos que adicionar algumas instâncias que não contêm menções ao produto ajuda a melhorar o desempenho do método, também contribuindo para maior robustez do método. Por esse motivo, em meio as 120 instâncias que contêm menções a produtos, também inserimos alguns exemplos que não contêm nenhuma menção a produtos, também retirados da coleção de páginas de reclamação.

Em uma outra melhoria proposta para o método, utiliza-se de uma base de dados externa que contém informações de produtos. Utilizamos uma base de dados de produtos eletrônicos pertencente ao site de comparação de preços Nhemu [2]. A base de dados contém descrições de produtos como celulares, Tvs, aparelhos leitores de DVDs, laptops, entre outros. Cada produto possui marca, modelo, atributos e tipo, informações que ao serem devidamente processadas, são usadas como características (*features*) para o método. Um exemplo de produto pode ser visto na Figura 4.1.

Figura 4.1: Exemplo de produto com tipo, marca, modelo e atributos.

4.1.1.3 Base de consulta por produtos

Para avaliar nosso método de forma geral, precisamos de um conjunto de produtos como entrada. Assim, tentando reproduzir um cenário real de uso do sistema, dispomos de 100 consultas por produtos eletrônicos obtidas de forma aleatória de um *log* de consultas do site de comparação de preços Nhemu. Embora as consultas sejam direcionadas a um produto específico, algumas consultas foram modificadas manualmente para conter informações mais precisas e completas dos produtos. Outras foram alteradas com a intenção de suprimir informações desnecessárias que poderiam afetar o método. É o caso da remoção de atributos como "Desbloqueado", "polegadas"entre outras modificações que podem ser vistas na Tabela 4.2.

	Consulta Original	Consulta Final
1	lg GM600 Scarlet GSM Desbloqueado	LG GM600 Scarlet
2	Windows 7 PHN 14545 Philco	Notebook Philco PHN 14545
3	MONITOR 18,5 lg LCD LED E1950T-PN	Monitor LG E1950T-PN
4	tv 32 Polegadas LCD lg 32LH20R	TV lg 32LH20R

Tabela 4.2: Tabela de exemplos de transformação de consultas

4.1.2 Metodologia da Avaliação

Os avaliadores foram orientados a avaliar cada página de reclamação como "relevante"ou "não-relevante". Também receberam a instrução que deveriam avaliar como relevante apenas se julgassem que a página exibida apresentava informações relevantes sobre defeitos do produto consultado.

4.1.3 Métricas de avaliação

Para avaliação dos resultados utilizamos diferentes métricas. Avaliamos a precisão, revocação e a medida F.

A precisão é uma métrica que indica a porcentagem de respostas relevantes no resultado de um sistema de busca. O cálculo dessa métrica pode ser visto na fórmula a seguir:

$$precisao = \frac{|relevantes \cap respostas|}{|respostas|} \tag{4.1}$$

onde *relevantes* se refere ao conjunto de publicações relevantes para um determinado produto, e *respostas* dá nome ao conjunto de publicações retornado pelo método para esse produto.

Por outro lado, no contexto de classificação, a precisão é dada pela razão entre o número de documentos que o modelo aprendido conseguiu classificar de forma correta sobre o total de documentos que o modelo tentou classificar. Da mesma forma, no contexto de extração, a precisão é dada pela razão entre o número segmentos de texto que o modelo conseguiu rotular corretamente sobre o total de segmentos rotulados pelo modelo.

A revocação corresponde ao número respostas relevantes retornadas sobre o total de relevantes existente para determinada busca. Essa métrica visa representar a abrangência dos resultados de um método e é dada por:

$$revocacao = \frac{|relevantes \cap respostas|}{|relevantes|} \tag{4.2}$$

onde *relevantes* e *respostas* foram previamente definidos na equação 4.1 e 4.2, respectivamente.

A medida-F ou F1 é definida como a média harmônica da precisão e revocação. É calculado conforme a fórmula 4.3.

$$F1 = 2 \cdot \frac{precisao \times revocacao}{precisao + revocacao} \tag{4.3}$$

Outra métrica importante é a precisão @ 3, também chamada de p@3, que demonstra a porcentagem de respostas relevantes entre os 3 primeiros resultados retornados. É calculada conforme a fórmula 4.4.

$$precisao@3 = \frac{|relevantes \cap respostas|}{min(|respostas|, 3)} \tag{4.4}$$

onde *relevantes* e *respostas* foram previamente definidos na equação 4.1, respectivamente. O denominador é dado pelo menor valor entre o número de respostas ou 3.

4.2 Resultados Experimentais

4.2.1 Classificação

Na primeira etapa do modelo proposto, utilizamos alguns métodos de aprendizagem de máquina e avaliamos o desempenho de cada um na base de treino escolhida. Os conjuntos de características utilizadas foram o *bag of words* (BOW), cujos resultados são apresentados na tabela 4.3, o BOW+DF_NH+MARCA, cujos resultados são apresentados na tabela 4.4. O conjunto de características BOW+DF_NH+MARCA é composto por, além das características BOW, pelas características relacionadas ao número de ocorrências dos termos na base de dados Nhemu (DF_NH) e a presença de marcas (MARCA). Para confirmarmos os resultados, utilizamos validação cruzada de 10-folds em cada método aprendido.

Método de Aprendizagem	Precisão
Naive Bayes	82%
J48 Tree	84%
1nn	56%
Svm c-csv linear	78%

Tabela 4.3: Precisão do método BOW para os diferentes métodos de aprendizagem

Método de Aprendizagem	Precisão
Naive Bayes	81%
J48 Tree	86%
1nn	58%
Svm c-csv linear	78%

Tabela 4.4: Precisão do método BOW DF NH MARCA para os diferentes métodos de aprendizagem

Ao observamos as duas tabelas, verificamos que, embora não exista uma grande diferença entre os métodos, a precisão tende ser um pouco maior para os métodos BOW+DF_NH+MARCA se comparados aos que utilizam apenas BOW.

Em relação às técnicas de aprendizagem, através da tabela 4.4, percebemos que o método que tem melhor desempenho é a árvore de decisão J48. Por este motivo, foi o método escolhido para as próximas etapas do modelo proposto. Essa árvore foi construída automaticamente utilizando a ferramenta WEKA [6] que foi também a ferramenta utilizada para treinar todos os outros métodos.

Por fim, apresentamos também as matrizes de confusão para os diferentes modelos aprendidos para cada conjunto de características.

Valor real		
V	F	Classificado como
38	11	V
7	44	F

(a) Naive Bayes

Valor real		
V	F	Classificado como
40	9	V
7	44	F

(b) J48 Tree

Valor real		
V	F	Classificado como
16	33	V
11	40	F

(c) 1nn

Valor real		
V	F	Classificado como
37	12	V
10	41	F

(d) SVM C-CSV linear

Tabela 4.5: Matriz de confusão para cada método de aprendizado utilizando BOW

Valor real		
V	F	Classificado como
38	11	V
8	43	F

(a) Naive Bayes

Valor real		
V	F	Classificado como
41	8	V
6	45	F

(b) J48 Tree

Valor real		
V	F	Classificado como
20	29	V
13	38	F

(c) 1nn

Valor real		
V	F	Classificado como
38	11	V
11	40	F

(d) SVM C-CSV linear

Tabela 4.6: Matriz de confusão para cada método de aprendizado utilizando BOW+DF_NH+MARCA

Através das tabelas 4.5 e 4.6, podemos visualizar informações como o número de falsos positivos, que no caso da árvore de decisão J48 e o conjunto BOW corresponde a 9 instâncias. Já para o modelo aprendido através de J48 e BOW+DF_NH+MARCA apresentou 8 falsos positivos.

4.2.2 Extração de produtos

Durante a tarefa de extração de produtos, nosso objetivo era obter as seguintes características de um produto: atributo, marca, modelo e tipo do produto. Um outro atributo "outros"também foi utilizado e consiste em qualquer termo que não representa nenhum dos itens que se deseja extrair.

A tabela 4.7 mostra os resultados obtidos com CRF utilizando com remoção de *template* enquanto a tabela 4.8 mostra os resultados obtidos incluindo BILOU *tagging*. Vemos de imediato o total de precisão demonstrar uma pequeno ganho de um método para o outro. Olhando mais atentamente os atributos, no entanto, vemos que registramos uma certa redução da precisão em alguns atributos. Apesar disso, é possível perceber boas melhorias na revocação na extração de marca, modelo e tipo de produto. Sendo estes atributos determinantes na identificação de um produto, é importante que consigamos extrair uma boa quantidade de atributos em relação ao total existente na base. Além disso, sendo F1 uma medida que leva em consideração ao mesmo tempo e de forma balanceada a precisão e revocação, isso indica que obtivemos um bom resultado, inclusive apresentando uma melhoria visível.

Rótulo	Acertos	Marcados	Manual	Precisão	Revocação	F1
atributo	26	29	63	89.655	41.269	56.520
marca	20	24	94	83.333	21.276	33.897
modelo	21	28	48	75.0	43.75	55.263
outros	6076	6270	6094	96.905	99.704	98.284
tipo do produto	27	36	88	75.0	30.681	43.547
Total	6170	6387	6387	96.602	96.602	96.602

Tabela 4.7: Resultados para CRF (Com remoção de template)

Rótulo	Acertos	Marcados	Manual	Precisão	Revocação	F1
atributo	26	30	63	86.666	41.269	55.913
marca	81	102	94	79.411	86.17	82.652
modelo	31	48	48	64.583	64.583	64.583
outros	6060	6152	6094	98.504	99.442	98.970
tipo do produto	43	55	88	78.181	48.863	60.139
Total	6241	6387	6387	97.714	97.714	97.714

Tabela 4.8: CRF (Com remoção de template + rotulamento BILOU)

Na figura 4.2, há um exemplo de página na qual realizamos a extração.

Figura 4.2: Exemplo de reclamação de um usuário

A tabela 4.9 mostra os atributos extraídos da figura 4.2.

atributo	
marca	samsung,
modelo	GT- S3350 (CH@T335)
outro	
tipo do produto	Celular

Tabela 4.9: Textos extraídos por atributo

No exemplo da tabela 4.9, foi identificado o celular samsung G-S3350, sendo que "samsung" foi extraído como marca (brand), "GT-S3350" como modelo (model) e "celular" foi extraído como tipo do produto (product). Estamos diante de um exemplo de caso ideal onde todos os atributos existentes foram extraídos corretamente.

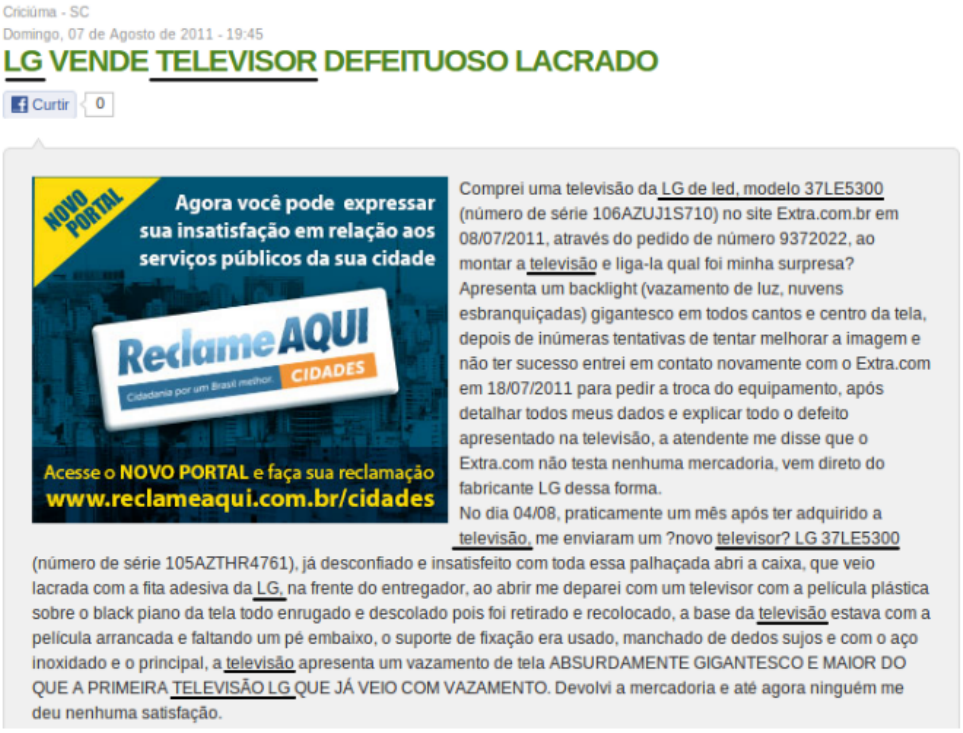

Figura 4.3: Exemplo de reclamação de um usuário

atributo	
marca	LG TELEVISÃO LG televisão LG LG, ,
modelo	37LE5300 modelo 37LE5300 LED 37LE5300 LG 37LE5300
outro	
tipo de produto	televisão televisor televisão, TELEVISOR LG de led TV

Tabela 4.10: Textos extraídos por atributo

Já no exemplo da publicação da figura 4.3 cujos resultados da extração são apresentados na tabela 4.10, vemos que conseguimos identificar a televisão de LED LG 37LE5300, sendo que “LG” foi extraído como marca (brand) e “37LE5300” como modelo (model). Por sua vez, “televisão”, “televisor” e “TV” foram extraídos como

tipo do produto (product).

Nesse caso, também vemos que alguns termos são identificados incorretamente como “Televisão” e “TELEVISÃO” em marca e “LED”, “modelo” e “LG” em modelo. Por fim, “LG”, “de led”, foram identificados erroneamente em tipo de produto.

Apesar disso, embora o método tenha errado na detecção ou na identificação de alguns trechos, na tabela 4.10, o resultado final ainda pode ser utilizado para identificar o produto, uma vez que possui todas as informações que necessitamos para reconhecimento do produto.

Também é importante notar que geralmente os atributos costumam ser mencionados mais de uma vez na publicação como no caso de “37LE5300”(que é mencionado três vezes), portanto, apesar do método errar na extração em uma das ocorrências por falta de contexto na vizinhança, frequentemente ele consegue extrair em outra menção na mesma publicação. Só precisamos extrair o atributo uma única vez, não estamos interessados em todas as diferentes menções ao produto. Desse modo, acreditamos que possuímos resultados satisfatórios para as extrações de marca, modelo e tipo de produto, que serão os campos utilizados na próxima etapa.

4.2.3 Match

Método	Precisão	Revocação	F1	p@3
vetorialAND	0.71428	0.97222	0.82352	0.53846
vetorialAND + Class	0.79508	0.89814	0.84347	0.66667
VetorialAND+Template	0.70635	0.82407	0.76068	0.53846
VetorialANDTemplate+Class	0.78846	0.75925	0.77358	0.65217
CRF + Templ	0.65625	0.19444	0.30000	0.57143
CRF + Templ + Class	0.73076	0.17592	0.28358	0.71429
CRF + Templ + N3	0.71250	0.52778	0.60638	0.51245
CRF + Templ + N3 + Class	0.79412	0.50000	0.61364	0.64296
CRF_BILOU + Class	0.82812	0.49074	0.61627	0.72641

Tabela 4.11: Resultados para os métodos implementados

Nesta etapa, a tabela 4.11 apresenta um comparativo entre os melhores resultados obtidos. Nesse comparativo, utilizamos a base original de reclamações ou posts como base para consulta por produtos, apresentado como vetorialAND, nosso baseline. A remoção de *template* e a remoção de publicações que não sejam classificados como reclamações de defeitos de produtos é sinalizada como Template e Class, respectivamente. Pelo lado do nosso método, experimentamos utilizar também os 3 termos vizinhos ao item extraído pelo CRF, apresentado como N3.

É possível perceber um ganho no método CRF_BILOU + Class na precisão. Da mesma forma, CRF_BILOU + Class consegue o melhor valor de p@3, embora próximo de CRF + Templ + Class na p@3.

No nosso problema, é muito importante ter bons resultados em precisão e, mais especificamente, nos 3 primeiros resultados, visto que na aplicação mais imediata do nosso método, a ideia principal é fornecer um número fixo de resultados que não

tome muito espaço na página onde o usuário está visualizando um produto.

Em relação à revocação, podemos ver que vetorialAND tem o melhor valor, o que é esperado, uma vez que, basicamente esse método casa com qualquer publicação que simplesmente contenha todas as palavras da consulta, assim, é o método que possui a maior abrangência, sendo que na maioria das vezes, os resultados dos outros métodos podem ser pensados como apenas um refinamento dos resultados do vetorialAND.

Vale ressaltar que nossa etapa de classificação contribui tanto para a melhoria do nosso método quanto para o baseline, é possível ver um aumento na precisão dos métodos que utilizando a classificação (Class) como o vetorialAND+Class e vetorialANDTemplate+Class sobre o original vetorialAND, comprovando a importância da classificação.

O método CRF + Templ + Class já consegue uma melhoria na precisão @ 3 do em relação aos métodos derivados do vetorialAND. Também conseguimos perceber a melhoria agregada pelo uso da classificação quando comparamos ao CRF + Templ. Apesar disso, percebemos uma revocação baixa, principalmente devido ao rotulamento incorreto e a própria revocação inferior apresentada na etapa de extração. Ainda que estejamos mais preocupados com a precisão, acreditamos que é importante termos uma revocação mais aceitável.

Ao observamos os resultados da extração, percebemos que várias vezes alguns termos eram rotulados corretamente, entretanto os vizinhos que tinham o mesmo rótulo não eram assinalados corretamente. Assim, adotamos uma heurística simples de aproveitarmos os vizinhos da direita e da esquerda juntamente como termo selecionado. Nessa situação obtivemos o melhor resultado utilizando 3 vizinhos de cada. Então para CRF + Templ + N3 e CRF + Templ + N3, observamos uma melhoria na revocação comparando com CRF + Templ + Class.

Finalmente, aplicando o rotulamento com a estratégia BILOU durante a etapa de extração, conseguimos resolver alguns dos problemas detectados na extração e

melhorar a extração. Como dito anteriormente, obtemos o melhor resultado para precisão e p@3 em CRF_BILOU + Class. É importante ressaltar que também conseguimos um melhor valor de revocação.

Capítulo 5

Conclusões e Trabalhos Futuros

Neste trabalho apresentamos um método capaz de usar informações externas para enriquecer a experiência de usuário durante a compra de um produto. O objetivo do método é, dado um produto que está sendo visualizado, exibir reclamações relacionadas ao produto relatadas por usuários.

O método proposto utiliza publicações de usuários que passam por três etapas durante o processo. Na primeira, classificamos as publicações e separamos as reclamações sobre defeitos de produto das outras publicações. Em seguida, realizamos a detecção e extração do produto mencionado utilizando os atributos, marca, modelo e tipo de produto. Na última etapa, realizamos o casamento através de uma consulta que representa o produto visualizado e uma base que representa os produtos mencionados nas publicações.

Utilizando uma base de publicações coletada da web, mostramos que o nosso método supera o baseline na precisão, o que torna o sistema mais confiável para utilização em sistemas de comércio eletrônico no mundo real. Através dos nossos experimentos, demonstramos que o CRF_BILOU + Class consegue superar o vetorialAND na precisão @ 3, inclusive mostramos que é possível melhorar o baseline utilizando nossa etapa de classificação, a qual chamamos de vetorialAND + Class.

Para trabalhos futuros, sugerimos a exploração de diferentes formas de exibição dos resultados retornados para o usuário. Uma opção mais imediata seria a utilização de resumos do texto em relação à consulta, fazendo uso de uma estratégia similar à empregada por máquinas de busca atuais.

Outra possibilidade de trabalho é verificar os resultados em outra base e para um domínio de produtos diferente de eletrônicos, avaliando as possíveis diferenças e necessidades de adaptação.

Um sugestão também interessante para o futuro é utilizar o trabalho desenvolvido por Hu et al [8] para, ao invés de exibir as publicações que relatam defeitos do produto, exibir especificamente cada uma das sentenças que relatam o defeito do produto. Apesar de nossa base apresentar algumas diferenças em relação a da base de revisões utilizadas pelos autores, é possível também estudar a possibilidade de agrupar as sentenças que descrevem um defeito de produto de acordo com as diferentes características e funções de um produto.

Referências Bibliográficas

[1] **Amazon.com**. `http://amazon.com/`. [Acessado em 20-Fevereiro-2013].

[2] **Nhemu**. `http://www.nhemu.com.br/`. [Acessado em 02-Abril-2012].

[3] **Reclameaqui**. `http://www.reclameaqui.com.br/`. [Acessado em 02-Abril-2012].

[4] **WordNet**. `http://wordnet.princeton.edu/`. [Acessado em 20-Fevereiro-2013].

[5] Ding, X., Liu, B., and Zhang, L. Entity discovery and assignment for opinion mining applications. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining* (New York, NY, USA, 2009), KDD '09, ACM, pp. 1125–1134.

[6] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. The weka data mining software: An update. *SIGKDD Explor. Newsl. 11*, 1 (Nov. 2009), 10–18.

[7] Hu, M., and Liu, B. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining* (New York, NY, USA, 2004), KDD '04, ACM, pp. 168–177.

[8] Huang, Y., Contractor, N., and Yao, Y. CI-KNOW: recommendation based on social networks. 27–33.

[9] Jindal, N., and Liu, B. Opinion spam and analysis. In *Proceedings of the international conference on Web search and web data mining* (New York, NY, USA, 2008), WSDM '08, ACM, pp. 219–230.

[10] Kim, S.-M., Pantel, P., Chklovski, T., and Pennacchiotti, M. Automatically assessing review helpfulness. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing* (Stroudsburg, PA, USA, 2006), EMNLP '06, Association for Computational Linguistics, pp. 423–430.

[11] Klinger, R., Tomanek, K., and Klinger, R. Classical probabilistic models and conditional random fields, 2007.

[12] Lafferty, J. D., McCallum, A., and Pereira, F. C. N. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning* (San Francisco, CA, USA, 2001), ICML '01, Morgan Kaufmann Publishers Inc., pp. 282–289.

[13] Liu, B. *Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data.* Data-Centric Systems and Applications. Springer, 2007.

[14] Ratinov, L., and Roth, D. Design challenges and misconceptions in named entity recognition. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning* (Stroudsburg, PA, USA, 2009), CoNLL '09, Association for Computational Linguistics, pp. 147–155.

[15] Salton, G., Wong, A., and Yang, C. A vector space model for automatic indexing. *Communications of the ACM 18*, 11 (1975), 613–620.

[16] Sutton, C., and McCallum, A. An Introduction to Conditional Random Fields. *ArXiv e-prints* (Nov. 2010).

[17] Tchalakova, M., Gerdemann, D., and Meurers, D. Automatic sentiment classification of product reviews using maximal phrases based analysis. In *Proceedings of the 2nd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis* (Stroudsburg, PA, USA, 2011), WASSA '11, Association for Computational Linguistics, pp. 111–117.